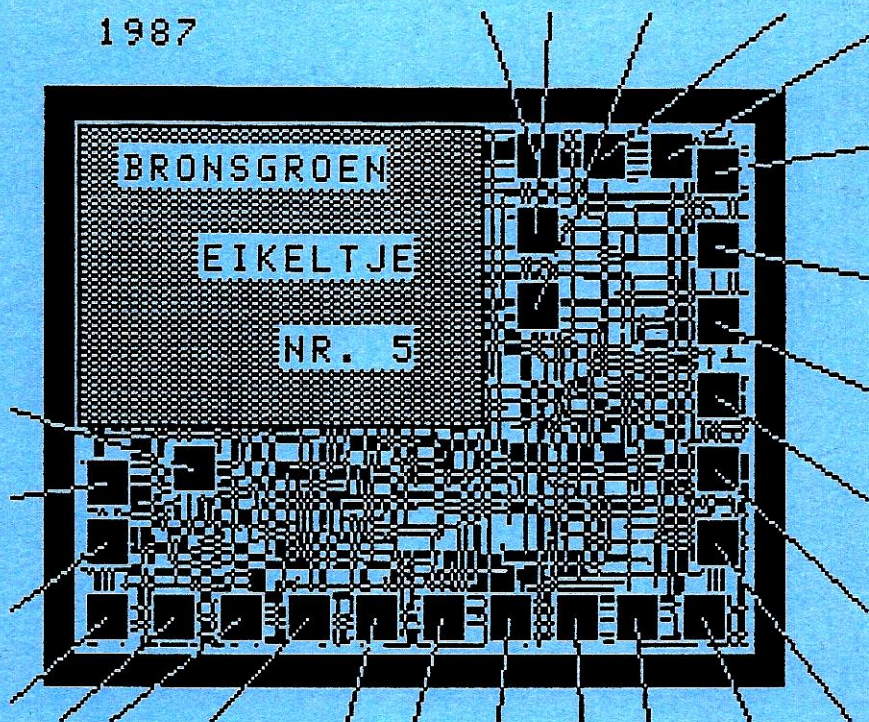


1987



ATOM COMPUTERCLUB

LIMBURG

prijs f 1,--

I N H O U D S O P G A V E

Inhoudsopgave	SPS		2
Voor de harde jongens	HARDWARE-TRACER	J. Swinkels	3
Eikeltjesmarkt			10
Diverse handige programma's van		R. Leurs	
w.o.:	Escape melding		11
	Atom als versterker (tje)		12
	CHAIN voor COS gebruikers		12
	CHECKER om prog. foutloos over te kunnen typen		13
	SPACE ROBOTS spelletje voor joystick (GAGS2.3)		14
	Voorgeschakelde eprom steeds in beeld		17
Belangrijke adressen			20

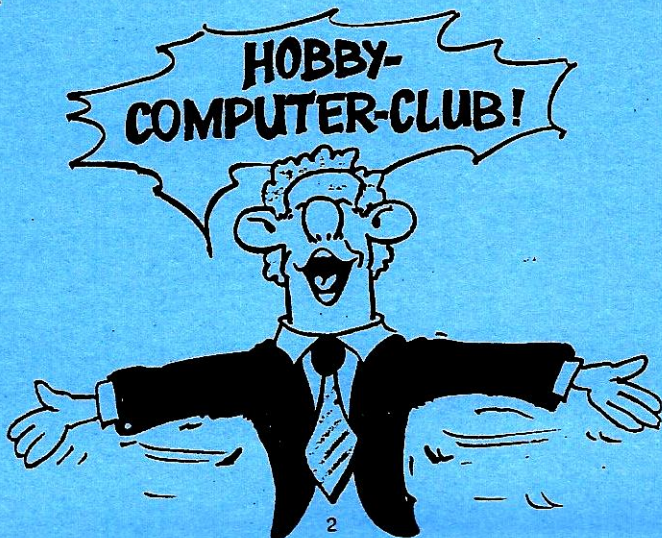
S P S

Sanders Print Service

Cassette interface print	5.00
MDCR interface print	5.00
Batterie backup printje	3.00
BK hoge gheugenprintje	5.00
#E000 naar #1000 voor oude schakelkaart	3.00
Omschakelprintje voor 80K kaart	3.00
Boostrapprintje de Moor	6.00
Voedingsprint MDCR 12V	5.00
Accoustische verbindingstester	5.50

Al deze printen zijn ook gebouwd en getest te bestellen tegen kostprijs onderdelen en een "symbolisch" bedrag. Even bellen graag. Tel. 04750-30401.

Bestellen: op de clubavond of door overmaking van het bedrag + 1,00 + vermelding van print op gironr. 794739 tnv E. Sanders, te Herten.



Tijdens het testen van mijn "Euro Atom" had ik de behoefte aan een "step mode", waarmee je een programma stap voor stap kunt laten runnen. Iedereen denkt dan natuurlijk meteen een de josbox, waar zo'n statement inzit. Helaas werkt mijn gehele systeem niet, waardoor dit dus uitgesloten wordt. Ik ben toen eens in verschillende electuurs gaan zoeken en vond daar allerlei "6502 tracers", zoals ze die noemen, maar deze werkten allen op interrupt basis, iets waar ik dus geen fluit mee kon doen.

De conclusie die ik hieruit trok was dat er dus geen echte hardware tracer voor de 6502 bestond. Dus wat doet een hardware man dan, die ontwerpt, maakt en gebruikt zijn eigen ontworpen tracer.

Zo gezegd, zo gedaan en zie hier. Naar alle waarschijnlijkheid de eerste hardware 6502 tracer in Europa!.

Aan alleen een tracer heb je niet heel erg veel, omdat je dan telkens aan de adres- en data-bus moet meten om er achter te komen op welk adres welke data staat. Daarom heb ik meteen ook maar een status-analyser ontworpen die het adres op de bus omzet in een leesbaar adres op het display. Hetzelfde heb ik met de databus gedaan. Door nu de cpu step by step te laten lopen zie je telkens welk adres en data op de bus staat.

Dit is niet het ei van columbus maar de 6502 cpu tracer van Jan Swinkels .

De kosten van deze tracer zijn ongeveer FL.115,00 incl. print.

Snelle hardware debugging van een programma is nu mogelijk zonder dat je gebruik van je interne registers maakt, waardoor het programma ongestoord verder werkt.

Zijn er personen die menen dat een 6502 processor ook te besturen is met een One-shot op de klokingang (pin 37), dan hebben ze een grote uitvinding gedaan want volgens de data sheets kan dit niet. Dit heb ik al reeds geprobeert en inderdaad bevestigd.

Bij een klokfrequentie lager dan 843 KHz, is een 6502 processor niet meer thuis.

