

1e jaargang / december 1985

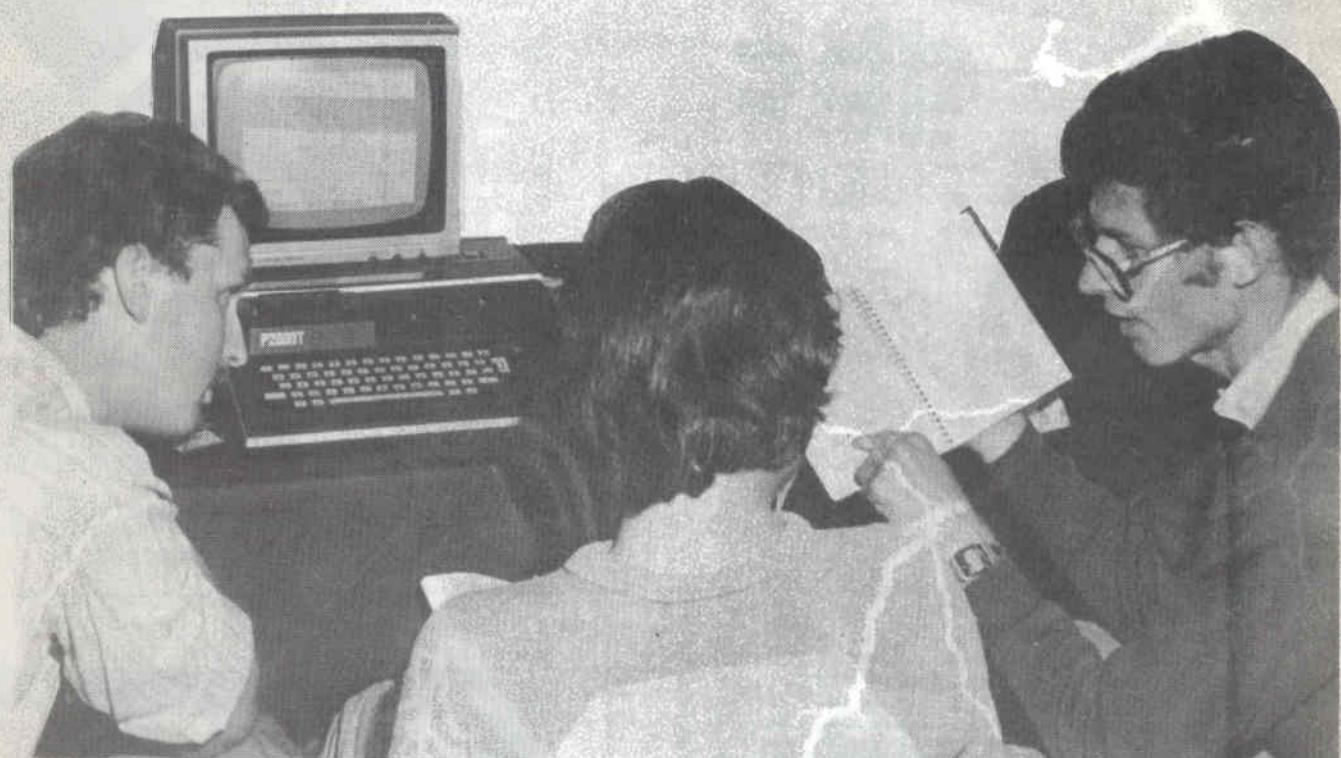
losse nummers f 5,85

INFO

nummer

6

BANK SWITCHING SPECIAL



In dit nummer o.a.

- Bankswitchingspecial • Ram Dos • Boekbespreking
- Afdelingsberichten • Ondernemingsspel • Afdelingsadressen

Officieel Contactorgaan van de Stichting Gebruikersgroep P Computers

Uitgever: GPC
Redactieadres: Postbus 3000
 2260 DA LEIDSCHENDAM

Hoofdredakteur: Erik Alers
Eindredakteur: Albert C. Veldhuis

Produktie: Peter Pit

Adviezen: Dick Pronk

Advertenties: Postbus 3000
 2260 DA LEIDSCHENDAM

Advertisertarief: op aanvraag

Abonnementen: Deelnemers aan de GPC krijgen het blad gratis
 Deelname voor scholen en bedrijven f 90,00
 met gratis toeozending van een nummer; elk extra
 abonnement f 25,00
 Losse nummers f 5,85

Copyright: De inhoud van dit blad mag niet gereproduceerd worden in welke vorm dan ook, zonder voorafgaande schriftelijke toestemming van de uitgever.
 De aansprakelijkheid uit hoofde van auteursrechten van ingezonden copy ligt bij de inzender.

De Stichting Gebruikersgroep P-Computers stelt zich ten doel, het gebruik van Philips-computers in de ruimste zin te bevorderen.

Deelname aan de Stichting wordt aangegeven voor tenminste 1 kalenderjaar en geldt tot schriftelijke opzegging.

Het deelnemerschap bedraagt f 45,00 per jaar, voor bedrijven en instellingen f 90,00.

Opgave voor het deelnemerschap aan het secretariaat van de GPC:
 postbus 6059,
 5960 AB HORST

Bereikbaar maandag t/m vrijdag van 9.00-12.30 uur en van 14.00-21.00 uur, onder telefoonnummer (04709) 4904

Betaalwijze: Binnen 14 dagen na aanmelding aan:
 ABN Katwijk rek.nr. 56.73.05.082 of
 via gironummer 240800 t.n.v. Stichting GPC
 te Katwijk

Adreswijzigingen: A.u.b. schriftelijk 6 weken van te voren opgeven aan het secretariaat van de GPC.

Redaktioneel

Helaas moeten we toegeven dat de verschijning van TRON 6 lang op zich heeft laten wachten. Maar we zullen er het komende jaar met nog meer energie tegen gaan en onze uiterste best doen de 6 nummers zo tijdig mogelijk te laten verschijnen.

Wij rekenen echter wel op uw aller medewerking. En daartoe vragen wij in de eerste plaats dringend de medewerking van programmamakers die LISTINGS ter beschikking willen stellen. De programma's hoeven niet geheel foutloos te zijn en het is zeker niet de bedoeling dat er listings met 65534 regelnummers worden ingezonden (Bij BASIC).

Dok de machinetalers kunnen volop reageren. De Auteurs van de diverse verhalen vragen zelfs om uw reactie.

Helaas zullen liefhebbers van dergelijke programma's deze keer weinig van hun gading tegen komen, doch de machinetaal fanaten destee meer.

Een tweede oproep willen wij richten aan alle afdelingsbestuurders om er vooral voor te zorgen dat zij in elke uitgave van TRON verslag doen van de activiteiten en gebeurtenissen in hun afdeling! Dat is deze keer ook maar poertjes! En heeft uw afdeling een eigen logo zoals bijv. afd.Noord (zie TRON 5)? Stuurt ons daarvan dan een afdruk, dat maakt uw afdeling herkenbaar!

Tenslotte verzoeken wij de inzenders van kopij dit bij voorkeur te doen op een bandje en gebruik makend van het programma Minitext.

Stuurt snel uw kopij op opdat het eerstvolgende nummer spoedig kan verschijnen!

Nog even dit:

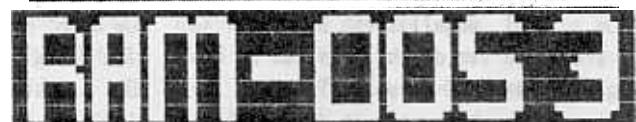
LEEST U VOORAL HET ARTIKEL VAN DE VOORZITTER WANT HIERIN WORDEN DE PLANNEN VOOR 1986 ONTVONDEN.

Albert Veldhuis ■

Inhoud

Pagina	4 Bestuursmededelingen
	5 Bankswitching Charles vd Linden
	14 Oproep aan programmeurs !!
	19 RamDos Jeroen Hoppenbrouwers
	20 Boekbespreking
	22 Van de afdelingen
	23 Ondernemingsspel
	24 Hoofdnoot
	Hoofdnoot oplossing
	Teleaccursus Personal Computer I
	25 Onderwijswerkgroep 'Probleemloos printen met de Juki 6100'

LAATSTE NIEUWS LAATSTE NIEUWS LAATSTE



Jeroen en Thijs Hoppenbrouwers hebben het programma 2 verbeterd. Het programma RamDos 3 is thans te verkrijgen via:

GPC regio-Zuid Postbus 551 Roermond	telefonisch via Jan Drohm tel: 04750-19389
---	---

Het programma is alleen geschikt voor een P2000 met 64K geheugenuitbreidingsprint.

LAATSTE NIEUWS LAATSTE NIEUWS LAATSTE

BESTUURSMEDEDELINGEN

Met ingang van 1 januari 1986 is een nieuw landelijk bestuur van de GPC aangetreden. Hoewel wel wat laat wil dit bestuur u en de uwen een gezond en fijn 1986 wensen.

De eerste vergadering van het nieuwe bestuur is onlangs gehouden. In het onderstaande geef ik de belangrijkste punten kort weer:

Taakverdeling binnen het bestuur:

Voorzitter	:	Herman Hietbrink
Sekretaris	:	Chris Noya
Penningmeester	:	Jan de Zeeuw
Regio-koordinator	:	Jan Droha
Hardware	:	Ronald Bijloo
Tron	:	Erik Alers
Ledenbestand	:	Cees Spruyt

Kontributie inning 1986

Na ampele overwegingen heeft het bestuur toch besloten over te gaan tot CENTRALE inning van kontributie. Voor wat betreft de hoogte van het bedrag zie het collofon. De bedragen zijn niet veranderd ten opzichte van 1985. Om een goede administratie van de betalingen te kunnen uitvoeren verzoekt de penningmeester om alleen via de bank op het banknummer te storten en via de giro alleen op het gironummer. DUS NIET KUISEN S.V.P.!

Sekretariaat

Het adres van het sekretariaat van de Landelijke Stichting GPC is:
POSTBUS 6059
5960 AB HORST

Inventarisatie van de regionale activiteiten:

Om zo snel mogelijk een overzicht te verkrijgen van de aktiviteiten op het gebied van HARD-ware, SOFT-ware, lezingen en kursussen in de verschillende klubs vragen wij de besturen hiervan om een vragenlijst in te vullen en zo snel mogelijk op te sturen naar het sekretariaat. Zo kan eenieder ook later zelf kennis nemen van wat zich zoal in den lande afspeelt.

Vergaderfrekwentie met de regio's

Wij stellen voor om minstens 3 maal

per jaar met de regio-besturen te vergaderen. De eerste bijeenkomst valt op 5 maart 1986.



Zoals u kunt zien hebben wij besloten om TRON 6 zo snel mogelijk te laten uitkomen. Wij hebben het stellige voornemen om in 1986 6 maal een TRON laten verschijnen.

Hardware service

Op hardware-, maar zeker ook op software-, gebied dienen zich voor onze klub spektakulaire mogelijkheden aan. Wij hebben besloten om een aparte rekening te openen, onder beheer van Ronald, waar de gelden voor de bestellingen direct heen gaan. Wij rekenen er zo op dat de service beter en sneller kan worden en dat de problemen uit het verleden vermeden zullen worden.

Regionale data-bases

Omdat zich nu een aantal interessante mogelijkheden voordoen zijn wij aan het onderzoeken of wij kunnen komen tot het oprichten van een aantal regionale data-bases van de GPC, die bijv. een maal per week de bestanden zodanig bijwerken dat elke GPC-base dezelfde informatie biedt. Ook tele-software sluiten we niet uit. Mocht u zelf geïnteresseerd zijn hieraan medewerking te verlenen, of iemand uit uw klub: de kontaktman is:

Jan Droha
Dr. Leurstraat 5
6041 KL Roermond
tel: 04750-19339
vidibus:400021509

Algemeen

Het nieuwe bestuur wil er, vooruit kijkend, lekker positief tegen aan. Wij hopen op uw steun te mogen rekenen! Indien er u iets dwars zit, neem dan zo snel mogelijk contact met ons op.

Het nieuwe bestuur heeft de volgende doelstellingen:

- * Gezond maken van de GPC als HOBBY-klub.
- * Zaken voor de deelnemers betaalbaar houden.
- * Een redelijke vergoeding geven aan diegenen die iets presteren voor de GPC.
- * Open en eerlijke communicatie.

Als bestuur zijn wij gemotiveerd hieraan heel hard te werken. Helpt u ons daarmee? Dan maken wij er SAMEN vast iets heel fijns van voor iedere P-computer gebruiker!

Kontributie betaling

DE NIEUWE NUMMERS :

Bank	A.B.N.-Katwijk, rekening nummer: 56.73.05.082
Giro	240800

Beide nummers t.n.v. STICHTING GPC - Katwijk

Betaal de verschuldigde bijdrage voor 1986 ZO SNEL MOGELIJK s.v.p.!

H.H.Hietbrink
voorzitter ■

Uw betaling wordt verwerkt door de penningmeester waarna u als deelnemer wordt genoteerd in de deelnemersadministratie. U ontvangt hierna van het landelijk secretariaat schriftelijke bevestiging. Indien uw gegevens onjuist staan vermeld kunt u uw bevestiging retourneren aan ledendienstadministratie die uw gegevens zal wijzigen. Wij verzoeken u slechts te betalen op het bankrekeningnummer indien u gebruik maakt van een bankrekeningnummer en per giro indien u gebruik maakt van een giorekening.

Bankswitching

Een artikel geschreven door Charles v.d. Linden

Charles

v.d. Linden

Charles van der Linden is al sinds februari 1985 bezig met te onderzoeken hoe de geheugenruimte in de P2000 P2000 met een 64 K uitbreidning gebruikt kan worden voor gegevensopslag. In eerste instantie werd deze breidning door MINIWARE verkocht zonder enige toelichting, waardoor praktisch alle kopers er niets mee aan konden vangen.

Later verscheen er een aanvullende gebruiksaanwijzing op het "zwarte" boekje. In de praktijk blijkt echter dat de eersten nog weinig plezier hiervan hebben door gebrek aan kennis van de beginselen van machinetaal.

Toen Charles zich met machinetaal ging bezighouden werd alles langzaamhand veel duidelijker en was hij in staat routines te maken om succesvol met de banken om te gaan. Het is hem nu mogelijk om van alles in de banken weg te bergen, naar cassette te schrijven, terug in te lezen en uit de banken weer op te vragen.

