
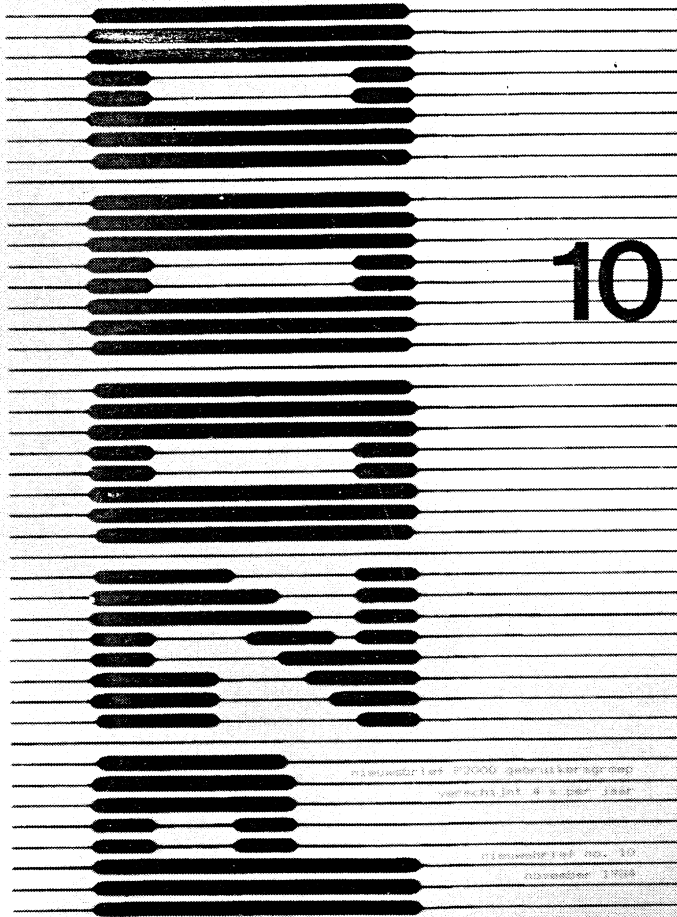
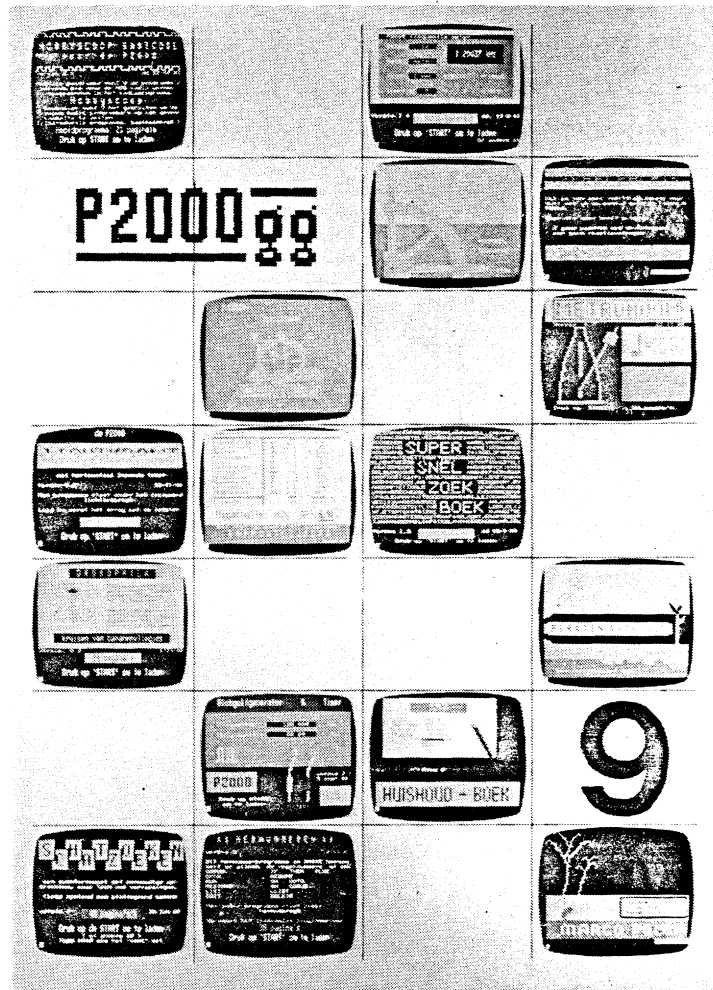
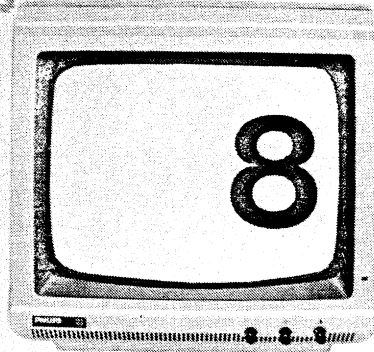