Bij deze tracer wordt, d.m.v. een signaal dat er speciaal voor ontworpen is, de cpu stilgezet, terwijl de klok nog doorfietst op 1-of 2-MHz.

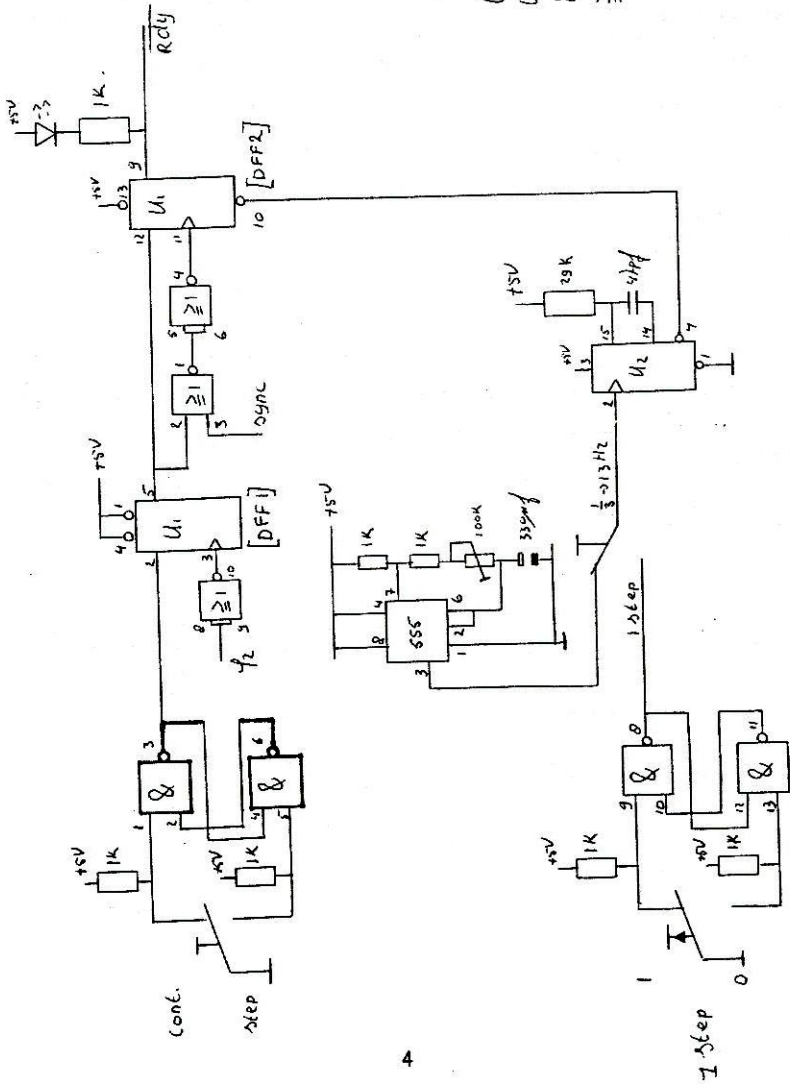
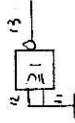
Dit Ready-sigitaal is bij een 2-80 processor ook aanwezig maar heet dan Halt. Het is dan rechtstreeks met een DFF aan te sturen zonder dat er eerst gekeken wordt naar andere signalen. Dit wordt intern allemaal geregeld.

Een 6502 is nu eenmaal een wat oudere processor, maar daardoor nog lang niet de slechtste.

SCHEMA 6502-TRACER

Jan Swinkels
 Dorpsstraat 41
 6893 EA Heythuysen
 TEL.04749 1235

$U_1 = 74LS74$
 $U_2 = 74LS125$
 $\& = 74LS00$
 $\geq 1 = 74LS02$



Werking Tracer en Status Analyser.

Op de tekening zijn twee schakelaars te zien. S voor in de step-mode en C voor de cpu een stap verder te verplaatsen. Indien de spanning op het systeem gezet wordt, dan zal de ready lijn hoog worden (er wordt automatisch een 1 in de data flip-flops [DFF] 1 en 2 geklokt). Hierdoor zal het systeem normaal opstarten.

Indien nu de schakelaar S op de step-mode gezet wordt dan zal uitgang Q een 0 worden. Deze 0 wordt aan de ingang van DFF1 aangeboden. Indien nu een instructie wordt opgehaald, zal fi 2 laag worden. De geïnverteerde fi 2 zal dus hoog worden en deze 0 inklokken.

Deze nul komt nu aan de ingang van DFF2 te staan. Doordat de uitgang van DFF1 nu laag wordt zal de Nor-poort niet meer geblokkeerd zijn voor het sync signaal.

Enige tijd nadat fi 2 laag geworden is (350 nsec max) zal de sync hoog worden. Hierdoor zal de data in DFF2 geklokt worden. Doordat deze data 0 was, zal de Ready lijn laag worden en dus de cpu stoppen. De cpu stopt dus op het moment dat de cpu een fetch heeft gedaan.

De cpu staat nu volkomen stil en op het display is nu de status van de cpu af te lezen. In deze stand kan de cpu gerust een hele nacht verblijven (heb ik uitgetest!).

Door nu op de schakelaar C te drukken zal de MMU een set-sigitaal genereren van ongeveer 40 nsec waardoor er een 1 in DFF2 wordt geklokt, de cpu loopt weer verder (schak. C is een terugverend type).

De instructie waarmee de cpu juist begonnen is wordt nu afgemaakt. Als deze instructie afgelopen is, wil de cpu weer een volgende instructie ophalen, echter het Sync-sigitaal klokt weer een nul in het DFF2 waarmee de cpu zichzelf weer stilzet. Op deze wijze is het dus mogelijk om step-by-step door een programma te lopen.

Bij een schrijffactie zal het Ready-sigitaal automatisch intern in de cpu genegeerd worden.

Als we de cpu weer "op volle toeren" willen laten draaien dan schakelen we schakelaar S weer terug in de stand Continue en er gebeurt niets. Dit klopt want op dit moment staat de cpu nog stil.

Wordt op de schakelaar C gedrukt, dan wordt DFF2 geset en daardoor de cpu weer opgestart. Bij de eerst volgend instructie zal deze 1 in DFF1 ingeklokt worden. Het Sync-sigitaal wordt weer hoog en automatisch deze 1 inklokken in DFF2.

Hiermee wordt automatisch de Nor-poort gesloten waardoor de cpu nu niet meer in de step-mode staat.

Schakelaar "var." dient ervoor om in de step-mode, de cpu weer automatisch op te laten starten. D.m.v. de potmeter is de frequentie in te stellen van 0.3 Hz tot 13 Hz. Je hebt dan een volautomatische stepper. Bij lange programma's is dit gemakkelijk en kost je geen blauwe vinger meer.