Zo kunnen banken een voor een van sette worden gevuld met meerdere basicprogramma's, arrays, strings en videobeelden.

Eenmaal ingeladen zijn ze in een fractie van een seconde binnen handbereik.

Om het onderwerp Bankswitching aan iedereen duidelijk te kunnen maken, zal hij hierbij eerst enige begrippen die belangrijk zijn bespreken.

Indien men namelijk geen kennis van een bepaald begrip heeft, zijn alle zaken die in het verlengde ervan liggen bij voorbaat onbegrijpelijk.

Misschien zal u Charles nog eens spreken op een volgende bijeenkomst en zal hij u de praktische toepassingen demonstreren.

Voor bankswitching zijn de volgende zaken van belang:

1. Begrip van de elementaire principes van de computer.
2. Vertrouwdheid met de geheugenindeling van de P2000.
3. Enig idee over de wijze van opslag van Basicregels, variabelen arrays en strings in het geheugen
4. Kennis betreffende de werking van het beeldscherm en het videogeheugen.
5. Op de hoogte zijn van de beginselementen van machinetaal.
6. Notie hebben van enige basismachinetaalroutines.

Waar het om ging was manieren te vinden en methodes te ontwikkelen die ervoor zorgen dat een blok gegevens naar en uit een bank kan worden getransporteerd.

Door middel van PEEK en POKE is het mogelijk het cijfer dat in een bepaalde geheugenplaats is opgeslagen te weten te komen en op een andere plaats vast te leggen.

Bijv. rbeeld:

Zet in plaats B de waarde van het getal dat zich in plaats A bevindt:
POKE B, PEEK (A)

Plaat in B de inhoud van (A).

Een reeks van gegevens (data) kan via een FOR-NEXT lus worden overgebracht. Stel C voor het aantal gegevens: FOR I=0 TO C-1

POKE B + I, PEEK (A + I)
NEXT I

Bij een grote hoeveelheid gegevens kan dit te lang duren. Vandaar dat voor zo'n groot blok getallen de toevlucht wordt genomen tot machinetaal, wat veel sneller is. Er behoeft n.l. geen Basic te worden vertaald.

Eerst zal ik de relatie van de computer met cijfers met u bespreken en daarna de geheugenindeling.

Om een basicprogramma, arrays, videobeelden of strings te verplaatsen zal men eerst het een en ander moeten weten over de Z-80 microprocessor. Ter afsluiting geef ik u de machinetaalroutines.

De computer kent slechts twee getallen n.l. 0 en 1.

Nul voor geen stroom, is er wel stroom dan betekent dat "1".

Op deze wijze worden cijfers omgezet in elektrische stroomstaotjes.

Bij nul wordt de stroom gestopt, bij "1" laat men de stroom doorgaan.

Volgens het Binaire stelsel kunnen nu getallen geformeerd worden.

Het tweetallig selsel gebruikt alleen 0 en 1.

Om een reeks getallen te maken, slaat men alle andere decimale getallen doodgewoon over. Die getallen bestaan dan gewoonweg niet.

BINAIR	DECIMAAL	HEXADECIMAAL
0	0	
	1	
10	2	
11	3	3
100	4	
101	5	
110	6	
111		
1000	8	
1001	9	
1010	10	
1011	11	B
1100	12	
1101	13	D
1110	14	
1111	15	
10000		

Hexad ciamaal is voortalli stelsel.

De ge allen tient voorg steld door ders A t/m F.

TOT DE MACHT	BIN	HEX
16	10000	1048576

0	-	16
4	1000	256
-	10000	4096
16	100000	65536
5	32	1048576
6	64	1048576
7	128	1048576
8	256	1048576

Een byte bestaat uit acht bits, het grootste getal dat is onder te brengen op een geheugenadres is 255. (255 is binair 1111111)

De Z-80 microprocessor binnen in de P2000 heeft een bereik van 64 Kilo-bytes. D.w.z. er zijn $64 \times 1024 = 65536$ (0-65535) geheugenplaatsen mogelijk.

65535 komt overeen met het 16 bits-getal 1111111111111111.

Elke plaats (byte) kan een getal van maximaal 255 bevatten. Wanneer we een getal groter dan 255 willen vastleggen, dan zijn er meer geheugenplaatsen nodig.

Een getal A%, b.v. 25927 wordt opgeslagen in twee bytes.

De tweede (hoge byte) wordt gevuld met het getal dat ontstaat door deling door 256 naar beneden afgerond. (integerdeling)

De eerste (lage) byte gaat het restant van de deling bevatten. (modulo)

$$\text{HB} = \text{A\% : } 256 \quad \text{**}$$

oftewel $25927 : 256$

$$\text{of HB} = \text{INT}(25927 / 256) = 101$$

$$\text{LB} = \text{A\% MOD } 256$$

oftewel $25927 \text{ MOD } 256$

$$\text{of LB} = 25927 - (256 * \text{HB}) = 71$$

Werken met hexadecimale getallen maakt alles eenvoudiger. Het getal 25927 is hexadecimaal 6547 n.l.

$$\text{HEX\$}(25927) = \&\text{H } 6547$$

$\&\text{H } 65$ is de hoge byte, $\&\text{H } 47$ de lage.

$$\text{LB} = \text{HEX\$}(71) = \&\text{H } 47$$

$$\text{HB} = \text{HEX\$}(101) = \&\text{H } 65$$

Een hexadecimaal getal wordt dus omgekeerd gesplitst.
 $\&\text{H}625C$ wordt $\&\text{H } 5C$ en $\&\text{H } 62$.

Een groot getal weer terug vormen uit twee bytes gaat als volgt;

$$\begin{aligned} \text{A} &= \text{inhoud lage byte} \\ &+ 256 * \text{inhoud hoge byte of} \end{aligned}$$

$$\begin{aligned} \text{A} &= \text{PEEK (LB)} \\ &+ 256 * \text{PEEK (HB)} \end{aligned}$$

Bijvoorbeeld:

$$\begin{aligned} \text{A} &= \text{PEEK } (\&\text{H } 6405) \\ &+ 256 * \text{PEEK } (\&\text{H } 6406) \end{aligned}$$

Itron

Bankswitchingspecial

&H dec.

- 0000 - 0- MONITOR in ROM (Read Only Memory)
Uit dit geheugen kan alleen worden gelezen en niets worden veranderd. Hierin bevinden zich routines voor Cassette, Disk en beeldscherm (Video).
 - 1000 - 4096- BASIC-insteekmodule in sleuf 1.
De Basic-Interpreter maakt het mogelijk om te programmeren in BASIC. Het vertaalt de codewoorden van de Basic-instructisch in een aantal handelingen die in machinetaal worden uitgevoerd. (ROM)
 - 5000 -20480- Het VIDEOgeheugen.
Deze bevat het linker en rechter beeldscherm.
24 regels en 80 kolommen, links en rechts 40 kolommen.
Met een 40/80 karakterkaart worden alle kolommen op het scherm weergegeven.
 - 6000 -24576- In gebruik door de Monitor voor registerruimten en bufferblokken. (tijdelijke opslag van gegevens)
 - 6200 -25088- Administratie en bufferruimte voor Philips Cassette Basic.
 - 6547 -25927- Basic-ruimte.
Het begin is uit te lezen op $\&\text{H } 625C$ en $\&\text{H } 625D$. Ze is gevuld met regel-nummers, regel-aanwijzers, BASIC-tokens en ASCII-waardes.
 - - - Variabelenruimte.
Het begin van de variabelenruimte is uit te lezen op $\&\text{H } 6405$ en $\&\text{H } 6406$. Hierin bevinden zich integer, enkel en dubbele precisie variabelen en stringvariabelen.
 - - - Array ruimte.
Het begin is uit te lezen op $\&\text{H } 6407$ en $\&\text{H } 6408$. Ruimte voor arrays is te reserveren d.m.v. DIMmen.
 - - - Vrije ruimte.
Het begin is uit te lezen op $\&\text{H } 6409$ en $\&\text{H } 640A$. Deze ruimte is afhankelijk van de grootte van andere ruimten.
 - - - STACK = stapel.
Het einde is uit te lezen op $\&\text{H } 64FB$ en $\&\text{H } 64F9$. Hierin worden o.a. de terugspringadressen van GOSUB en FOR-NEXT bewaard.
 - - - String ruimte.
Normaal 50 bytes. Ze is in te stellen met CLEAR, bijvoorbeeld CLEAR 4000.
 - - - Gereserveerd.
Het begin (-1) is uit te lezen op $\&\text{H } 63B8$ en $\&\text{H } 63B9$. Ze is in te stellen met bijvoorbeeld CLEAR 50, $\&\text{H } DFFF$ waarbij de DFFF hier de grens is.
- FFFF -65535- EINDE GEHEUGEN

** = ':' staat voor integerdeling

Een Basic programma begint op de plaats die wordt aangeduid door de adressen &H 625C en &H 625D. Normaal staan hier de waarden &H 47 en &H 65. U weet nog wel hoe we die geheugenplaatsen uitlezen.

```
?PEEK(&H625C)+256*PEEK(&H625D)[ENTER]
25927
Ok
?HEX$(25927)
6547
Ok
```

Het einde van een programma is te vinden met dezelfde formule in de geheugenplaatsen **WH 6405** en **WH 6406**.

De lengte van een programma is dus te berekenen door van het eindadres het beginadres af te trekken.

D.m.v. blockmove naar het bestemmingsadres &H E002 kan zo het hele basicprogramma in een bank worden geparkeerd. De lengte leg ik vast in &H E000 en &H E001.

Op deze wijze zijn alle banken te vullen met verschillende programma's, mits de lengte niet meer dan 8 KB bedraagt. (8×1024 bytes = 8192 minus 2 bytes voor de adressen waarin de lengte staat vermeld)

Terug uit de bank naar de basic-ruimte is zo ook eenvoudig via blockmove te realiseren, mits het eindadres van het programma op de plaatsen &H 6405 en &H 6406 eerst wordt afgebakend.

Een volgende fase is de programma's in de banken achter elkaar te laden, om zodoende de ruimten die leeg gebleven zijn bij programma's kleiner dan 8 K, toch te benutten.

Nog een stap verder is het om het hele geheugen te vullen met programma's. Hiernaast ziet u 12 programma's over het geheugen verdeeld, totaal samen 76 K. Twee bijna volle cassettekanten kunnen achter elkaar in het geheugen worden geschreven.

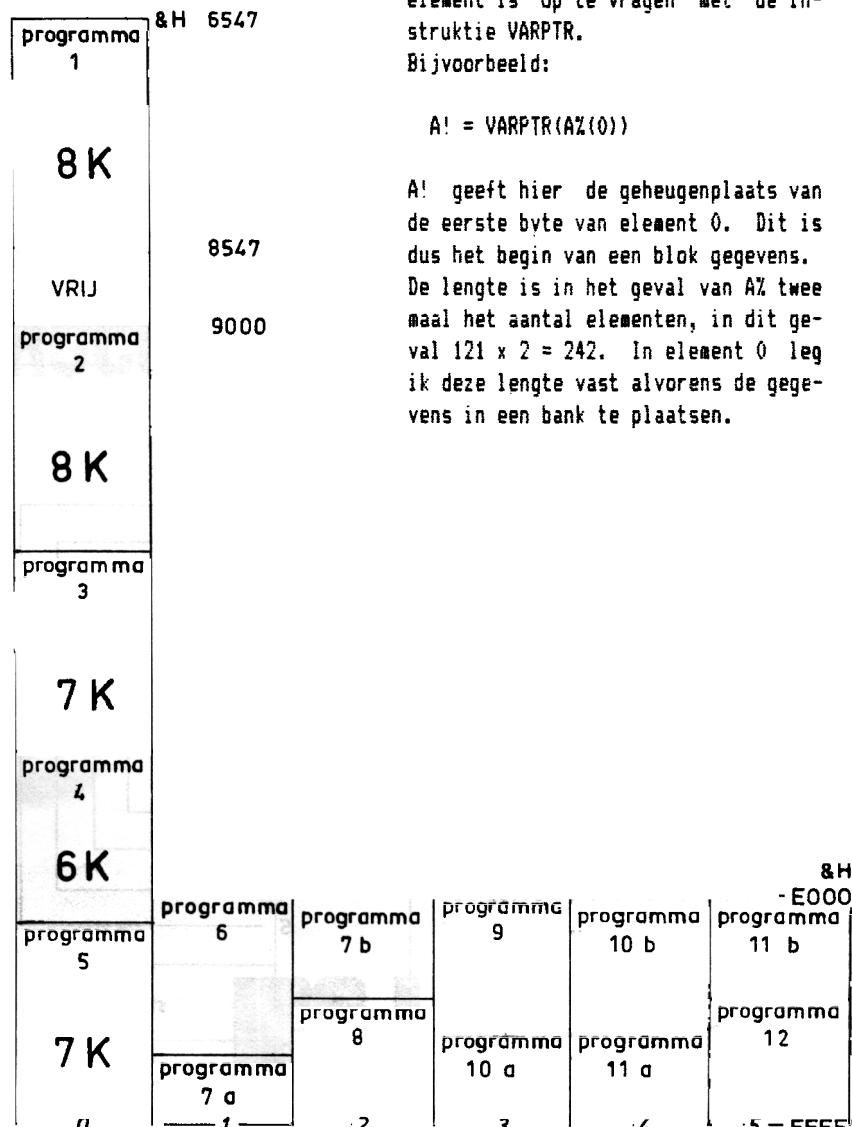
Programma 1 bevindt zich reeds in de eigenlijke Basic-ruimte.
Programma 1 en 2 kunnen onderling verwisseld worden.

Wanneer andere programma's naar de basicruimte worden gekopieerd sneuvelt het reeds erin staande programma.

Een dergelijke methode opent ook wel perspectief voor toepassingen in een 48 K, 32 K en zelfs een 16 K geheugen. Evenzo kunnen daar een aantal, weliswaar minder, kleinere programma's tegelijk in het geheugen worden opgeslagen en op afroep beschikbaar zijn.