De tracer werkt uitstekend. Alleen is er nu nog het probleem dat ik steeds met een scoop op de processor moet gaan meten om erachter te komen op welk adres, welke data staat. Omdat dit een zeer tijdrovend en vervelend werkje is ben ik op zoek gegaan naar IC's die een hexadecimale code omzetten in een zevensegments code. IC's hiervoor bestaan er wel maar, omdat die niet vaak gebruikt worden, zijn deze peperduur FL 11,70 per stuk (ik heb er 6 nodig). "9368".

Ik ben toen eens gaan informeren wat prom's kosten. Deze bleken de helft goedkoper te zijn en heb daarom besloten om zes prom's te kopen en ze dan zelf te programmeren.

Van een oude wekker heb ik het display afgesloopt en met nog wat weertandjes heb ik nu een complete status-analyser gemaakt.

Met de tracer zet ik de cpu stop en de analyser zet de gegevens van de cpu om op een zevensegments display.

D.m.v. een schakelaar is het mogelijk om de cpu continue te laten werken of in de step-mode. Met een klein drukknopje is het nu mogelijk om in de step-mode de cpu step-by-step te laten lopen. Indien de cpu op "volle toeren" draait, dan zullen alle display's het cijfer 8 laten oplichten.

In de step-mode kan precies worden nagegaan op welk adres de cpu stilstaat en op het data display is dan de data van de cpu zichtbaar. Het mooie van deze analyser is dat bij een read- of write-operatie de data niet door de analyser beïnvloedt wordt, iets waar ik eerst bang voor was.

Nu is het ook voor een hardware man mogelijk om eens te achterhalen hoe het operating-system werkt, zonder dat hij zich zorgen hoeft te maken bij een branch of de cpu nu wel of niet zal springen en waar naartoe. De status-analyser zal hem precies laten zien waarmee de cpu bezig is.

Het debuggen van zelfgemaakte programma's is nu een fluitje van een cent geworden (nauw ja, 1 cent, de complete print gaat FL.115,-- kosten, all-in)

Als ik deze JGS-tracer met status-analyser 9 maanden eerder had gebouwd dan was de euro-Atom al bijna weer ouderwets geworden. Het is me nu wel duidelijk geworden dat ik een computersysteem ontworpen heb dat in feite meteen werkte, maar door een rare kronkel in de software, 9 maanden later pas operationeel werd. Deze tracer met analyser heeft mij tot nu toe al meer plezier gegeven, dan de Atom in al die jaren en me ooit zal kunnen geven.

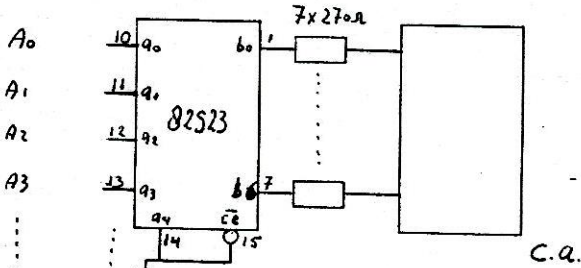
Eigelijk is deze print bedoeld voor iemand die systemen ontwerpt. In principe is deze analyser op ieder processor aan te sluiten. De pinbezetting van de cpu-connector en het Ready-of Halt-sigitaal moet veranderd worden. Het idee zal echter gelijk blijven.

Diegene die nu nog menen dat de hardware commissie niets doet zal waarschijnlijk bedoelen dat sommige figuren zich met zaken bezig houden waarmee de Atomist niets zal kunnen doen.

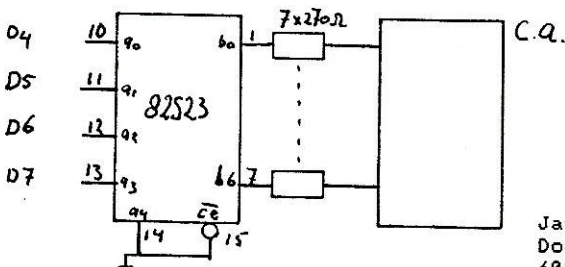
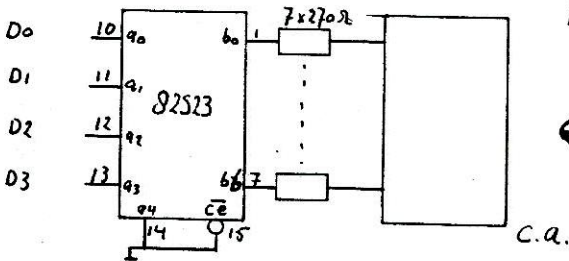
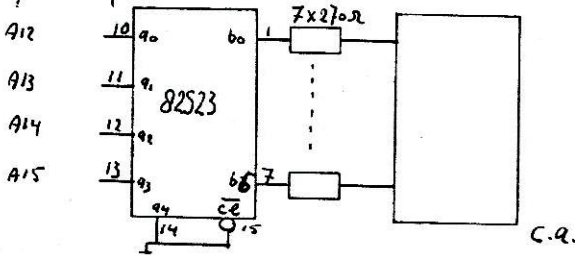
B.v Turbo op MDCR, tracers en statusanalysers waar geen hond wat mee kan doen.

Ik ben me dit terdege bewust en zal nu eerst eens het " ruisprobleem " op de Atom onder hand gaan nemen. Ik meende toch eerst de voorafgaand dingen te moeten realiseren omdat ikzelf hiermee meer belang heb. Je kunt beter een werkende Atom hebben die soms wat sneeuw op het scherm tevoorschijn tovert, dan een Atom die helemaal niets doet.

SCHEMA 6502 STATUS-ANALYSER.



a
11 | b
e | 5 | c
d
segmenten



Jan Swinkels
Dorpsstaat 41
6093 EA Heythuysen
04749 - 1235.

Programmering PROM's

De PROM's zijn van het type 82S23. De 5 adres-ingangen zorgen dat er 8 data-uitgangen beschikbaar zijn. De PROM heeft open collector uitgangen, waardoor ze direct op een zeven-segment kunnen worden aangesloten. D.m.v. weerstanden wordt de stroom binnen de specificatie gehouden. De uitgang mag max. 20 mA leveren bij 1,7 Volt. De weerstand van 270 Ohm zorgt voor een stroombegrenzing van $5V/270 \text{ Ohm} = 18,5 \text{ mA}$. We hebben maar 4 adres-ingangen nodig, vandaar dat we er een aan massa leggen (A5). Van de 8 uitgangen hebben we er maar 7 nodig voor het zeven-segment display. We gebruiken dus een uitgang (D7) gewoon niet.

Voor de aansturing van de segmenten dienen we er rekening mee te houden welk display we gebruiken, een met C.A of een met C.C.

De geprogrammeerde waardes moeten dan juist geïnverteerd worden. Bij de PROM wordt indien de data "1" is, de uitgang hoog waardoor de led van het segment niet oplicht.

De tabel voor C.A display's staat hieronder afgebeeld.

adres	cijfer	segmenten aan							Data in PROM								
		G	F	E	D	C	B	A	D	7	6	5	4	3	2	1	0
0000	0		F	E	D	C	B	A	0	1	0	0	0	0	0	0	0
0001	1						C	B	0	1	1	1	1	0	0	0	1
0010	2	G		E	D			B	0	0	1	0	0	1	0	0	0
0011	3	G			D	C	B	A	0	0	1	1	0	0	0	0	0
0100	4	G	F				C	B	0	0	0	1	1	0	0	0	1
0101	5	G	F		D	C		A	0	0	0	1	0	0	0	1	0
0110	6	G	F	E	D	C		A	0	0	0	0	0	0	0	1	0
0111	7						C	B	0	1	1	1	1	0	0	0	0
1000	8	G	F	E	D	C	B	A	0	0	0	0	0	0	0	0	0
1001	9	G	F		D	C	B	A	0	0	0	1	0	0	0	0	0
1010	A	G	F	E		C	B	A	0	0	0	0	1	0	0	0	0
1011	B	G	F	E	D	C			0	0	0	0	0	0	0	1	1
1100	C			F	E	D		A	0	1	0	0	0	1	1	0	0
1101	D	G		E	D	C	B		0	0	1	0	0	0	0	0	1
1110	E	G	F	E	D			A	0	0	0	0	0	1	1	0	0
1111	F	G	F	E				A	0	0	0	0	1	1	1	0	0

Helaas ben ik niet in staat om een printontwerp van deze tracer en status-analyser te maken. Met het plakken van printbanen en vervaardigen van printen ben ik reeds 6 jaar geleden mee gestopt. Het ontwerpen van schakelingen en die daarna met mijn "roller-road pen" systeem werkt zo goed, dat ik definitief van het vervaardigen van printen, met de "ouderwetse" methode, ben afgestapt. Het gebeurt te vaak dat er een wijziging moet worden aangebracht. Bij geestste printen is dit nooit mogelijk op een fatsoenlijke manier. Het doorkrassen van printbanen vind ik zo'n "boerse en onbeholpen" manier dat ik hieraan niet mee wil helpen.

Zijn er personen die graag deze print willen hebben verzoek ik om zelf een print te maken. Het schema is door mij vrij gegeven. Het prototype, dat gebouwd is d.m.v. mijn "roller-road pen" systeem werkt zeer goed. Het schema mag daarom als foutief beschouwd worden.

Het is misschien wel verstandig om na te gaan wie er nog meer een print wil hebben, waardoor er maar een ontwerp gemaakt wordt en niet zoals bij de MDCR dat er 10 versie's zijn die allemaal iets anders "kunnen".

Bij de status-analyser heb ik gebruik gemaakt van PROM's die ik geprogrammeerd heb met de Electuur PROM-programmer.

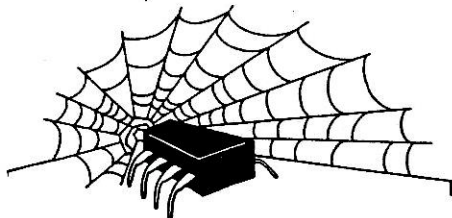
Deze staat beschreven in Electuur juli/augustus 1980. Printnummer 80556.

Voor diegene die geen PROM-programmer bezitten wil ik wel de PROM's programmeren. Het is ook mogelijk om de bestaande IC's te gebruiken. Hiervoor wordt gebruik gemaakt van IC 9368. Dit is een BCD-Hex decoder. De werking van dit IC garandeer ik niet omdat ik met dit IC geen ervaring heb.

Ik wens de Heren succes met het ontwerp van de print en alvast een plezierige vakantie, voor zover deze nog niet voorbij is.

Groetjes,

Jan Swinkels



 EIKELTJES MARKT

TE KOOP AANGEBODEN:

Home-computer ACORN ATOM
 Uitgebreidt tot 32K RAM met spec. Lab. voeding 7 Amp.
 Eprom-kaart met div. EPROM's
 Cassette recorder met div. programma's en boeken.
 Eventueel met monitor (groen beeld)

Te bevragen bij: tel. 045-751386

TE KOOP GEVRAAGD:

ATOM disk pack
 Prijs nader overeen te komen.

Te bevragen bij: Kierkels



TE KOOP AANGEBODEN:

Wegens overcompleet te koop aangeboden:

- Acorn Atom, geheel bezet + editor + eprommer +
 I/O experimenteerboard + combikaart + joystick +
 boekwerken, BGE's en AAN's 250,00
- Z80-kaart (geheel gebouwd) 250,00
- Atom drive 195,00
- IBM Selectric I/O typewriter + linten + papier 125,00
- VIC20's zonder kast:

 - getest 45,00
 - ongetest/defect in voeding 35,00
 - los keyboard (ook voor CBM64) 25,00
 - los VIC20 board 25,00

- Hallicrafters SX110 ontvanger geschikt voor ontvangst
 van morse, telex, amtor, tor, packet-radio.
 Inclusief org. LS en verhuistrafo 425,00
- Realistic DX190 en SP150 ontvanger+LS.
 Geschikt voor dezelfde doeleinden als hierboven 300,00

Alles is compleet met documentatie en in zeer goede staat of
 nieuw.

- Diverse printen met beschrijving van de elektuur
 labvoeding 35V/3.5A regelbaar 15,00

Voor mensen die eventueel interesse hebben in de "world modem
 chip" AM7910, die kan ik bij voldoende interesse uit de brand
 helpe voor circa fl. 45,00
 (Afhankelijk van de dollar-dagprijs).

Te bevragen bij: Maarten Hodenius

*** EEN VERHAALTJE door ROLAND LEURS ***

Inleiding:

Nadat ik een hele tijd niets meer heb laten publiceren in ons clubblad vond ik het hoog tijd worden dat daar eens verandering in kwam. Het resultaat ligt voor U.

Hoofdstuk 1: WRITE-PROTECT OP 32 K-KAARTJE

Om te beginnen heb ik heel lang geleden aan Henk Bastings gevraagd of het mogelijk was om op het 32 K-kaartje write-protect te zetten. Uiteraard mogelijk zonder veel kosten en zeer handig. Hiervoor zijn slechts een weerstand en een schakelaartje nodig.