In dit kader is het ook mogelijk om diverse programma's achter elkaar in de basic-ruimte te zetten. Opvragen van een programma kan dan door de wijzer van het begin van de basic-ruimte n.l. de plaatsen &H 625C en &H 625D naar het bewuste programma te laten wijzen.

Zo zijn er wel meer mogelijkheden te bedenken als je er voor gaat zitten.



Een array is een reeks variabelen met dezelfde variabele-naam. Voordat een array aangemaakt wordt moet in het geheugen een ruimte gereserveerd worden, groot genoeg om de gegevens en de elementen van het array te bevatten. Dit gebeurt d.m.v. DIMMen.

Bijvoorbeeld:

DIM AZ(120)

Er zijn nu 121 elementen aangemaakt van het array A% geteld van 0 t/m 120. Bij het opstarten van de P2000 wordt er indien een array niet tevoren werd geDIM'd door de BASIC-INTERPRETER van uitgegaan dat men hooguit 11 elementen zal gebruiken. Als men dan een arrayelement gebruikt dat een grotere index dan 10 heeft dan volgt een foutmelding.

De plaats in de arrayruimte van een element is op te vragen met de instructie VARPTR.

Bijvoorbeeld:

A1 = VARPTR(AZ(0))

A! geeft hier de geheugenplaats van de eerste byte van element 0. Dit is dus het begin van een blok gegevens. De lengte is in het geval van A% twee maal het aantal elementen, in dit geval $121 \times 2 = 242$. In element 0 leg ik deze lengte vast alvorens de gegevens in een bank te plaatsen.

M.b.v. blockmove worden deze 242 bytes in een klap in een bank gebracht. Vanuit de bank kan het array naar cassette worden geschreven en later weer terug van cassette naar een bank.

Meerdere arrays kunnen zodoende opgeslagen worden in afwachting van gebruik, terwijl er maar voor een arrayruimte gereserveerd hoeft te worden.

Dok is het mogelijk arraygegevens rechtstreeks uit de nk te halen, zodat de eigenlijke rayruimte vrij blijft.

BASIC
VARIA-
BELEN

Een stringarray is een verhaal apart.
Ze wordt als volgt opgeslagen:
In de arravruimte wordt een ruimte
gereserveerd d.m.v. het statement
DIM. Elk element heeft drie bytes no-
dig. Deze bytes heten de stringdes-
criptoren.(aanwijzers)
Net als bij variabele-arrays zijn de
elementen te vinden met de opdracht
VARPTR.

A! = VARPTR(A\$(0)) wijst naar de eerste byte van A\$(0). Deze eerste byte geeft de lengte van de string aan. Vandaar dat een string niet meer dan 256 tekens lang mag worden. Een groter getal dan 255 past nu eenmaal niet op een geheugenplaats. Byte 2 en 3 wijzen naar de plaats van de string in de stringruimte. U weet natuurlijk hoe we dit uitlezen.

Terug van cassette gaat als volgt:

- 1 Van cassette in een bank laden.
- 2 Voor het aantal strings een ruimte reserveren in de arrayruimte.
- 3 De stringdescriptoren via blockmove van achter de strings in de bank naar de geDIMde arrayruimte brengen.

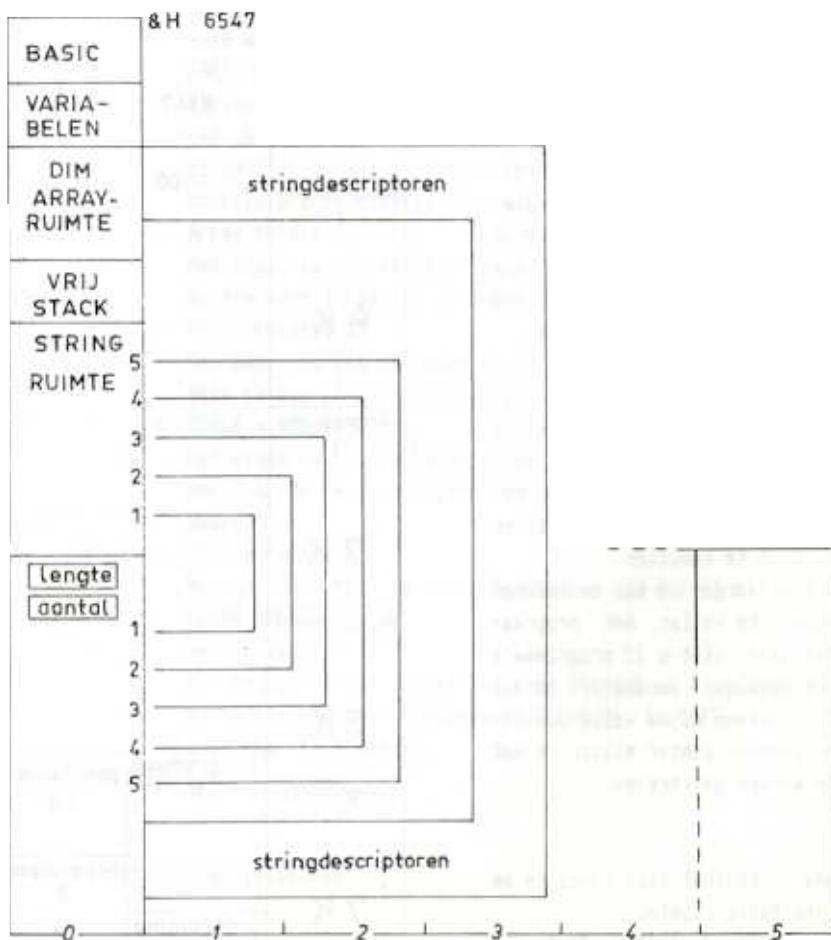
B! = PEEK (A!+1) + 256 * PEEK (A!+2)

Via `blockmove` kunnen we de strings een voor een vanuit de stringruimte in een bank kopiëren. Telkens moeten we de stringdescriptors wijzigen, zodanig dat deze string zijn nieuwe plaats in de bank gaat aanwijzen.

Nadat alle strings in de banken zijn overgeheveld wordt het blok descriptoren uit de arrayruimte achter de stringarray in de bank geplaatst. Voorin de bank wordt de totale lengte en het aantal strings vastgelegd. Alles is nu klaar om naar cassette te schrijven.

De strings worden nu uit de bank gelezen en zodoende wordt de normale stringruimte gespaard.

Wanneer de strings worden verbeterd, komen de veranderde strings (met dezelfde elementnummers) in de stringruimte te zitten. Door elke string apart weer uit te lezen en in een andere bank op te slaan, kunnen de verbeterde versies uit die bank weer naar cassette worden geschreven.



ARRAY byte

TYPE	1	2	1
NAAM	1	90	2
	1	65	3
LENTE	1	205	4
vanaf hier	1	0	5
DIMENSIE	1	1	6
Aantal elementen	1	101	7
	1	0	8
Element 0	1	12	9
	1	23	10
Element	1	45	11
	1	19	12
Element 2	1	127	13
	1	33	14
enzoorts			

Integer array A% gediimensioneerd op 100 d.m.v. de opdracht DIM A%(100)

Links ziet u hoe een integer array in het geheugen wordt opgeslagen. U ziet dat vanaf de negende byte de gegevens worden opgeslagen. In byte 1 t/m 8 worden enkele administratieve gegevens opgeslagen. Dat we hier met een integer array te maken hebben zien we aan de eerste byte. We passen hierbij het volgende tabelletje toe:

2 = integer array
3 = string array
4 = enkele precisie
8 = dubbele precisie

Waarom niet oplopend van t/m 4?

Dit zal alles te maken hebben met het aantal geheugenplaatsen dat per element in gebruik wordt genomen. We hebben al gezien dat bij een integer array per element twee geheugenplaatsen in beslag werden genomen en bij stringarrays drie stuks. Inderdaad, bij enkele precisie zijn we er per element vier en bij dubbele precisie zelfs acht per element kwijt. Dit is dan ook een van de

redenen waarom men u altijd adviseert om bij het programmeren zoveel mogelijk gebruik te maken van integer arrays. Deze nemen veel minder geheugenruimte in beslag.

In byte 2 en 3 vind u de variabele-naam in ASCII waardes. In ons voorbeeld 90 en 65 voor de hoofdletters Z en A.

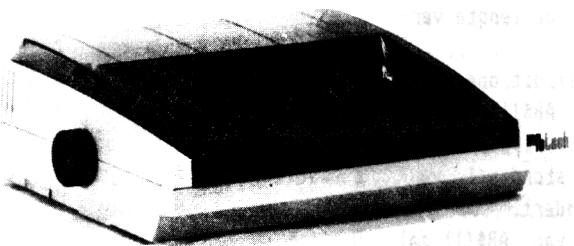
Tot byte 4 geldt voor alle soorten arrays. Alvorens byte 4 en 5 te bekijken gaan we het eerst hebben over byte zes. DIMENSIE staat er voor. Allereerst wat wordt bedoeld met dimensies?

Ieder element van een array heeft een of meer volgnummers en zoveel volgnummers ieder element aangeeft zoveel dimensies heeft dat array waarbij van iedere dimensie de lengte kan worden ingesteld d.m.v. het DIMmen.

In ons voorbeeld heeft het array ZA% slechts een dimensie. Van een dimensie hoeft dus slechts de lengte (aantal elementen) te worden genoteerd.

Zoveel dimensies er zijn, zoveel maal twee geheugenplaatsen volgen er na de zesde byte voor opslag van het

LOW-COST PRINTER VOOR PHILIPS MICROCOMPUTERS



- 100 Cps printsnelheid, bi directioneel
- 9x8 matrix
- Dot adressable grafics
- Compressed, double width, emphasize, elite, underlining, super-/subscript
- 10/18 cpi 80 cpl bij 10 cpi, 142 cpl bij 18 cpi
- 6/8/10 lines per inch
- Hor./vert. tabulatie
- Frictionfeed papiertransport met push tractors
- Interfaces: 8 bit parallel, RS 232 C
- Papierbreedte max. 10 inch (254 mm.)
- Epson compatible

MT80 + Matrixprinter

voor GPC-leden geldt een speciale GPC-prijs

infotech
computers

stationsplein 26
6512 AB NIJMEGEN
tel.: 080-23 15 20

aantal elementen per dimensie. Als we als voorbeeld het array ZA% hadden geDIMd met DIMZA%(100,10,10) dan hadden er na de zesde byte 3 (dimensies) maal 2 (geheugenplaatsen) gevuld. Element 0 vind u dan pas op de 13e byte.

Nu we hebben gekeken naar de dimensies en hun lengte kunnen we byte 4 en 5 makkelijk verklaren.

Op deze geheugenplaatsen vind u de lengte van het array vanaf byte 6. We gebruiken hiervoor de formule: lengte arraygegevens + 2 x het aantal dimensies + 1. In ons voorbeeld:

$$(2*101) + (2*1) + 1 = 205$$

Alvorens de stringarray wat nader te bekijken zullen we eerst een kijkje nemen in de stringruimte. Hiervoor gebruiken we:

```
AB$(0) "de"
AB$(1) "plaats"
AB$(2) "van"
```

als voorbeeld.

De bovengrens van het geheugen is in ons voorbeeld 65535.

		string
65525	118 v	
26	97 a	2
27	110 n	

65528	112 p	
29	108 l	
30	97 a	
31	97 a	
32	116 t	
33	115 s	

65534	100 d	0
35	101 e	

Opvallend is dat de strings geheugenplaatsen vult vanaf de bovenkant van het geheugen naar "beneden".

De stringdescriptoren, waar al eerder over is gesproken, wijzen naar de eerste byte van elke string. In ons voorbeeld naar 65534, 65528 en 65525 voor respectievelijk AB\$(0), AB\$(1) en AB\$(2). Deze verwijzingen moeten dan ook terug te vinden zijn in de stringarray samen met de lengte van iedere string afzonderlijk. De lengtes zijn dus 2, 6 en 3 bytes.

S T R I N G A R R A Y byte

TYPE	3	

NAAM	65	2
	66	3

LENGTE	244	4
vanaf hier	2	5

DIMENSIE	1	6

Aantal	251	7
elementen	0	8

Element 0	2	9

	254	10
	255	11

Element	6	12

	248	13
	255	14

Element 2	3	15

	245	16
	255	17

Element 3	7	18

	238	19
	255	20

enzovoorts

Stringarray AB\$ gedimensioneerd op 250 d.m.v. de opdracht DIM AB\$(250)

Nu weet u hoe en waar de strings in het geheugen worden opgeslagen. Echter, wat gebeurt er als we de strings veranderen en de lengte veranderen?

Stel we nemen AB\$(1) uit ons voorbeeld en veranderen AB\$(1) in "letters". U neemt natuurlijk aan dat de plaats waar AB\$(1) stond n.l. vanaf 65528 niet verandert.

Wel, de oude inhoud van AB\$(1) zal vanaf 65528 blijven staan! Op een nieuwe plaats in het geheugen wordt de nieuwe inhoud van AB\$(1) genoteerd en de stringdescriptor zal nu naar die nieuwe plaats gaan wijzen.

Het zal u dan ook wel duidelijk zijn dat hierdoor de stringruimte wel erg gauw "vol" zal zijn.

Zodra de P2000 constateert dat de

stringruimte vol raakt gaat hij "reorganiseren. De "oude" niet meer in gebruik zijnde strings worden nu overschreven door de nieuwe strings en de hele zaak wordt opgeschoven zodat er weer ruimte ontstaat voor nieuwe strings.

Deze "reorganisatie" is ook te activeren met de opdracht A=FREE(""). Na uitvoering van deze opdracht bevat A de na "opruiming" overgebleven stringruimte.

We hebben nu gekeken naar variabele arrays en string arrays. De elementen van de variabele arrays hebben we eigenlijk nog niet bekijken.

De elementen van de integer arrays geven niet zoveel problemen. Deze gebruiken twee bytes om getallen op te slaan waarbij we gewoon onze formule toe kunnen passen. De elementen van enkele precisie en dubbele precisie arrays zijn veel gecompliceerder.

Om het verhaal over arravs niet te lang te maken laat de redactie dit onderwerp rusten tot een volgende keer. Bij bankswitsching wordt in de gepubliceerde routines slechts gemaakt van integer arrays.