Van de 6264's die onder wp komen te staan (bijv #4000-#8000) buigt U pin 27 uit. Tussen de pin en het voetje plaatst U de schakelaar en tevens soldeert U tussen pen 27 en pen 28 een weerstand van ongeveer 4k7. Als U meerdere 6264's tegelijk wilt schakelen dan gewoon alle pinnen 27 uitbuigen en doorverbinden. Dat is alles.....

Hoofdstuk 2: MELDING NA 'ESCAPE'

Iets heel handigs: de melding ESCAPE AT LINE xxxx als U op escape drukt. Dit kan handig zijn als U een fout in een programma zoekt. Deze routine werkt samen met vrijwel iedere VDU-routine en is geschreven m.b.v. SALFAA.

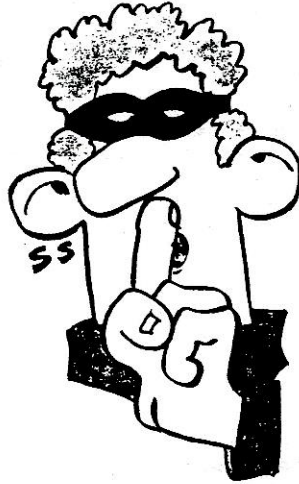
LISTING:

```
10 REM ESCAPE-MELDING
20
30 \ DIT PROGRAMMA GEEFT
40 \ DE MELDING "ESCAPE"
50 \ EN EVENTUEEL NOG
60 \ "AT LINE XXXX".
70
80 \ DIT PROGRAMMA WERKT
90 \ OOK SAMEN MET ANDERE
100 \ VDU-ROUTINES !!
110
120 \ ROLAND LEURS 041086
130
140 PASS 2 ; GOSUB a
150 PASS 1 ; GOSUB a
160
170 ASM-V ; ASM-U ; ASM-S
180 END
190
200aASM-BEGIN ASSEMBLEREN
210
220 :BASE = #3F00
230 :WERK = #3FFE
240 :RDCH = #020A
```

```

250
260 .OPTION #50
270 .LIST
280 .CODE BASE
290
300 :INIT
310 NOP
320 LDA RDCH ; STA WERK
330 LDA RDCH+1 ; STA WERK+1
340 LDA @(ESC)%256 ; STA RDCH
350 LDA @(ESC)/256 ; STA RDCH+1
360 RTS
370
380 :ESC
390 LDA #B001 ; CMP #DF
400 BEQ MELDING
410 JMP (WERK)
420
430 :MELDING
440 LDA #B001 ; CMP #DF
450 BEQ MELDING
460 JSR #F7D1
470 .BYTE #08,#20,#0A,#0D
480 .ASCII \ESCAPE \
490 LDA #01 ; BNE REGEL
500 JSR #F7D1
510 .BYTE #0A,#0A,#0D,#3E,#EA
520 JMP (WERK)
530
540 :REGEL
550 LDA @(TEKST)%256;STA #05
560 LDA @(TEKST)/256;STA #06
570 JMP #C2F2
580
590 :TEKST
600 .ASCII \P."AT LINE " !1&#FFFF';?1=0;?2=0;E.\, #0D
610
620 .EINDE ASSEMBLEREN
630
640 RETURN

```



Hoofdstuk3: ATOM ALS VERSTERKERTJE

Een oneliner die wel eens van pas kan komen om na te gaan of op een cassette computerprogramma's staan of een concert van Deep Purple is:

```
>P=T.;(LDA#B002;ROLA;ROLA;ROLA;STA#B002;JMPT.;)LI.T. <ret>
```

Als U nu de band laat lopen hoort U de pieptootjes of het concert door de luidspreker van het Atoompje.

Deze korte routine is misschien ook leuk om te verwerken in een load-routine, laat Uw fantasie en kennis er maar op los!

Hoofdstuk4: HET STATEMENT 'CHAIN' VOOR COS-GEBRUIKERS

Nu volgt een statement dat voorkomt ergens in de DOS of DISKrom maar dat natuurlijk ook voor COS gebruikers te maken is: CHAIN. CHAIN laadt een programma en RUNT het meteen. Zoiets als *RUN dus, maar hier is het executie adres niet van belang.

Syntax: CHAIN "prog" <ret>
LISTING:

```
10 PROGRAM CHAIN.STAT
20 REM ROLAND LEURS
30 REM MEI 1987
40 DIM RRO
50 PRINT $21
60 FOR I=5 TO 6
70 PRINT $I
80 P=A
90I :RRO
100 JSR #CEFA
110 DEY
120 STY #56
130 JSR #FFE0
210 JMP #AFAF
220J
230 NEXT I
240 $T="CHAIN"
250 T=T+LEN(T)
260 ?T=RR0/256:#80
270 T?1=RR0%256
280 T?2=#80
290 T=T+3
300 END
```

LISTING 4

Hoofdstuk5: HET PROGRAMMA VAN DE EEUW (????)

Het volgende programma kan reuze handig zijn bij het intikken van lekker lange programma's zoals bijv. 'TYPE EN TALK' in Acorn Nieuw nr 1 Jaargang 4 blz 82. Dat is een programma dat lastig is in te tikken zonder fouten. Als U eerst deze CHECKER aanlinkt krijgt U na een return een checksum van de ingetikte regel. Als nu iedereen zijn lange ingestuurde copy met checker laat afdrukken wordt intypen gewoon kinderspel.....

Het programma was oorspronkelijk geschreven voor een BBC en/of Electron, maar was dankzij de fantastische programmeertechniek van de echte schrijver, A.M. Rutkin, gemakkelijk aan te passen voor onze Atom, zowel in Atom-mode als in BBC-mode.

Ik heb dit programma op de Electron geassembleerd, toen in de Atom geladen en met de sourcemaker van A.Marshall een source gemaakt en daarna lichtelijk aangepast. Om het programma in BBC-mode te gebruiken moet regel 70 als volgt aangepast worden: LL2=#20E;LL3=#20F.

Een ander belangrijk gegeven is LL6, dit is werkruimte voor de routine en verder worden zeropage adressen #80 t/m #83 gebruikt. Als het programma ingetikkt is kunt U het controleren door LIST te tikken en de checksum van iedere regel te vergelijken met de checksum in de listing.

Ik hoop dat ik de club en haar leden met dit fantastische programma een groot plezier doe.