Het videoscherm, de geheugenplaatsen die corresponderen met het beeldscherm, bevindt zich van &H 5000 t/m &H 57FF. (20480-22527) Het deel dat wordt weergegeven op het scherm is opgebouwd uit 24 regels en 80 kolommen. De linkerhelft vormt het normale beeldscherm. (40 kolommen)

Het beeldscherm kan worden opgenomen door de geheugenplaatsen vast te leggen in een array, als strings, in een bank of weg te schrijven naar cassette.

Opslaan in een bank is mogelijk door telkens 40 plaatsen van het beeld via blockmove achter elkaar te kopiëren.

Een scherm bestaat dan uit:

$$24 \times 40 = 960 \text{ bytes.}$$

In een bank kunnen dan 8 beelden worden opgeslagen.(8 x 960 = 7680 bytes)
Indien 22 regels i.p.v. 24 worden opgeslagen dan 9 beelden.(x 880 bytes)

De inhoud van een bank kan dan worden weggeschreven en later weer terug van cassette in een bank worden gebracht. Vanuit de bank kunnen de beelden weer achter elkaar op het scherm worden geprojecteerd.

Met een 40/80 karakterkaart is het mogelijk het volledige beeldscherm van 80 kolommen zichtbaar te maken. Deze beelden bedragen 1920 bytes, zodat hiervan slechts 4 beelden in een bank passen.

Wanneer het normale geheugen en de banken volledig worden gevuld, is het mogelijk om 40 beelden op te slaan. Dit komt overeen met 15 A4-pagina's van 64 regels.

$$\begin{aligned} 64 \times 80 &= 5120 \text{ bytes} = 5 \text{ kb.} \\ 15 \text{ pagina's} \times 5120 &= 76800 \text{ bytes.} \\ 40 \text{ beelden} \times 1920 &= 76800 \text{ bytes.} \end{aligned}$$

Op deze manier is het mogelijk grote hoeveelheden tekst, direct zichtbaar

L N K E R S C H E R M

	kolom	0	3	
regel	&H	dec	0123456789 0123456789 0123456789	dec &H regel
1	5000	20480	20519 5027 1
2	5050	20560	20599 5077 2
3	50A0	20640	20679 50C7 3
4	50F0	20720	20759 5117 4
5	5140	20800	20839 5167 5
6	5190	20880	20919 51B7 6
7	51E0	20960	20999 5207 7
8	5230	21040	21079 5257 8
9	5280	21120	21159 52A7 9
10	52D0	21200	21239 52F7 10
11	5320	21280	21319 5347 11
12	5370	21360	21399 5397 12
13	53C0	21440	21479 53E7 13
14	5410	21520	21559 5437 14
15	5460	21600	21639 5487 15
16	54B0	21680	21719 54D7 16
17	5500	21760	21799 5527 17
18	5550	21840	21879 5577 18
19	55A0	21920	21959 55C7 19
20	55F0	22000	22039 5617 20
21	5640	22080	22119 5667 21
22	5690	22160	22199 56B7 22
23	56E0	22240	22279 5507 23
24	5730	22320	22359 5757 24

te maken en achter elkaar uit te laten printen met behulp van de routine die wordt aangeroepen met SHIFT-00.

R E C H T E R S C H E R M

	kolom	0	1	2	3
regel	&H	dec	0123456789 0123456789 0123456789	dec &H regel	
1	5028	20520	20559 504F 1	
2	5078	20600	20639 509F 2	
3	50C8	20680	20719 50EF 3	
4	5118	20760	20799 513F 4	
5	5168	20840	20879 518F 5	
6	51B8	20920	20959 51DF 6	
7	5208	21000	21039 522F 7	
8	5258	21080	21119 527F 8	
9	52A8	21160	21199 52CF 9	
10	52F8	21240	21279 531F 10	
11	5348	21320	21359 536F 11	
12	5398	21400	21439 53BF 12	
13	53E8	21480	21519 540F 13	
14	5438	21560	21599 545F 14	
15	5488	21640	21679 54AF 15	
16	54D8	21720	21759 54FF 16	
17	5528	21800	21839 554F 17	
18	5578	21880	21919 559F 18	
19	55C8	21960	21999 55EF 19	
20	5618	22040	22079 563F 20	
21	5668	22120	22159 568F 21	
22	56B8	22200	22239 56DF 22	
23	5608	22280	22319 572F 23	
24	5758	22360	22399 577F 24	

De Z80 processor werkt met geheugenaanplaatsen die registers worden genoemd. Van belang zijn de registers:

A (accumulator)
F (vlagregister)
de secundaire registers;
B,C,D,E en H
SP (stack-pointer)
PC (program counter)
de index registers;
IX en IY

De secundaire registers worden ook in paren gebruikt, om getallen groter dan 255 te kunnen opslaan;

B-C, D-E, H-L

De processor voert operaties uit aan de hand van de Z80 Code.(Zie tabel)
Een machinetaalprogramma is een reeks van instructies in binaire getallen.

In de praktijk wordt gewerkt met hexadecimale getallen. Het programmeren gaat gemakkelijk door te werken met Assembleertaal. Deze taal hanteert afkortingen (Mnemonics) van de uiteindelijke operatie die van de Z80 verwacht wordt. Zie hiernaast.

Deze instructies worden d.m.v. het ASSEMBLER - programma vertaald in hexadecimale getallen. Deze getallen worden uiteraard aan de processor binair aangeboden.

Bijvoorbeeld:

```
LD BC,0800 01 00 08
LD (DE),A 12
DEC H 25
OUT 94 D3 94
SBC HL,SP ED 72
LD IX,7000 DD 21 00 70
```

Omgekeerd, vertalen van de Z80 Operation Code naar Assembleertaal heet Disassemble.

In de P2000 INTERPRETER kan op adres &H 105C de schrijf-routine en op adres &H 1059 de lees-routine worden aangeroepen. De registers moeten dan als volgt gevuld worden:

In register A de beginletter van de naam (in ASCII-code) waaronder het programma moet worden opgeslagen of ingelezen.

In registerpaar HL het startadres.

In registerpaar DE het eindadres + 1.

Blockmove is het d.m.v. een machine-taalroutine verplaatsen van een blok gegevens in het geheugen. In feite wordt er niets verplaatst, maar gekopieerd. Op de oude geheugenplaatsen zijn na afloop alle gegevens terug te vinden mits ze niet worden overlapt door de verplaatste gegevens. Het HL register moet worden gevuld met het startadres. In BC het bestemmingsadres en BC dient het aantal te "verplaatsen" bytes bevatten.

Bankswitchingspecial

LD A,B	Laad in register A het getal uit register B.
LD C,(HL)	Laad in register C het getal dat de geheugenplaats in HL aanwijst.
LD HL,5000	Laad in reg.paar HL het getal 5000. In H komt 00 en in L 50.
LD (E000),HL	Laad in geheugenplaats E000 en E001 het getal in HL.
LD A,(650D)	Laad in register A het getal uit geheugenplaats &H 650D.
LD (HL),L	Laad in de geheugenplaats die HL aanwijst het getal uit register L.
INC DE	Increase (=verhoog met 1) het getal in register DE.
ADD HL,BC	Addition (=optellen) bij HL de inhoud van BC.
SBC HL,DE	Subtract (=aftrekken) van HL de inhoud van DE.
JP 9000	Jump (=spring) naar adres &H 9000 om vanaf daar de instructies uit te voeren.
CALL 1059	Call (=roep aan) de subroutine vanaf adres &H 1059 aan en keer dan terug.
PUSH BC	Vul DE met de inhoud die op de stapel werd gezet.
POP DE	Plaats de inhoud van DE op de STACK.(stapel)
RET	Return (=terug) naar vanwaar naar de routine werd gesprongen.
CP 01	Compare (=vergelijk) het getal in register A met het getal &H 01.

Indien de P2000 over het volledige geheugen beschikt, zijn er 64 Kb. adresseerbaar. Echter diverse zaken eisen geheugenruimte op zoals we hebben gezien bij de geheugenindeling.	moet het toevlucht worden gezocht tot Bankswitching.
MONITOR	4K
BASIC-INTERPRETER	16K
VIDEOGEHEUGEN gereserveerd	4K
Administratie en bufferruimten	1351 bytes
Stringruimte start op met	50 bytes
Registers	16 bytes
	TOTAAL 25993 bytes

OVER aan vrij geheugen:
 $65535 - 25993 = 39542$ bytes

Indien men meer gegevens kwijt wil,



STARTPROGRAMMA [eerst runnen, daarna het programma 1 of 2 apart laden.]

```

10 CLEAR 50,&HDE7F
20 DATA 3A,0D,65,D3,94,ED,5B,70,60,3A,72,60,2A,73,60,01,2B,00,C5,ED,B0,
     C1,09,3D,B7,C8,18,F6 :REM USR4(N)
30 DATA 3A,0D,65,D3,94,2A,70,60,3A,72,60,ED,5B,73,60,01,2B,00,C5,ED,B0,
     EB,C1,09,EB,3D,B7,C8,18,F4 :REM USR5(N)
40 DATA 2A,75,60,ED,5B,77,60,ED,4B,79,60,ED,B0,C9 :REM USR6(O)
50 DATA 3A,7B,60,2A,7C,60,ED,5B,7E,60,CD,59,10,C9 :REM USR7("n")
60 DATA 2A,E7,DE,44,4D,2A,0D,65,C5,22,EB,DE,3A,E9,DE,D3,94,AF,47,4E,23,
     5E,23,56,EB,11,00,DF,D5,C5,ED,B0,3A,EA,DE,D3,94,C1,D1,EB,ED,5B,
     ED,DE,D5,ED,B0,ED,53,ED,DE,2A,EB,DE,23,D1,73,23,72,23,C1,0B,78,
     B1,20,C6,2A,E5,DE,22,00,E0,2A,E7,DE,22,02,E0,29,ED,4B,E7,DE,09,
     44,4D,2A,0D,65,ED,5B,ED,DE,ED,B0,C9 :REM USR9(V)
70 FORI=OT057:READA$:POKE&H61AE+I,VAL("&H"+A$):NEXTI
80 FORI=OT027:READA$:POKE&H6360+I,VAL("&H"+A$):NEXTI
90 FORI=OT095:READA$:POKE&HDE80+I,VAL("&H"+A$):NEXTI
100 DEFUSR4=&H61AE:DEFUSR5=&H61CA:DEFUSR6=&H6360:DEFUSR7=&H636E:
      DEFUSR9=&HDE80

```

programma 1.

STRINGS MAKEN :

10 CLEAR8200,&HDE7F	stringruimte op 8200
200 PRINTCHR\$(12)	schoon scherm
210 DEFINTA-T	variabelen A-T zijn integers
220 DEFSTRU-Z	variabelen U-Z zijn strings
230 DEFFNA! (A!)=PEEK(A!)+256*PEEK(A!+1)	maakt 16 bits getal
240 DEFFNB(A!)=INT(A!/256)	splitst 16 bits getal,
250 DEFFNA(A!)=A!-256*FNB(A!)	in hoge en lage byte
260 Z!=57344	begin bank
270 PRINT"max.aantal strings";	te verwachten max.
280 INPUTA	
290 DIMU(A)	
300 GOSUB910	
310 PRINT"hoeveel strings maken";	
320 INPUT H	aantal strings
330 K=0	
340 FORI=1TOH	
350 PRINT"string" I	
360 LINEINPUTU(I)	invoer strings
370 IFU(I)=="THEN350	indien alleen return
380 K=K+LEN(U(I))	lengte strings tesamen
390 NEXTI	

VERTALEN NAAR BANK :

410 GOSUB810	overzicht strings
420 GOSUB710	welke bank
430 K=K+4+3*H	lengte totale file
440 M=N	
450 GOT0530	uit stringruimte
510 PRINT"naar bank";	
520 INPUTM	
530 POKE&HDEE5,FNA(K):POKE&HDEE6,FNB(K)	uiteesadressen,
550 POKE&HDEE7,FNA(H):POKE&HDEE8,FNB(H)	invullen
570 POKE&HDEE9,N	van bank
580 POKE&HDEEA,M	naar bank
590 POKE&HDEED,4:POKE&HDEEE,&HE004	opslaan v.a.&HE004
610 A=USR9(VARPTR(U(1)))	stringroutine
620 N=M	
630 GOSUB810	overzicht strings
640 GOT0910	naar keuze

```

710 PRINT"welke bank";
720 INPUTN                                banknummer
730 OUT&H94,N                               inschakelen bank
740 RETURN
750 FORI=1TOH                               overzicht aantal strings
760 PRINTI;
770 PRINTU(I);
780 PRINT" ";
790 NEXTI
800 PRINT:RETURN
810 PRINT"1=strings maken   ";            menu
820 PRINT"2=nakijken"
830 PRINT"3=vertalen naar bank  ";
840 PRINT"4=opnemen   ";
850 PRINT"5=laden";
860 INPUTA
870 ONAGOTO310,1010,510,1210,1230
STRINGS UIT BANK :
1010 GOSUB710                                bank
1020 A=0
1030 K=FNA!(Z!)
1040 H=FNA!(Z!+2)                            lengte file
1050 Y!=Z!+K-3*H                            aantal strings
1060 G!=VARPTR(U(1))                         plaats van descriptoren
1070 IFG!<OTHENG!=G!+65536                  plaats van stringdescriptor
1080 POKE&H6077,FNA(G!):POKE&H6078,FNB(G!)  in arrayruimte
1090 POKE&H6075,FNA(Y!):POKE&H6076,FNB(Y!)
1100 POKE&H6079,FNA(H*3):POKE&H607A,FNB(K*3)  vastleggen uitleesgegevens
1110 GOSUB810                                overzicht strings
1120 GOTO910                                 naar menu
OPNEMEN EN LADEN :
1210 POKE&H6379,&H5C:GOTO1250                byte voor schrijven
1230 POKE&H6379,&H59                           byte voor lezen van cassette
1240 K=8191                                  gehele bank
1250 GOSUB710                                bank
1310 PRINT"welke naam";
1320 INPUTN$                                 ie letter van naam vullen met
1330 POKE&H607C,FNA(Z!):POKE&H607D,FNB(Z!)    de door de machinetaal
1350 POKE&H607E,FNA(Z!+K):POKE&H607F,FNB(Z!+K)  uit te lezen adressen
1370 POKE&H607B,ASC(N$)                      naam
1380 A$=USR7(N$)                            cassetteroutine

```

Oproep !!!

De redactie is op zoek naar een teken programma waarmee plaatjes kunnen worden uitgeprint. Aangezien wij zelf niet de capaciteiten bezitten om een dergelijk programma te maken, doen wij een beroep op programmeurs die denken ons van dienste te kunnen zijn.



Onze wensen liggen als volgt.

- Een nieuw ontwikkeld programma dat plaatjes aanmaakt volgens het view-dataprincipe en ze volgens de Bit Image Printroutine uitprint. En als het mogelijk is aangevuld met een routine waarmee plaatjes op cassette kunnen worden weggeschreven (of op disk);
- Een printprogramma voor plaatjes die zijn aangemaakt met behulp van Peter's Plaatjes Programma.
- Aanpassing van het Peter's Plaatjes Programma met een Bit Image Printroutine.

Het programma hoeft niet perse met allerlei 'luxe' afgewerkt te zijn. Een eenvoudig programma redt ons uit de nood.

Namens de redactie Peter Pit ■



programma 2.

VIDEOTEXT

```

10 CLEAR 50,&HDE7F
110 DEFINTA-U
120 DEFSTRV-Z
130 DEFFNC$=CHR$(4)+CHR$(24)+CHR$(1)
140 DEFFNA!(A!)=PEEK(A!)+256*PEEK(A!+1)
150 DEFFNB(A!)=INT(A!/256)
160 DEFFNA(A!)=A!-256*FNB(A!)
170 Z!=57344

TEKST OP SCHERM
200 PRINTCHR$(12)"1=tekst maken ";
210 PRINT"2=bank naar beeld"
220 PRINT"3=opnemen 4=laden";:INPUTA
230 ONAGOTO310,1010,2010,2110
310 GOSUB850
320 PRINTCHR$(12)
330 PRINTCHR$(2);
340 Q=20480
400 J=PEEK(Q)
410 U=PEEK(Q)
420 IF J>159THENJ=J-128
430 POKEQ,J+128
440 POKE&H600C,0
450 A=INP("")
460 IFA=35THENA=95ELSEIFA=95THENA=35
470 IFA>31ANDA<128THENJ=A:K=1:GOT0610
500 IFA=16THENK=-1
510 IFA=17THENK=-80
520 IFA=18THENK=80
530 IFA=19THENK=1
540 IFA=7THEN910
550 IFA=130THEN200
560 IFA=13THENGOSUB710:GOSUB810:Q=00:
      GOT0750
570 IFA=8THENGOSUB710:Q=Q-1:POKEQ,32:
      J=32:GOT0610
580 IFA=9THENGOSUB710:Q=Q+8:GOT0750
600 J=U
610 POKEQ,J
620 Q=Q+K
630 IFQ<20480RQ>22399THENQ=Q-K
640 IF (Q-20480)MOD80>39thenQ=Q+40
650 GOT0400
710 K=0
720 POKEQ,U
730 RETURN
750 J=PEEK(Q)
760 GOT0610
810 O=INT(Q-20480)/80
820 IFO<24THENOO=20480+(O+1)*80:RETURN
830 OO=22320:RETURN
850 PRINT"welke bank";
860 INPUT N
870 OUT&H94,N
880 RETURN

variabelen A-U zijn integers
variabelen V-Z zijn strings
regel 24 kolom 1
maakt 16 bits cijfer
splitst 16 bits cijfer,
in hoge en lage byte
&HE000 = begin bank

schoon scherm
keuze

welke bank
schoon scherm
normale cursor uit
startplaats linksboven &H5000
inhoud cursorplaats
idem
indien karakter geinverteerd
karakter inverteren
inputbuffer leeg maken
toetsindruk in ASCII-code
omwisselen # en £
indien letter of cijfer
cursortoets naar links
cursortoets omhoog
cursortoets omlaag
cursortoets naar rechts
OPN beeldscherm naar bank
DEF naar menu
RETURN regel omlaag vooraan regel

wis 1 plaats terug

TAB 8 plaatsen verder
karakter blijft gelijk
druk karakter af
cursor naar nieuwe plaats
indien buiten beeld, terug
indien rechterscherm
terug
cursor blijft op plaats
geinverteerd uit

karakter blijft gelijk

regelnummer
naar volgende regel
indien regel 24

banknummer
bank N wordt ingeschakeld

```

BEELD NAAR BANK :

```

910 GOSUB 710          geïnverteerd uit
920 GOSUB1110          welk beeld in bank
930 A=24               24 regels
940 GOSUB 1210          uit te lezen adressen vullen
950 AA=USR4(N)          routine beeld naar bank
960 GOTO340             terug naar beeld
1110 GOSUB 1310          regel 24 even opslaan
1120 PRINTFNC$;          op regel 24
1130 PRINTCHR$(2);       cursor uit
1140 GOSUB1810           welk beeld
1150 GOSUB1410           regel 24 weer terug
1160 RETURN
1210 GOSUB1910          berekening plaats in bank
1220 POKE&H6070,FNA(A!):POKE&H6071,FNB(A!)  vastleggen plaats
1240 POKE&H6072,A      aantal regels
1250 POKE&H6073,0:POKE&H6074,&H50            begin scherm &H5000
1270 RETURN
1310 A!=&H5730          regel 24,
1320 B!=&H6260          naar inputbuffer
1330 POKE&H6075,FNA(A!):POKE&H6076,FNB(A!)  vastleggen om uit te lezen,
1340 POKE&H6077,FNA(B!):POKE&H6078,FNB(B!)  door machinetaalroutine
1350 POKE&H6079,&H50:POKE&H607A,0            lengte &H50 = 80 karakters
1370 AA=USR6(0)          blockmove
1380 RETURN
1410 A!=&H6260          uit inputbuffer,
1420 B!=&H5730          naar regel 24
1430 GOTO1330           enz.
1810 PRINT"welk beeld ? ";    plaats in bank
1820 B=INP("")-48          ASCII-code -48 = cijfer
1830 IFB<1THEN1820         getal tussen 1,
1840 IFB>8THEN1820         en 8
1850 RETURN
1910 A!=Z!+2+(B-1)*A*40
1920 RETURN
BANK NAAR BEELD :
1010 GOSUB850            welke bank
1020 PRINTCHR$(12)        schoon scherm
1030 GOSUB1110           welk beeld
1040 GOSUB1060           beeld uit bank
1050 GOTO1510
1060 A=24               24 regels
1070 GOSUB1210           vullen adressen
1080 AA=USR5(N)          routine bank naar beeld
1090 RETURN
1510 GOSUB1310           regel 24 opbergen
1520 PRINTFNC$CHR$(2);    op regel 24, cursor uit
1540 PRINT"1=edit  2=bladeren"; keuze
1550 A=INP("")-48          invoer cijfer
1560 GOSUB1410           regel 24 weer terug
1570 IFA=2THEN1610        bladeren naar 1610
1580 GOTO330              EDIT, naar scherm
1610 A=INP("")            toets in ASCII-code
1620 IFA=16THENB=B-1      toets naar links, beeld terug
1630 IFA=19THENB=B+1      toets naar rechts, verder
1640 IFB=0THENGOSUB1710   naar vorige bank
1650 IFB=9THENGOSUB1750   naar volgende bank
1660 IFA=32THENGOSUB1410:GOTO330  spatie, naar EDIT
1670 GOSUB1060             beeld naar bank
1680 GOSUB1310             regel 24 opbergen
1690 PRINTFNC$N;B;          nummer bank en beeld

```

1700 GOTO1610
 1710 N=N-1
 1720 IFN<OTHENN=0:B=1:RETURN
 1730 B=8
 1740 RETURN
 1750 N=N+1
 1760 IFN>5THENN=5:B=8:RETURN
 1770 B=1
 1780 RETURN
OPNEMEN EN LADEN :
 2010 POKE&H6379,&H5C
 2020 GOSUB2210
 2030 GOTO340
 2110 POKE&H6379,&H59
 2120 GOSUB2210
 2130 B=1:GOSUB1060
 2140 GOTO340
 2150 POKE&H607C,FNA(A!):POKE&H607D,FNB(A!)
 2170 POKE&H607E,FNA(B!):POKE&H607F,FNB(B!)
 2190 A\$=USR7(N\$)
 2200 RETURN
 2210 PRINT"1=beeld 2=bank";
 2220 INPUTA
 2230 GOSUB2510
 2240 GOSUB850
 2250 POKE&H607B,ASC(N\$)
 2260 ONAGOTO2310,2410
 2310 GOSUB1810
 2320 PRINTB;
 2330 GOSUB1910
 2340 B!=A!+960
 2350 GOSUB2150
 2360 RETURN
 2410 A!=57344
 2420 B!=A!+7682
 2430 GOSUB2150
 2440 RETURN
 2510 PRINT"welke naam";
 2520 INPUTN\$
 2530 RETURN

doorbladeren
 bank 0 blijft
 beeld 8
 volgende bank
 bank 5 blijft
 beeld 1
 byte voor opnemen
 cassetteroutine
 terug naar EDIT-stand
 byte voor laden
 cassetteroutine
 beeld 1 naar scherm
 naar EDIT-stand
 vastleggen om uit te lezen
 idem
 cassetteroutine
 keuze om op te nemen
 naam
 banknummer
 letter voor routine
 beeldnummer
 beginadres in bank
 eindadres block
 naar cassetteroutine
 begin bank
 eindadres
 naar cassetteroutine
 1e letter naam op cassette

MACHINE-TAAL PROGRAMMA'S BANKSWITCHING P-2000 T / 102

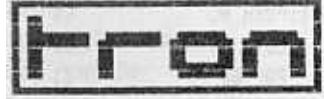
USR 5(N)	USR 2("n")	USR 1(N)
VIDEO-BEELD naar BANK N	BANK naar CASSETTE naam"n"	BASIC naar BANK N
LD A,(650D)	3A 0D 65	LD A,(650D)
OUT 94	D3 94	OUT 94
LD A,(6070)	3A 70 60	LD DE,(625C)
LD HL,(6071)	2A 71 60	ED 5B 5C 62
LD DE,(6073)	ED 5B 73 60	LD HL,(6405)
LD BC,0028	01 28 00	SBC HL,DE
start:PUSH BC	C5	ED 52 01
LDI	ED B0	LD (E000),HL
POP BC	C1	22 00 E0
ADD HL,BC	09	LD B,H
DEC A	3D	44 00 00
OR A	B7	LD C,L
RET Z	C8	4D 00 00
JR :start	18 F6	EX DE,HL
		EB
		LD DE,E002
		11 02 E0
		LDIR
		ED B0 00
		RET
		C9 00 00

USR9(V)		USR 4(N)		USR 3("n")	
STRINS naar BANK, V=VARPTR(U\$(1))		BANK N naar VIDEO-BEELD		CASSETTE naar BANK naam"n"	
LD HL,(DEE7)	2A E7 DE	LD A,(650D)	3A 0D 65	LD HL,(650D)	2A 0D 65
LD B,H	44	OUT 94	D3 94	INC HL	23
LD C,L	4D	LD A,(6070)	3A 70 60	LD E,(HL)	5E
LD HL,(650D)	2A 0D 65	LD HL,(6071)	2A 71 60	INC HL	23
start:PUSH BC	C5	LD DE,(6073)	ED 5B 73 60	LD D,(HL)	56
LD (DEEB),HL	22 EB DE	LD BC,0028	01 28 00	LD A,(DE)	1A
LD A,(DEE9)	3A E9 DE	start:PUSH BC	C5	LD HL,E000	21 00 E0
OUT 94	D3 94	LDIR	ED B0	LD DE,0000	11 00 00
XOR A	AF	EX DE,HL	EB	CALL 1059	CD 59 10
LD B,A	47	POP BC	D1	RET	C9
LD C,(HL)	4E	ADD HL,BC	09		
INC HL	23	EX DE,HL	EB		
LD E,(HL)	5E	DEC A	3D	USR 7(0)	
INC HL	23	OR A	B7	CASSETTEROUTINE	
LD D,(HL)	56	RET Z	C8		
EX DE,HL	EB	JR :start	18 F4	LD A,(6070)	3A 70 60
LD DE,DF00	11 00 DF			LD HL,(6071)	2A 71 60
PUSH DE	D5			LD DE,(6073)	ED 5B 73 60
PUSH BC	C5	USR 8(V)		CALL 1059	CD 59 10
LDIR	ED B0	ARRAY naar BANK	V=VARPTR(6%(0))	RET	C9
LD A,(DEEA)	3A EA DE				
OUT 94	D3 94	LD HL,(605D)	2A 0D 65		
POP BC	C1	PUSH HL	E5		
POP DE	D1	DEC HL	2B	USR 6(0)	
EX DE,HL	EB	DEC HL	2B	BLOCKMOVE	
LD DE,(DEED)	ED 5B ED DE	DEC HL	2B		
PUSH DE	D5	DEC HL	2B	LD HL,(6071)	2A 71 60
LDIR	ED B0	LD B,HL	46	LD DE,(6073)	ED 5B 73 60
LD (DEED),DE	ED 53 ED DE	DEC HL	2B	LD BC,(6075)	ED 4B 75 60
LD HL,(DEEB)	2A EB DE	LD C,HL	4E	LDIR	ED B0
INC HL	23	DEC BC	0B	RET	C9
POP DE	D1	LD (E000),BC	ED 43 00 E0		
LD (HL),E	73	DEC BC	0B		
INC HL	23	DEC BC	0B		
LD (HL),D	72	LD DE,E002	11 02 E0		
INC HL	23	POP HL	E1		
POP BC	C1	LDIR	ED B0		
DEC BC	0B	RET	C9		
LD A,B	78				
OR C	B1				
JR NZ start	20 C6				
LD HL,(DEE5)	2A E5 DE	USR 0(N)			
LD (E000),HL	22 00 E0	BANK N naar BASIC			
LD HL,(DEE7)	2A E7 DE				
LD (E002),HL	22 02 E0	LD A,(650D)	3A 0D 65		
ADD HL,HL	29	OUT 94	D3 94		
LD BC,(DEE7)	ED 4B E7 DE	LD DE,6547	11 47 65		
ADD HL,BC	09	LD HL,(E000)	2A 00 E0		
LD B,H	44	ADD HL,DE	19		
LD C,L	4D	LD (6405),HL	22 05 64		
LD HL,(650D)	2A 0D 65	LD HL,E002	21 02 E0		
LD DE,(DEED)	ED 5B ED DE	LD BC,(E000)	ED 4B 00 E0		
LDIR	ED B0	LDIR	ED B0		
RET	C9	RET	C9		

mini

Te koop P2000T 32Kb + floppy-controller- kaart + 80 karakterkaart.
 Dubbel floppystation.
 Printer PER 3100, Monitor (groen),
 Epromprogrammer.