LISTING van het programma van de eeuw:

```
10REM CHECKER (BE94)
20J=23;DIMLLJ;F.I=0TOJ;LLI=FFF;N.;DIMA4;@=0 (ACF4)
30P.$12" checker" (61F0)
40DO PRINT$198;UNTIL COUNT=31 (8094)
50IN." "GEEF STARTADRES BV #3C00 "Z;IF?18=Z/256;G.50 (AD3C)
```

```

60IN."PRINTER J/N"*A (0990)
70LL1=#103;LL2=#208;LL3=#209;LL4=#230;LL5=#231 (9DAF)
80LL6=#9800;LL23=Z+#E9 (D139)
90P.#21;F.I=1T02;P=Z;IFI=2;P.#6;IF?A=74P.#2 (6059)
100I (A11E)
110;:LL0;:LDA@#01;EORLL23;STALL23;BEQLL7;LDALL2 (2D3A)
120STALL4;LDALL3;STALL5;LDA@#30;STALL2 (2CD3)
130LDA@Z/256;STALL3;JMPLL22 (1D94)
140;:LL7;:LDALL4;STALL2;LDALL5;STALL3;RTS (47BE)
150PHA;TXA;PHA;TYA;PHA (844B)
160TSX;LDALL1,X;CMP@#0A;BEQLL12;CMP@#0D (9205)
170BNELL8;JSRLL22;JMPLL10 (9F19)
180;:LL8;:CMP@#7F;BNELL9;LDA#82;CMP@#02;BCCLL10 (6BFC)
190DEC#82;BNELL10 (62C9)
200;:LL9;:CMP@#20;BCCLL10;LDX#82;STALL6,X;INC#82 (BE38)
210;:LL10;:PLA;TAY;PLA;TAX;PLA (C63D)
220;:LL11;:JMP (LL4) (BEAA)
230;:LL12;:LDY#82;BEQLL10;LDY@#00 (65E5)
240;:LL13;:LDXLL6,Y;JSRLL14;INY;CPY#82;BNELL13 (8A4C)
250LDY#83;BEQLL10;LDA@#20;JSRLL11;LDA@#28 (AF35)
260JSRLL11;LDA#81;JSRLL19;LDA#80;JSRLL19 (890D)
270LDA@#29;JSRLL11;JMPLL10 (5FEC)
280;:LL14;:LDA#83;BNELL15;TXA;CMP@#3E;BEQLL18 (7187)
290CMP@#21;BCCLL18;INC#83 (1939)
300;:LL15;:TXA;LDX@#07;EOR#81;STA#81 (0D9B)
310;:LL16;:LDA#81;ROLA;BCCLL17;LDA#81;EOR@#10 (3154)
320STA#81;LDA#80;EOR@#21;STA#80 (C9BC)
330;:LL17;:ROL#80;ROL#81;DEX;BPLLL16 (ED7C)
340;:LL18;:RTS (782D)
350;:LL19;:PHA;LSRA;LSRA;LSRA;LSRA (9049)
360JSRLL20;PLA (086C)
370;:LL20;:AND@#0F;CMP@#0A;BCCLL21;ADCE#06 (3146)
380;:LL21;:ADCE#30;JMPLL11 (FB8D)
390;:LL22;:LDA@#00;STA#80;STA#81;STA#83;STA#82 (1995)
400RTS (66DD)
410J;N.;P."CODE VAN "&Z" TOT "&P"#3;E. (BC6E)

```

N.B.: Ik weet niet met welke VDU-routines CHECKER wel en niet werkt maar het gaat in ieder geval fout als U het AUTO-statement gebruikt ! Als U checker samen met het TEXT-statement gebruikt tikt U eerst TEXT <return> en dan linkt U de checker aan.

HOOFDSTUK 6 : SPACE ROBOTS , een spel voor GAGSROM

Het volgende programma is een spel dat ik lang geleden eens geschreven heb maar nooit het laten publiceren. De bedoeling is om aanvallende ruimtewezens in de vorm van een robot te vernietigen. Dit doet U door met het vizier te richten en te vuren. Als een robot beneden aankomt wordt de 'warning-line' steeds groter. Als de warning-line een geheel vormt hebt U verloren. Als U een moederschip raakt wordt de warning-line weer een unit kleiner. Klinkt simpel, is het ook maar het kan ook best leuk zijn.

Het programma loopt op een ATOM met gestapeld geheugen en is het maakt eens gebruik van de normaal o zo hinderlijke ruis van de grafische mode. Om het intikken makkelijk te maken is dit programma afgedrukt met checker. Veel plezier ermee!

LISTing van SPACE ROBOTS:

```

10 REM ***** (89EF)
20 REM *** SPACE ROBOTS *** (6551)
30 REM ***** (2182)
40 REM * GAME WRITTEN BY * (FFD8)
50 REM * ROLAND LEURS * (B225)
60 REM ***** (950C)
70 REM (2331)
80 W=0;?#E1=W (E7D7)
90 GAGS;BASE #04;P." SPACE ROBOT ; RL'87" (BD3E)
100 DIM RR(30),NN(0) (F3C7)
110 REM ***sprites*** (7B4A)
120 CREATE POTEN,248,248,248,248,248,248,248,15,15,15,15,
15,15,15,15 (34DC)
130 FOR C=31 TO 37 (C9F9)
140 CREATE LASER,127,127,127,63,63,63,111,199,254,254,254,252,
252,252,246,199/A:C;NEXT (D2E7)
150 FOR C=1 TO 20 (3733)
160 CREATE/V ROBOT,195,36,24,189,255,126,126,126,36,60,36,0,0,
0,0,0/A:C;NEXT (D47C)
170 FOR C=20 TO 30 (EBF2)
180 CREATE MOEDER,0,15,15,240,240,15,15,0,0,240,240,15,15,240,
240,0/A:C;NEXT (330F)
190 CREATE VIZIER,253,129,129,63,63,129,129,253,191,129,129,
252,252,129,129,191 (DCDD)
200 CREATE ZERO,#38,#C6,#C6,#C6,#C6,#C6,#C6,#38,0,0,0,0,0,0,
0,0/A:48 (EA66)
210 CREATE ONE,#3C,#18,#18,#18,#18,#38,#18,#08,0,0,0,0,0,0,0,
0/A:49 (5F4D)
220 CREATE TWO,#FE,#62,#30,#18,#0C,#86,#C6,#7C,0,0,0,0,0,0,0,
0/A:50 (727A)
230 CREATE THREE,#7C,#C6,#8C,#18,#30,#18,#C,#FE,0,0,0,0,0,0,0,
0,0/A:51 (171E)
240 CREATE FOUR,#06,#06,#06,#FF,#C6,#66,#36,#E1,0,0,0,0,0,0,0,
0,0/A:52 (D5E5)
250 CREATE FIVE,#7C,#C6,#86,#06,#FC,#C0,#C0,#FE,0,0,0,0,0,0,0,
0,0/A:53 (A4B8)
260 CREATE SIX,#7E,#C3,#C3,#C3,#FE,#C0,#C0,#7E,0,0,0,0,0,0,0,
0/A:54 (D8BB)
270 CREATE SEVEN,#18,#18,#18,#18,#0C,#06,#83,#FF,0,0,0,0,0,0,0,
0,0/A:55 (B209)
280 CREATE EIGHT,#7E,#C3,#C3,#C3,#7E,#C3,#C3,#7E,0,0,0,0,0,0,0,
0,0/A:56 (A2B3)
290 CREATE NINE,#7E,#C3,#83,#03,#7F,#C3,#C3,#7E,0,0,0,0,0,0,0,
0,0/A:57 (5CBE)
300 CREATE HI,#C3,#C3,#C3,#C3,#FF,#C3,#C3,#C3,#7E,#18,#18,#18,
#18,#18,#18,#7E (1383)
310 CREATE SC,#7E,#C3,#83,#3E,#7C,#C1,#C3,#7E,#3E,#63,#61,#60,
#60,#61,#63,#3E (A576)
320 CREATE OR,#7C,#C6,#C6,#C6,#C6,#C6,#7C,#C3,#C6,#CC,#DB,
#FE,#C3,#C3,#FE (BDC6)
330 CREATE E,#FF,#C1,#C0,#C0,#FC,#C0,#C1,#FF,0,#18,#18,0,0,
#18,#18,0 (5649)
340 GOSUB1 (2947)
350 REM ***beeldopbouw*** (DCE9)
360 CLEAR4;CUBE1,0,1,168,253,19 (9DFA)
370 IMAGE:SC,7,181;IMAGE:OR,24,181;IMAGE:E,41,181 (116C)