Inlichtingen: telefoon 01828-19165.



RAM-Dos Jeroen Hoppenbrouwers

Zoals in de vorige TRON al min of meer was aangekondigd, volgt hier het vervolg over RDOS, en wel de machine-taal. Om nu te voorkomen dat een groot gedeelte van de TRON-lezers direct afhaakt omdat ze nog niets van machinetaal afweten, zal ik geen droge listings oplepelen, maar de circa tweeduizend bytes die RDOS groot is opspliten in kleine, overzichtelijke stukjes die bijna instructie voor instructie besproken worden. Dat het een vervolgverhaal wordt ligt voor de hand.

Maar eerst: wat is machinetaal en waarom heb ik RDOS niet gewoon in BASIC geschreven?

Dat laatste is wel duidelijk: het kan niet in BASIC omdat RDOS nu juist programma's verplaatst in het geheugen. Om RDOS te kunnen RUNnen zou het eerst in de BASIC-ruimte gezet moeten worden en dat deed het nou net zelf.. De tweede reden is het enorme snelheidverschil tussen BASIC en machinetaal. BASIC biedt veel service aan de programmeur, zoals PRINT, IF..THEN ELSE en SIN(x) bijvoorbeeld.

Die verrichten ongemerkt een heleboel werk! Ook kent BASIC talloze foutmeldingen, maar... dat gaat allemaal ten koste van de snelheid.

BASIC is dus TRAAg. "Hoezo traag?", zult u zeggen, "ik kan het allemaal toch al niet bijhouden, zo'n computer rekent iedereen voorbij!" Jawel, maar RUN nu eens een van de vele programma's waarin DATA-regels met machine-taal staan. Deze code's worden met een lus POKE- voor POKE- in het geheugen gezet, en dat kost zichtbaar veel tijd!

Stel nu dat RDOS in BASIC geschreven zou zijn, dan zou het - om een redelijk programma te verplaatsen - toch zeker enige duizenden kerren een lus moeten doorlopen. Dat zou werkelijk tientallen minuten gaan kosten! Een losse recorder is nog sneller.....

Machinetaal dus. Geen service, alles zelf doen en eigenlijk maar een foutmelding: een droge piep en PHILIPS CASSETTE BASIC op het scherm.

Waar begin je toch aan? Nou, dat valt allemaal best mee. Laten we eerst maar eens gaan kijken naar het hart van de P2000, de Z80-microprocessor, een hart dat ongeveer twee-en-een-

half miljoen keer per seconde klopt. In die processor zitten een tiental zogenaamde registers, kleine geheugentjes waar we 1 byte in kunnen stoppen (een byte is een geheel getal tussen nul en 255).

Alle registers hebben een naam, een letter lang. Het belangrijkste register is register A (ook wel eens accumulator genoemd). Om nu bijvoorbeeld het getal 0F (hexadecimaal, in machinetaal is alles hexadecimaal) in dat register A te stoppen geven we de processor de instructie

LD A,0F

LD is de afkorting van LOAD, vrij vertaald staat er dus "laat A met 0F". Zo zijn er nog meer opdrachten, bijvoorbeeld om op te tellen:

ADD A,B

telt de inhoud van B bij die van A op en stopt de uitkomst terug in A.

SUB A,B

trekt B van A af enzovoort.

Ook is het mogelijk om twee registers samen te voegen, zodat ze samen een groter getal kunnen bevatten; 1 byte is immers maximaal $2^7 = 256$, 2 bytes gaan tot $2^{15} = 65535$. Als u dit niet begrijpt, geen nood, zolang de processor het maar begrijpt. Maximaal 256 per register!

Zo worden de registers H en L heel vaak gekoppeld tot HL. We kunnen dus schrijven

LD HL,5A8B

LD H,5ABB mag niet, want zo'n groot getal kan H in zijn eentje nooit opslaan!

Als we nu alleen de registers van de microprocessor konden gebruiken, waren we snel uitgeprogrammeerd.

Nee, een computer heeft ook een geheugen, bestaande uit een ontzettend groot aantal hokjes waarin we weer 1 byte kunnen stoppen. Om elk hokje aan te kunnen geven worden ze gewoon getnummerd, en dat nummer heet dan het adres van de geheugencel. Zo'n hokje is in BASIC alleen met POKE en PEEK toegankelijk. In machinetaal kan het op veel meer manieren en veel sneller.

Zo bestaat bijvoorbeeld de instructie

LD (6F53), A

Volgens het standaardrecept wordt dus geheugenplaats nummer 6F53 geladen met de inhoud van A. Let op het gebruik van de haakjes!

6F53 wil zeggen: het getal 6F53;

(6F53) wil zeggen: het getal dat staat op geheugenplaats 6F53! Goed onthouden! Omdat de computer een hele hoop geheugencellen heeft, in ieder geval meer dan 256, zijn er ook altijd minstens 2 bytes nodig om een adres aan te geven, en bij grote computers wel meer.

Nu is het ook mogelijk om een registerpaar met een adres te laden, en zo via dat registerpaar bij een geheugenplaats te komen:

LD HL,67A3

LD A,(HL);Let op de haakjes!

Laadt in A het getal dat staat op adres 67A3. Lijkt dit u nutteloos? Een voorbeeld: Als we een programma willen maken dat op tien opeenvolgende adressen het getal 65 neerzet, dan kan dat natuurlijk met

LD A,65

LD (5000),A

LD (5001),A

LD (5002),A

etcetera, maar dat staat niet mooi en kost veel programmeergeheugen.

Veel mooier en korter is:

LD HL,5000

Lus LD (HL),65 ; Laadt 65 in ; adres nr. HL

INC HL ; Hoog HL op met ; 1 (INCREMENT)

JP Lus ; Spring (Jump) ; naar "Lus"

Tekens wordt HL met 1 verhoogd, en wijst dus steeds naar het volgende adres. Slimmerikken zien meteen dat het programma nooit zal stoppen: aan het terug springen komt geen einde!

Na enkele ogenblikken loopt de P2000 over zijn programma heen (dat staat tenslotte ook in het geheugen) en het gevolg is: DE foutmelding...!

We hebben iets nodig om zijn energie wat te temperen. Meteen maar doorprogrammeren:

LD A,00

LD HL,5000

Lus LD (HL),65 ; Laadt 65 in ; adres nr. HL

INC HL ; Hoog HL op met ; 1

INC A ; Hoog A op met ;

CP OA ; Vergelijk A met ; OA (Compare)

JP NZ Lus ; Jump if No Zero ; naar "Lus"

Aiai, dat wordt ingewikkeld. De instructie CP, wat doet die nu precies? Nou, voorlopig is het genoeg om het

volgende te onthouden:

CP getal

trekt dat getal af van de inhoud van A, maar verandert A niet! De processor onthoudt alleen of het resultaat wel of niet nul is. Hoe hij dat onthoudt, komt later wel.

En dan de laatste instructie; als het resultaat van de voorafgaande instructie niet nul was, dan springt de Z80 naar "Lus". Als het wel nul was, dan gaat de processor verder na die JP-opdracht.

Het hele programmaatje loopt dus als volgt: er wordt 65 op een adres gezet, de adressesteller wordt met 1 verhoogd en de lusteller A ook. Als A nog geen tien is (hex. 0A), dan draait de boel nog een keer, maar als A wel 0A is, levert CP 0A nul op (0A - 0A = erg nul). Aan de "vooraardelijke sprong" wordt niet voldaan, de processor kapt ermee.

Nu nog 1 ding en dan zijn we toe aan het eerste stukje RDOS. Nu komt de sub-routine. He, die ken ik van BASIC! Precies, de machinetaal-subroutine werkt net zoals de BASIC-versie, alleen heet hij niet GOSUB maar CALL (roep aan), en wordt de subroutine beëindigd met RET(urn).

Goed, nu dan eindelijk RDOS.

De routine van vandaag is de machine-taalversie van de print-opdracht. Niet wereldschokkend nee, maar wel leerzaam en om mee te beginnen heel geschikt.

In RDOS wordt een bericht als volgt op het scherm gezet:

```
LD HL,Bericht ; Bericht
                ; stelt een
                ; adres voor.
CALL Output    ; PRINT het
                ; bericht op
                ; het scherm
```

Dat ziet er simpel uit, nietwaar? HL wordt geladen met een adres (dat kan alles zijn) waar de eerste van een rijtje ASCII-codes in het geheugen staat, en dan wordt Output aangeroepen:

Outp	LD A,(HL)	; In A de in
		; houd van het
		; adres waar
		; HL naartoe
		; wijst
CP FF		; Gelijk aan
		; FF?

RET Z	; Zo ja, dan is	-Stop in A de ASCII-code van het gewenste teken (mag ook een bestrijkingsteken zijn, zoals "wisscherm" = 12 oftewel hexadecimaal 0C).
	; Output klaar	
	; (RETurn on	
	; Zero)	
CALL 104A	; PRINT de	-CALL 104A
	; letter in A	Dit werkt precies hetzelfde als in
INC HL	; Hoog HL met 1	BASIC PRINT CHR\$(x).
	; op	
JP Output	; Spring naar	Zo, dat was het dan voor deze keer.
	; Output	In de volgende afleveringen zal het wat rustiger gaan, maar de eerste keer was het echt even doorbijten.

Dit verhaal gaat net zolang door tot op het door HL aangewezen adres FF staat, dan keert de routine terug naar het hoofdprogramma.

Het te PRINTen bericht MOET dus worden afgesloten met FF, anders stopt de routine niet en schiet het hele geheugen over het scherm. (overigens best een grappig gezicht)

De subroutine Output maakt zo te zien gebruik van een andere subroutine om het echte werk te laten doen.

Deze routine zit in de BASIC-interpreter, dus in ROM, maar dat maakt niets uit. Als deze routine nu maar afblijft van registerpaar HL, dan kan het geen kwaad om vanuit deze subroutine weer een subroutine aan te roepen, en desnoods nog eens. Gelukkig houdt 104A zijn handen thuis.

Het gebruik van dit stukje BASIC op 104A heeft echter wel een nadelaatje: het gaat niet snel. Voor spelletjes, waarbij van alles over het scherm raast, dus niet geschikt, maar tekst is er heel gemakkelijk mee op het scherm te zetten. Daarbij is een griezelig hoge snelheid meestal niet nodig. Dus, om zelf mee te experimenteren:

Dit werkt precies hetzelfde als in BASIC PRINT CHR\$(x).

Zo, dat was het dan voor deze keer. In de volgende afleveringen zal het wat rustiger gaan, maar de eerste keer was het echt even doorbijten. Mocht u de machinetaalkriebels al voelen (dat gebeurt!) dan zijn er verschillende boeken en boekjes over de Z80 op de markt waar u uit kunt putten. Ze beginnen over het algemeen nog wat eerder dan ik, namelijk met de uitleg van het binaire en hexadecimale talstelsel, maar ik had behoorlijk weinig mogelijk wiskunde, of wat daar op lijkt, te gebruiken.

Let er wel goed op dat u iets over de Z80 opsnort! Ook al lijken ze van buiten bedriegelijk veel op elkaar, processoren verschillen inwendig veel meer dan u kunt vermoeden!

De volgende keer gaan we vrolijk (?) verder met machinetaal, zodat u over niet al te lange tijd in staat zult zijn de rijen instructies elders in dit blad redelijk goed te volgen.

In afwachting van uw reacties,

Jeroen Hoppenbrouwers
Wilhelminapark 8
5554 JE VALKENSWAARD
Telefoon: 04902-13808
Vidibus 400020237

Boekbespreking

Titel : Fantastische spellen voor uw PHILIPS MSX
Auteurs : H.Renko & S.Edwards
Uitgever: Addison-Wesley Europe BV
Prijs : 28,95

Dit boek, dat in november j.l. verschenen is, is er een uit een hele reeks die deze auteurs over computerspellen geschreven hebben. Je mag dus aannemen dat zij de behoeften van het grote publiek onderkennen en een interessante verzameling spellen bijeengebracht hebben. Dat is inderdaad het geval! Hoewel ik zelf geen MSX gebruiker ben

raakte ik bij het bestuderen van de spellen uit dit boek zo gefascineerd dat ik geneigd was een aantal ervan om te werken voor mijn P2000! Ook viel het op dat de auteurs programma's van enkele andere, met name genoemde "spelletjes-programmeurs" in hun boek opgenomen hebben. Misschien een suggestie voor de MSX-programmeurs onder onze lezers? De redactie verstrekkt u desgewenst het adres van de uitgever.

De 31 programma's in dit boek worden stuk voor stuk door een boeiende inleiding vooraf gegaan waardoor je

alleen met nog meer enthousiasme aan het intypen zult slaan. De meeste listings beslaan slechts een paar bladzijden en de programma-regels zijn kort. Ook wordt in de programma's veel 'REM' - toelichting gegeven. Een plezierige en leesbare opzet van het geheel. Jammer is het dat het alleen maar om het amusement gaat en dat er nauwelijks enige aandacht is geschonken aan edukatieve- of instructieve programma's. Of de programma's "HET REKENWONDER" en "ESCHER figuren" zouden hiertoe gerekend moeten worden. Behalve een 5-tal heel korte programma's van slechts 1 bladzijde zijn er ook een paar waar wel een paar uurtjes voor nodig zullen zijn om ze ingetikt te krijgen. Het langste spel heet "I.T.-DE ADVENTURE (ja, zo staat het er!) VAN DE EEUW". De andere lange heet "LAS VEGAS A GOGO" oftewel de fruitautomaat.

Als deze net zo goed is als de Jackpot, een programma voor de P2000 van v.d.Avoort & Willemsen (en daar

ziet het wel naar uit), dan is hij beslist de moeite van het intikken waard!

Een merkwaardig programma vind ik "MAANFASE ASTROLOGIE". Hier wordt bij een programma van nog geen pagina 6 bladzijden "naslagwerk" gegeven! Men vulle slechts zijn geboortedatum in het programma in. Hieruit wordt de maanfase berekend en met de toelichting uit het boek kunnen we dan iets over ons zelf te weten komen.

Om de geinteresseerde MSX-werker een zo volledig mogelijk beeld van dit boek te geven volgt hier een opsomming van alle listings:

- Galactische monsters
- Zombies in het moeras
- Keyboard Memory
- Las Vegas a Gogo
- Papapapegaai
- Duindigt
- Het niet zo verschrikkelijke kleurenprobleem
- Qui Vive
- S(hort) T(erm) M(emory)
- Van 5 naar 0

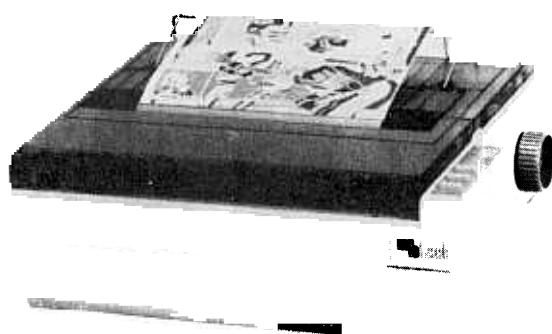
- Escher
 - Het rekenwonder
 - Haaienjacht
 - De Shakespeare Shuffle
 - Explosie
 - New York, New York
 - Code Kraken
 - Blackbox
 - Schatzoeken
 - I.T.-De Adventure van de eeuw
 - De wolf en de 5 kleine geitjes
 - Formule 1 race
 - Stuiterbal
 - De marktkramer
 - Doorzakken
 - UFO aanval
 - Mini Mancala
 - Stop it!
 - BAAB
 - Klinkers en medeklinkers
 - Maanfase astrologie
- Tenslotte moet nog vermeld worden dat precies hetzelfde boek onder de titel

MIRACULEUZE SPELEN VOOR
UW MSX COMPUTER
verkrijgbaar is voor de niet-PHILIPS
MSX gebruikers.

Veel plezier

acv

GELUIDLOZE INKJETPRINTER VOOR PHILIPS MICROCOMPUTERS



- 150 Cps printsnelheid, bi directioneel
- 9x9 matrix
- 8-bits image graphics
- Compressed, double width, emphasized, underlining
- 10/12/17 cpi 80 cpl bij 10 cpi, 96 cpl bij 12 cpi, 136 cpl bij 17 cpi
- % lines per inch + programmeerbaar per 1/2"
- Hor./vert. tabulatie
- Papiertransport d.m.v. tractors en friction feed
- Interfaces: 8 bit parallel, RS 232 C, TTY 20 mA
- Inktreservoircapaciteit 5 miljoen tekens
- Papierbreedte max. 9% inch (250 mm.)
- Epson compatible
- óók leverbaar in 132-koloms-uitvoering,
type pt 89 i, max. papierbreedte 15 1/4 inch (400 mm)
- óók leverbaar met viditel uitbreiding

pt 88 i inkjetprinter
voor GPC-leden geldt
een speciale GPC-prijs

infotech
computers

stationsplein 26
6512 AB NIJMEGEN
tel.: 080-23 15 20

Van de afdelingen

Venray

Na weer wat bijgekomen te zijn van de drukte die de open dag van september met zich meebracht is de afdeling venray weer vol goede moed begonnen met de organisatie van haar maandelijkse bijeenkomsten.

Het nieuwe computerseizoen is bij het perse gaan van deze TRON al weer bijna 4 maanden aan de gang. Dat betekent dat we al weer vier bijeenkomsten hebben gehad.

Het programma voor de nog te volgen vier bijeenkomsten is al weer gemaakt. Dat programma treft u hierbij aan.

Let goed op de data want door omstandigheden kunnen de bijeenkomsten in januari en februari niet op de TWEEDE dinsdag van de maand gehouden worden. Daarom zijn de bijeenkomsten voor deze twee maanden verzet naar de DERDE dinsdag van de maand.

18 februari 1986

In februari zal MSX centraal staan

in een MSX speciaal. Enkele MSX computers zullen ten tonele verschijnen. De BASIC cursus zal verder vervolgd worden.

11 maart 1986

In maart zal de driemaandelijkse kloavond weer plaatsvinden. De kloavond is een praktijkavond waarop de BASIC cursisten hun kennis kunnen loslaten op programmeerproblemen die door de samenstellers van de cursus worden samengesteld aan de hand van de cursussen die de laatste maanden zijn gegeven.

8 april

Naast de BASIC cursus proberen we een demonstratie te organiseren van de personal computer waar je geen NEE tegen kunt zeggen, de Philips YES.

Een en ander is natuurlijk onder voorbehoud maar het bestuur van de afdeling Venray zal er al het mogelijke voor doen om het bovenstaande te realiseren.

Mocht u meer informatie willen hebben over de afdeling Venray en de activiteiten neem dan 's-avonds contact op. (Zie afdelings-adressen)

Riko Peeters.

Den Haag

De afdeling Den Haag raakt haar 'stekkie' in de sociale werkplaats van Rijswijk kwijt. De laatste bijeenkomst, in de maand januari, heeft immiddels plaatsgevonden.

Wie helpt ons uit de problemen en kan ons zo snel mogelijk aan een nieuw onderkomen helpen?

Graag uw reacties aan de sekretaris van de afdeling Den Haag (Peter Greve, telefoon 079-515285).

Utrecht

Op maandag 25 november is er weer een bijeenkomst gehouden van de afdeling Utrecht welke onlangs werd opgericht. Een voorlopig bestuur is gevormd dat ongeveer een half jaar zal aanblijven. Hierna zal een definitief bestuur worden gekozen.

Het voorlopige bestuur onderzocht welke plaats het meest in aanmerking kwam voor het houden van de maandelijkse bijeenkomsten.

Men overwoog: Hotel Hes in de Maliette Utrecht, de huisvesting van de Schoolbegeleidingsdienst te Zeist en buurhuis Kerkveld te Nieuwegein.

De definitieve keuze werd genomen. De meesten onder u werden hiervan op reeds op de hoogte gesteld.

Peter Vierbergen startte eind september met het verzenden van brieven aan

zo'n 250 Philips-computerbezitters in de regio Utrecht. Op de eerste avond waren er twaalf belangstellenden aanwezig. Nu zijn er reeds dertig deelnemers ingeschreven.

De meesten zijn in het bezit van een P2000 T computer, maar op de bijeenkomst in Utrecht waren ook andere tv's aan vertegenwoordigd.

Uiteraard zullen wij als voorlopig bestuur trachten aan ieders wensen tegemoet te komen.

Het is daarom zeer noodzakelijk dat u uw wensen aan ons kenbaar maakt. Alle reacties zijn dan ook bijzonder welkom.

Spoedig hopen wij u meer te kunnen vertellen over de ontwikkelingen van onze afdeling.

Namens het bestuur,
Rob ter Brugge.

Het voorlopig bestuur van de afdeling Utrecht is als volgt:

A. Kurver Voorzitter
Couwenhoven 56-01
3703 XP Zeist

W.H. aarhoven Penningmeester
Linze bord 10
3991 P Houten

P. Vierbergen Contaktpersoon
v.d. Mondestraat 40a
3515 BJ Utrecht
Tel.: 030-718378

De bijeenkomsten beginnen om 19.30 u. Ze wordt gehouden op de genoemde locatie in Zeist.

Bijeenkomsten worden gehouden op de hieronder vermelde data:

17/2, 17/3, 14/4, 12/5, en 9/6

Eemland

Zaterdag 19 oktober hebben wij de algemene ledenvergadering. De opkomst was niet uitermate groot, maar men zal meer belangstelling hebben voor het computeren dan voor vergaderen, gezien het feit dat het in de middag drukker werd.

De voorzitter opende de vergadering met een welkomwoord aan de aanwezigen, en gaf een toelichting op het ontstaan, reilen en zeilen van de E.P.C.

Het Huishoudelijk reglement voor de regio werd door alle aanwezigen goed-

gekeurd. Verzocht werd of iemand interesse had in de functie van secretaris, maar tot nu toe is er geen vervanger gevonden. Bel dus even naar het afdelings-secretariaat als je er wat voor voelt. (Zie afdelingsadressen) Er waren geen tegen-kandidaten voor de andere bestuursfuncties, dus deze blijven gehandhaafd.

Allen hebben een anquette-formulier thuisgestuurd gekregen. STUUR HET S.V.P. ALSNOG IN als je het per ongeluk vergeten mocht zijn.

Vele reacties zijn al binnen en daar wordt op het ogenblik hard mee ge-

werkt om te zien hoe we de afdeling nog beter kunnen laten functioneren.

De Basic-cursus is nu in volle gang, en erg populair, gezien de steeds groeiende opkomst op de open dagen.

Ze begint steeds om 14.00 uur voor ongeveer een uur.

Tot zover enkele bestuurlijke mededelingen van de afdeling Eemland en wij zien iedereen weer graag terug in "De Oranjeboom" op de derde zaterdag van de maand.

Ondernemings- spel

Wij zijn inmiddels 2 spelronden, ofwel 2 kwartalen, verder in de concurrentie strijd op de markt van chocoladerepen.

Wij hebben zo'n idee dat elke deelnemer met spanning de resultaten van de vorige ronde afwacht om te zien hoe hij gevorderd is. Heeft hij de juiste beslissingen genomen? Heeft hij niet te veel of te weinig geïnvesteerd?

Hoe liggen zijn winstcijfers t.o.v. die van zijn konkurrenten?

Op deze vragen krijgt men antwoord als er voldoende marktonderzoek geïnvesteerd is.

In de geplaatste grafiek zijn de resultaten verwerkt van het laatste halve jaar van de groep van 7 deelnemers waartoe wij behoren. (Wij zijn groep 'G')

Hieruit blijkt dus dat wij het tot op heden nog niet zo slecht hebben gedaan in 1985.

Hoe het spelverloop in z'n totaal is blijkt echter niet. Om daar wat meer inzicht in te krijgen lezen we de verslagen in het NRC-Handelsblad die steeds 2 weken na sluitingsdatum van inzending gepubliceerd worden.

Daaruit valt dan af te leiden dat wij de hoogste winsten behalen!

Dat dit niet "zomaar een spelletje", doch wel degelijk een serieuse praktijk-simulatie is, door velen benut als studiemateriaal, blijkt hier ook weer uit.

Zo heeft het oudste accountantskantoor van Nederland, Moret & Limpberg, meer dan 100 van haar medewerkers in de gelegenheid gesteld om aan dit spel deel te nemen! Zij streeft er naar om, op een wat langere termijn, de dienstverlening uit te kunnen breiden in de richting van advies bij planning en strategiebepaling.

De lessen die dit spel je leren kunnen daar kennelijk toe bijdragen!

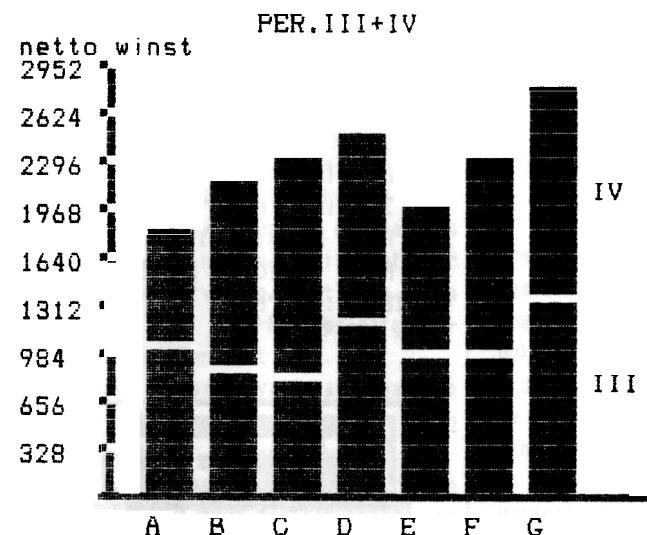
Het is dan ook niet toevallig dat het bij Unilever werd ontwikkeld.

Voor de komende periode hebben we het volgende overwogen:

- het voorafgaande halve jaar heeft wel aangetoond dat onze prijzen (de hoogste in de markt) niet meer omhoog mogen
- dat juist door deze hoge prijzen de verkoop altijd wat onder de verwachtingen ligt
- dat dit tot op heden niet zo erg was omdat wij toch vrijwel alle geproduceerde repen konden verkopen
- dat stilstand achteruitgang betekent en dat dus uitbreiding overwogen dient te worden
- dat dit financieel ook wel kan daardoor wij tot nu toe de winst gebruikt hebben om af te lossen

- dat de tijd gunstig is daar de rente momenteel laag is: (3%)

- dat wij vooruit moeten denken, want wat we NU produceren kunnen we pas in de VOLGENDE periode verkopen. Wij besloten dat ook in deze periode de productie met 1 eenheid uit te breiden en



de volgende periode nieuwe verkopers aan moeten stellen om het geproduceerde ook af te kunnen zetten! Overigens bleek uit de marktonderzoek resultaten dat niet iedereen serieus bezig is. Wat moet dan wel de dieper liggende gedachte zijn bij het ontslaan van alle verkopers en het stopzetten van alle reclame-aktiviteiten bij deeln.A? C is geroutineerd bezig. Dit bedrijf heeft kennelijk al

in periode 3 geïnvesteerd en is met veel verkopers en veel reclame tegen aantrekkelijk lage prijzen gaan verkopen. Zijn Netto winst steeg dan ook aanzienlijk! Hij moet daarom als onze gevaarlijkste tegenspeler beschouwd worden!