```

```

380 IMAGE:HI,113,181;IMAGE:SC,129,181;IMAGE:OR,146,181 (E9F6)
390 IMAGE:E,163,181 (7EB6)
400 Y=30;FOR X=20 TO 250 STEP 42;Y=Y+1 (748D)
410 IMAGE:POTEN,X,20;SET Y,X,28 (7617)
420 NEXT X;V=W;T=180;GOSUB930;V=0;GOSUBs (3B62)
430 FILL #9700 #9800 #FF (5CDB)
440 REM ***start game*** (95DB)
450 V=0;A=5;D=1;P=11;L=27;SET:VIZIER,120,92 (0FC5)
460 FOR C=0 TO 30;RR(C)=0;NEXT C;B=0;IF V>750;D=10 (C71C)
470 DO FOR C=D TO (D+A);IF RR(C)=0;IF RND%5=0;RR(C)=1;SETC,
  ((ABSRND%10+1)*24),170;B=B+1 (BDDE)
480 IF RR(C)=1;SHOVEC,0,-P;POS C,X,Y;IF Y<26;GOSUBa;
  B=B-1 (926E)
490 GOSUB j (02DA)
500 WAIT;NEXT C;UNTIL B=0 (0BA2)
510 UNSET:VIZIER;PAUSE 120 (9E32)
520 SET:VIZIER,128,96 (B5D0)
530 FOR I=1 TO 50 (BDDF)
540 SOUND (ABSRND%255),10;GOSUB j (0CBE)
550 NEXT I;A=A+2 (7BAB)
560 IF A>16 THEN P=P+3;A=6 (60DB)
570 MOVE 255,20;PLOT7,255,192 (9C4B)
580 GOTO 460 (BC7C)
590 REM ***joystick*** (1E89)
600jF=0;G=0;H=0;JOYSTK F,G,H;POS:VIZIER,X,Y (B1BE)
610 IF F=1 AND X<232;SHOVE:VIZIER,8,0 (F431)
620 IF F=255 AND X>8;SHOVE:VIZIER,-8,0 (89A9)
630 IF G=1 AND Y<156;SHOVE:VIZIER,0,8 (CABB)
640 IF G=255 AND Y>44;SHOVE:VIZIER,0,-8 (061D)
650 IF H=0 THEN RETURN (E107)
660 GOSUB b;FOR H=1 TO 30;IF RR(H);ATHIT H,:VIZIER;
  GOSUB c (314B)
670 NEXT H;RETURN (4182)
680 REM ***robot beneden*** (B7C6)
690aUNSET C (32DA)
700 IF C>20 GOSUB 710 (6DF3)
710 BLOCK 0,L,2,5,4 (6072)
720 BLOCK 0,(244-L),2,5,4;MODE7 (D76E)
730 L=L+5;FORX=1TO50;LINKNNO;NEXT (2C5B)
740 PIXEL (L+1),4,X;IF X=0 THEN GOTO e (A53B)
750 MODE8;RETURN (F0D2)
760 REM ***vuur*** (83FA)
770bPOS:VIZIER,X,Y;X=X+8;Y=Y-4 (E92E)
780 POS(ABSRND%6+31),F,G;F=F+8;G=G-4 (C453)
790 FOR H=1 TO 2;MOVE F,G;PLOT 6,X,Y;NEXT;LINKNNO (CFBF)
800 FORZ=1 TO 2;FORH=1 TO 15STEP3;CIRCLE2,X,Y,H;NEXT;NEXT;
  RETURN (1A79)
810 REM ***raak*** (05F2)
820cIF H>20 THEN V=V+5 (64AB)
830 V=V+5;GOSUB s (D264)
840 UNSET H;RR(H)=2;B=B-1 (13C9)
850 FORJ=1TO10;SOUND50,10;NEXT;SOUND 255,20 (8367)
860 IF H<21 THEN RETURN (8134)
870 IF L=27 THEN RETURN (59B7)
880 L=L-5;BLOCK 1,L,2,5,4 (DE0B)
890 BLOCK 1,(244-L),2,5,4 (E4EA)
900 RETURN (DFCF)
910 REM ***zet score*** (C096)