Wij hopen met deze overwegingen de lezers die aan dit spel meedoen op

enigerlei wijze van dienst te zijn. Wellicht zijn er lezers die van de groep waarin zij spelen eveneens de resultaten met een grafiek aanschouwelijk willen maken of het met ons niet eens is. Stuur uw commentaar naar postbus 3000, want daaruit zullen wij allen weer kunnen leren.

acv ■

Hoofdnoot

'Hier volgt een goed programma voor de computerclub. Als de huisregel is gezellig aantal mensen omkeren, maar toch programma. Je spelven door tjes en/of spellen. Veel plezier!'

een door elkaar gegeven KNOW HOW, de DR III. t op de juiste reën je een willekeurige rijtje lettertjes en een eenvoudig spelletje o'm eenvoudig proef zelf nog wat mee nbranchen van kleurtjes bijvoorbeeld. , de goede oplossing komt in

el letter:
c-Kopfnus:

```
1000 H=1 :PRINT "EP -  
1001 PRINTCHR$(12):REM BEELDSCHERM  
SCHOON  
1002 PRINT"=====
```

```
1004 PRINT"BEREKENING VAN UW HUID  
OPPERVLAKTE "  
1006 PRINT"=====
```

1008 PRINT:PRINT
1010 FOR H=1 to 40:H\$=H\$+" ";NEXTH
1020 PRINT"GEEF UW LICHAASGEWICHT
OP IN"
1030 PRINT"KILOGRAMMEN EN DE LICHA
AMSLENGTE IN"
1040 PRINT"CENTIMETERS. HET PROGRA
MMA BEREKT"

Hoofdnoot oplossing

```
1050 PRINT"HIERUIT HET LICHAAMS-OP  
PERVLAK IN"  
1060 PRINT" M2.":PRINT  
1070 PRINT"GEWICHT IN KG : "  
1080 INPUTA$:A=VAL(A$)  
1090 IF A<2 OR A>500 THEN 1070  
1100 PRINT"LENGTE IN CM : "  
1110 INPUTB$:B=VAL(B$)  
1120 IF B<30 or B>245 THEN 1100  
1130 C=A*.425  
1140 D=B*.725  
1150 E=.7184*C*D  
1160 F=INT(E+.5)/1E2  
1170 G=LEN(STR(F))  
1180 PRINT LEFT$(H$,31+G)/2  
1190 PRINT "UW TOTALE HUIDOPPER-  
VLAKTE IS ":"F;"M2."
```

Teleaccursus PERSONAL COMPUTER I

Op 10 februari start Teleac met een o.i. zeer interessante cursus van 12 televisielessen en 12 radiolessen over het gebruik van de PC in het bedrijf.

Het gaat hier niet om een opleiding in een programmeertaal of de studie van een bepaald operating systeem, doch veeleer om de systematische aanpak van de vragen die optreden voor-, tijdens- en na de aanschaf van "de computer" in het bedrijf.

Heel bewust heeft een team van deskundigen een cursus opgezet voor het kleinschalig gebruik van de computer om aan te sluiten bij wat vooral in het midden- en kleinbedrijf plaatsvindt ten aanzien van het gebruik van de PC. Deze cursus helpt bij de besluitvorming in de aanschaffingsfase. Niet alleen voor wat betreft de keuze van apparatuur en programmatuur, maar u wordt bijv. ook gewezen op de organisatorische veranderingen die het gevolg zijn van de invoering van de PC in het bedrijfsgebeuren. Ook zal niet iedereen in het bedrijf even enthousiast zijn over de invoering van de PC zodat er rekening gehouden moet worden met de noodzaak dat een, bewuste of onbewuste, weerstand overwonnen moet worden. De 12 lessen uit het fraaie, en rijk met kleurenfoto's geïllustreerde, cursusboek zijn getiteld:

1. Wat kan een PC voor mijn bedrijf betekenen
2. Oneigenlijke argumenten om PC's aan te schaffen
3. Apparatuur en programmatuur
4. Tekstverwerking met de PC
5. De organisatie rond tekstverwerking met een PC



6. Automatisering van de financiële administratie
 7. Toepassingen van paketten voor boekhouding en financiële adm.
 8. Keuze en toepassingen van spreadsheets
 9. Werken met gegevensbestanden
 10. Grafische toepassingen
 11. Datacommunicatie
 12. Geïntegreerde programmatuur
- Bij elke les horen 2 vragen die u voor uzelf moet beantwoorden en waarvoor u dus geen cijfer krijgt!

In de radiolessen volgende op de TV-les worden de antwoorden op deze vragen besproken.

De TV-lessen zijn steeds op maandagavond van 18.25 u tot 18.55 u (met een herhaling op zaterdagmorgen) en de radio-lessen zijn op donderdagavond van 22.25 u tot 22.55 u. De cursus kan, desgewenst, afgesloten worden met een deelname aan 1 of 2 studiedagen in Den Haag.

De prijs van het cursusboek = 85,-.

De kosten voor de studiedagen (op 30 en 31 mei) bedragen 75,- voor 1 dag of 125,- voor beide dagen.

Het boek kan besteld worden door starten op girorek. 54 42 32 t.n.v. Teleac te Utrecht.

In het voorjaar van 1987 zal er een vervolg zijn in de cursus "Personal Computer II".

acv. ■



PROBLEEMLOOS PRINTEN MET EEN MARGRIETWIELPRINTER AAN EEN P2000

Wij hebben op onze basisschool "De Berkhaag" in Herwen (daar waar de Rijn het land binnen komt) in de paasvakantie van 1985 een printer gekocht, want we hadden middels Tekst 2301-2 ontdekt wat tekstverwerking kon betekenen voor onze school en zeker in een tijd van schoolwerkplanontwikkeling.

We waren te werk gegaan volgens de 3 fasen die besproken stonden in het blad "Personal Computer Magazine" en waar ik er nog een paar aan wil toevoegen:

1. lees eerst en bepaal welke printer je wilt.
2. zoek een handelaar/winkel die het betreffende apparaat heeft.
3. eis een demonstratie op school of neem de P2000 mee naar de winkel en verlaat de winkel niet voordat de combinatie het voortreffelijk doet.
4. neem vooral het zelfde "kabeltje" mee als waarmee de combinatie werkte.
5. verlaat ogenblikkelijk de winkel als je merkt dat de handelaar maar even twijfelt aan de P2000 en de bewuste printer.
6. doe dat ook als je merkt dat de handelaar niet op de hoogte is van de ontwikkelingen in het onderwijs en met name in het basisonderwijs (als je een basis onderwijs-man/vrouw bent).

Onderwijswerkgroep Philips Computers

Secretariaat:

Mw. E. van Frankenhuyzen
p.a. P.T.I.S. B.V.
Postbus 16830
2500 BV 's-Gravenhage
Telefoon 070 - 762827

Tekst 2301-2 het "vrij simpel" moet zijn om dat onderstreepprobleem te veranderen zodat je ook met Tekst 2301-2 kunt onderstrepen of dat niet alle tekst onderstreept wordt.

Aangezien de importeur zijn potentiële markt verloren dreigde te zien gaan ging hij hard aan het werk. Het resultaat van het werk was:

1. in de printer-chip werd het onderstreepkommando verwijderd met de toegezegging dat zodra het onderstreepprobleem was opgelost al de JUKI printers gratis gemodificeerd zouden worden.
2. uit experimenten bleek dat de printer geen form-feed gaf als Tekst 2301-2 ingesteld was met 1.5 of 2.0.
3. uit contacten met Philips bleek dat Tekst 2301-2 afgesteld was op de Epson FX 80 en dat de Philips-margrietwielprinter 5700,- kostte.

Inmiddels waren we ook achter het bestaan van MINITEXT in Viditel gekomen.

Ook met Minitext deden zich echter problemen voor.

We hebben daarop kontakt gezocht met Jan Gieles, de auteur van Minitext, en hem de problemen en wensen voorgelegd.

De importeur stelde hem de JUKI 6100 ter beschikking en zie met MINITEXT kan ik alles met de JUKI 6100 doen

waarvoor hij gemaakt is nl. tekstverwerken! Ik kan:

1. onderstrepen, ook al is het onderstreepp commando uit de printerchip verwijderd! Op deze manier kan ik toch Tekst 2301-2 blijven gebruiken met de kinderen (als ze taalstrukturoefeningen doen) en voor de collega's die MINITEXT nu nog te ingewikkeld vinden.
2. alle accenten maken (zie de voorbeelden). Dat is erg belangrijk voor de korrespondentie die we met de printer willen verzorgen. Tevens is het van belang voor de stencils die we aan ouders sturen.

We hebben ontdekt dat we de tekst ook rechtstreeks op typestencils kunnen printen. Dat betekent besparing van geld en tijd: een inbrandstencil kost fl 0.80 en 6 minuten inbrandtijd, een typestencil kost

fl 0.20 en geen inbrandtijd!

3. ik kan mooie vierkanten maken en vertikale lijnen (zie voorbeeld).

TEXT bij de printer te leveren, want maar weinig scholen hebben VIDITEL.

In Microtel stond hoe je in MINITEXT links-boven in de witte balk een regelteller kunt krijgen. Jan Gieles heeft me medegedeeld dat voor bezitters van 16 K machines het vrijwel onmogelijk is om dat te plaatsen.

Voor 32 K machines is te voegen in regel 44:
 poke &H5001,int((U+V+1)mod1000/100) +
 48: poke &H5002,int((U+V+1)mod100/10)
 +48: poke &H5003,(U+V+1)mod10+48:

Omdat er nu probleemloos met margriet wielprinter JUKI 6100 geprint kan worden geeft de importeur de scholen 10% korting! (En het geinteresseerde GPC-lid dan?? red.)

Er blijft nog maar 1 wens over en dat is dat met MINITEXT ook tot 120 regelbreedte gewerkt kan worden (net als met Tekst 2301-2).

Wim Woonings,
regio-koordinator O.W.G.,
regio Doetinchem

De importeur van de JUKI printers is zo enthousiast dat hij nog 2 printers beschikbaar heeft gesteld voor MINITEXT en ook de JUKI matrixprinter met standaard NLQ-kwaliteit. Hij wil ook onderzoeken of het mogelijk is MINI-

Testbestand MINITEXT met JUKI 6100

Printer: JUKI 6100

----- Bijzondere tekens: -----

0-400 S² K³ 22°C privé coöperatie

rijp bêta Adèle 50mA 12½ cm f 14,75
 : : : : : halve spaties B_{max} e_{ax+b} Ax+Bx²+Cx³
 : : : : : hele spaties

START

IF A ≠ B THEN .. door beide ouders.

(regelaafstand 1)

Enkele dubbelslagen: ¥ ¢ ↑ Ø ♀ > Ø + ±

Kerkblad parochie St.MARTINUS Herwen en Aerdt

* * * * *

jaargang:XVIII nr:01 datum: 11 - 01 - 1986

De beide voorbeelden zijn geprint met de chip waarin het onderstreepp commando verwijderd is.

Afdelingsadressen

Vermelde adressen en telefoonnummers zijn UITSLUITEND voor vragen en opmerkingen a.b.t. afdelingsaangelegenheden. Voor vragen omtrent deelnemerschap, kontributie etc. verwijzen wij u naar het Landelijk secretariaat in Horst.

Afdeling Twente
p/a Telefoondistrikt Hengelo
t.a.v. D.C.C.Brand
Postbus 5000
7550 GE Hengelo (geen postzegel)
Tel kantoor: 074-456591
Tel prive: 05407-62722
Viditel pagina #491801#

Afdeling West-Friesland
Tureluurshof 9
1602 NL Enkhuyzen
Tel: 02286-3553

Afdeling Utrecht
Kruisbeklaan 52
3722 TJ Bilthoven
Tel: 030-718318

Afdeling Bergen op Zoom
Moorseweg 5
4681 SM Nieuw Vossemeer
Tel: 016762-2309

Afdeling Alphen a/d Rijn
Haarlemmerstraat 223
2312 BR Leiden
Tel: 01720-76856



Laat ontvangen bericht: Problemen bij betaling per giro.

Afdeling Den Haag
Schubertrode 31
2717 HH Zoetermeer
Tel: 079-515285

Afdeling Oss
M.v.Bourgondiestraat 15
5346 PT Oss
Tel: 04120-35079

Afdeling Eindhoven
Distel 26
5527 KC Hapert
Tel: 04977-5030

Afdeling Rotterdam
Postbus 34012
3005 GA Rotterdam

Afdeling Tilburg
Mascagnistraat 445
5049 BK Tilburg

Afdeling Noord
C.W.Lubbersstraat 13
9643 LA Veendam
Tel: 05987-19589

Afdeling Eemland
Boekweitland 2
3833 CK Leusden

Afdeling Midden Limburg
Parallelweg 1
6082 NA Buggenum
Tel: 04759-1711

Afdeling Venray
Putterstraat 30
5961 VZ Horst
Tel: 04709-1200

Inhoud TRON 3

In de volgende Tron zullen we zeker de volgende artikelen publiceren:

- Diastuurprogramma voor de meerkeuzetoets en de P2000.
- Reaktie op het artikel 'Download op de gemodificeerde Star Gemini-10X'
- Hoofdnoot Tron 7 en oplossing Hoofdnoot Tron 6.
- Ontstaan van de 'DAG-Markten'.
- Meer logo's van de 'DAG-Markten'.
- De P7000-computer.
- Wiskundige functies
- Aanpassing Assembler programma.
- Afdelingsnieuws
- Ondernemingsspel
- Boekbespreking

Wanneer U zelf eens iets wilt laten publiceren verzoeken wij U dit op de volgende manier te doen. Bij voorkeur met behulp van het programma Minitext. Bij gebruik van dit programma zijn er een tweetal zaken waar U op moet letten. Enerzijds: Wanneer Uw artikel zo lang wordt dat het geheugen van een P2000 met een grotere standaard geheugenruimte van 32K wordt overschreden, verzoeken wij U het programma in tweeven te 'knippen'. Anderzijds: Stel de regelbreedte in op 37 karakters. U kunt vervolgens het bandje met copy opsturen naar het redactieadres; zo snel mogelijk na ontvangst zullen wij U het bandje retourneren.

Vlak voor het moment dat de Tron naar de drukker zou gaan bereikte ons nog het volgende bericht:

Inmiddels zullen een aantal deelnemers reeds hun kontributie hebben betaalt. Degenen die per giro trachten hun kontributie te voldoen kregen de mededeling dat het gironummer was opgeheven. Inmiddels is aan het probleem gewerkt en heeft de postbank toegezegd zeker aan het eind van de maand januari het gironummer weer opengesteld te hebben. Dus voor dege-

nen die reeds geprobeerd hebben te betalen:

In de week van 28 januari kunt U (nogmaals?) uw bijdrage zeker kwijt!