```



```

920=BLOCK0,55,173,50,10;T=57 (761F)
930 F=((VZ10000)/1000+1)+47 (9B90)
940 G=((VZ1000)/100+1)+47 (C853)
950 X=((VZ100)/10+1)+47 (6BB6)
960 Y=(VZ10+1)+47 (390E)
970 IMAGE F,(T+10),181 (D202)
980 IMAGE G,(T+20),181 (9EA6)
990 IMAGE X,(T+30),181 (636C)
1000 IMAGE Y,(T+40),181 (0BDE)
1010 RETURN (7EB3)
1020 REM ***machinetaal*** (0BD7)
1030i PRINT $21;P=#9800;ILDY@0 (6755)
1040 :NNO TAX;ANDE$;STA #B002;TXA;ANDE#60;ADC#20;ASLA;
ASLA (FE93)
1050 ROL #80;TXA;ROL A;INY;CPY@#FF;BNE NNO;RTS;JPRINT#6 (CD32)
1060 REM ***instructies*** (4FD4)
1070 GRMOD;PRINT"***** space * robots *****";
?#E1=0 (D82B)
1080 PRINT"GEBRUIK JOYSTICK OM HET VIZIER" (12F9)
1090 PRINT"TE BESTUREN - VERNIETIG IEDERE" (5EE9)
1100 PRINT"SPACE ROBOT' EN ELK MOEDERSCHIP" (AAC3)
1110 PRINT"-- GEBRUIK 'FIRE' OM TE SCHIETEN" (42D0)
1120 PRINT"===== SCORE =====" (7762)
1130 PRINT" " SPACE ROBOT : 5 PUNTEN " (CF79)
1140 PRINT" " MOEDERSCHIP : 10 PUNTEN " (BAC0)
1150 PRINT""HET SPEL EINDIGT ALS DE warning-line GESLOTEN IS."
(C5E0)
1160 PRINT"<DRUK TOETS>";INV (D329)
1170 SET:ROBOT,30,85;SET:MOEDER,28,55 (113E)
1180 DO SOUND (ABSRNDZ255+1),5;KEY X;UNTILX<>0;RETURN (7C50)
1190 REM ***einde game*** (F9E5)
1200e FORX=50TO1 STEP-1;MODE7;LINKNNO (313B)
1210 MODE8;PAUSEX;NEXT;MODE7;FORX=1TO255;LINKNNO;NEXT;
MODE8 (1BEE)
1220 FOR C=31 TO 36;POS C,X,Y (A1AE)
1230 UNSET C;X=X+8;Y=Y-4;FORH=1TO2;FORG=1 TO 10 STEP2 (8270)
1240 ?#B002=?#B002:4;CIRCLE2,X,Y,6;NEXTG;NEXTH;LINKNNO;
LINKNNO;NEXTC (FDS0)
1250 FOR C=#8320 TO #97FF STEP4 (4704)
1260 !C=0;NEXT C (9C9A)
1270 PRINT $30'""*****" (2532)
1280 PRINT"***** SPACE * ROBOTS *****" (101F)
1290 PRINT"*****" (9751)
1300 PRINT" ** DRUK OP EEN TOETS **" (07DF)
1310 LINK #FFE3;?#13=0;?#14=0;?#15=0;REM CLEAR STACKS (7C19)
1320 IF V>W THEN W=V (4E60)
1330 GOTO 360 (FE95)

```

HOOFDSTUK 7: INTIKKEN & RUNNEN MAAR : schaduwbyte.

De volgende routine loopt onder interrupt en dus moeten een VIA en link2 aanwezig zijn. Na aanroepen krijgt U linksboven in beeld het nummer van de voorgeschakelde eprom. Indien deze eprom gelocked is, is het schaduwbyte geïnverteerd. De routine wordt aangeroepen met het IRQ-statement (zie eventueel BGE nr.1 van 1986 bladzijde 17) maar U kunt natuurlijk ook op een normale manier de interrupt activeren.

LISTing schaduwbyte-routine

```
100 REM SCHADUWBYTE
110 INPUT "ASSEMBLEREN NAAR "A;P=A
120 I
130 LDA #BFFF
140 AND @#OF
150 CLC
160 ADC @#30
170 STA #8000
180 LDA #BFFF
190 AND @#40
200 BNE P+3
210 RTS
220 LDA #8000
230 CLC
240 ADC @#80
250 STA #8000
260 RTS
270 J
280 IRQ A,1
290 END
```



HOOFDSTUK 8: AFDELING VERKOOP

In mijn prive-verkoop heb ik nog voor U liggen:
<1> een 16 K-kaart van de club, echter zonder RAM's.
<2> een complete ATOM met ongeveer 17 Kbytes RAM en 30 Kbytes ROM + literatuur en software.
<3> binnenkort leverbaar: de ED64-B gerelocceerd naar #1000 en verder voor mensen die een tekstverwerker om een of andere reden niet op #A000 plaatsen. U wordt natuurlijk op de hoogte gehouden!

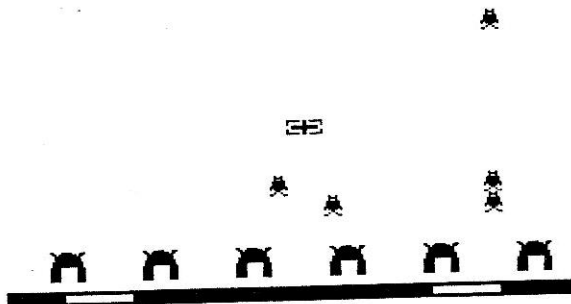
HOOFDSTUK 9: SLOT

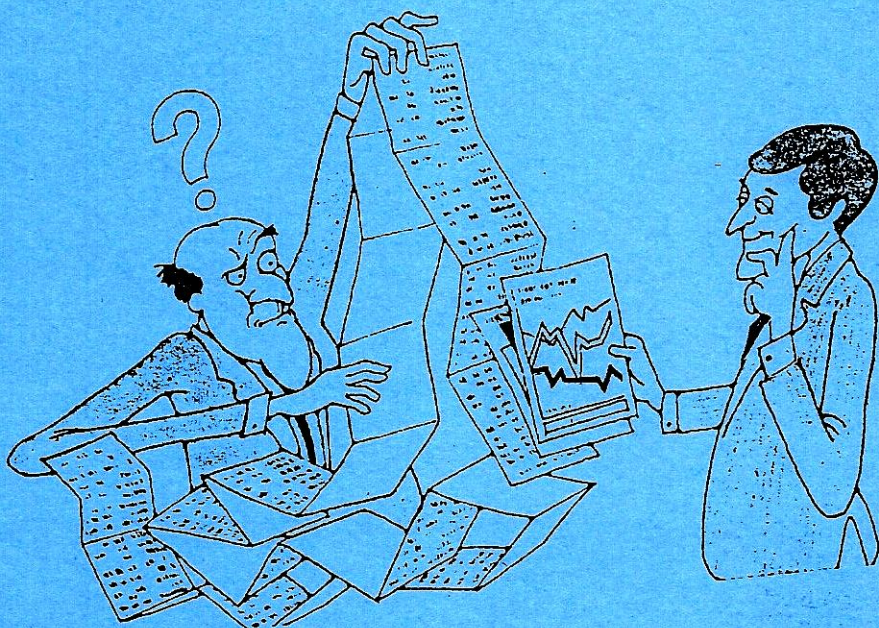
Zo, mijn verhaaltje is nu af. Ik hoop dat U er iets mee kunt doen, er was immers genoeg afwisseling.
Tot een volgende keer; met de vriendelijke groeten van Roland.

N.B. FOUTEN ZOWEL IN TEKST ALS IN PROGRAMMA'S VOORBEHOUDEN!

SPACE ROBOTS

SCORE: 0000 HISCORE: 0000





Sorry meneer de voorzitter, maar dit is de dalende statistiek die over de ontvangst van copy gaat.