P2000

P2000 



10

nieuwsbrief P2000 gebruikersgroep
 verschijnt 4 x per jaar
 nieuwsbrief no. 11
 februari 1985

11

PHILIPS
 VG8010
 P2000C
 TX

nieuwsbrief P2000 gebruikersgroep
 verschijnt 4 x per jaar
 nieuwsbrief no. 10
 november 1984

<u>Titel</u>	<u>Type informatie</u>	<u>pag.</u>
<u>Algemeen</u>		
Inhoud		1
Bij deze samenvatting	redactioneel	3
<u>Nieuwsbrief 8</u>		
Drie uitbreidingen voor soldeerboutridders	hardware-tip	4
De P2000 komt eraan	artikel	11
Tekstverwerking met de P2000 (1)	programmeerinformatie	12
U spreekt met de P2000 gebruikersgroep	diverse tips	16
<u>Nieuwsbrief 9</u>		
BASIC voor beginners	programmeerinformatie	19
BASIC voor gevorderden	boekbespreking	21
Stringsave	programmeertip	24
Spelen met de cassetterecorder	programmeertip	26
Professionele toepassingen van de P2000	artikel	29
Terugspoelautomaat	hardware-tip	34
Tekstverwerking met de P2000 (2)	programmeerinformatie	36
Mensen en de P2000	interview	41
Assembler in ROM-pack	programma-informatie	44
Cassetteperikelen	hardware-tip	45
Netwerkanalyse en filterkarakteristieken	programma-informatie	46
Overzicht aansluitingen P2000	hardware-tip	47
U spreekt met de P2000 gebruikersgroep	diverse tips	48
<u>Nieuwsbrief 10</u>		
Spring eens uit de BASIC	programmeertips	49
Leesfouten ?	hardware-tip	58
Oeps - correcties op Nwsbr. 8	rectificatie	59
P2000 communicatiemodule	hardware-informatie	60
Dopmoeren op printerconnector	hardware-tip	61
Terugspoel-OUT-omaat	hardware-tip	62
Uitcoderen van I/O-adres	hardware-tip	63
Worteltrekken met dubbele precisie	programmeertip	64
Afdrukken met het (super)snelzoekboek	programmeertip	67
Voorkeur-kleurencombinaties	programmeertip	67
Hoe verkoopt u uw P2000 ?	dialog	68
Toetsen afvangen	programmeertip	71
Piepend toetsenbord	hardware-tip	74
Het koppelen van programma's	programmeertip	75
Aansluiten van P2000 op 2e cassetterecorder	hardware-tip	75
Sneller printen op het scherm	programmeertip	76
Bedreigt de computer het huwelijk ?	persbericht	77
Zelf functietoetsen definiëren	programmeertip	78
Voeding van de P2000 verzwaren	hardware-tip	79

Nieuwsbrief 11

Vereniging Philips Thuiscomputergebruikers	ledeninformatie	81
Stickers printen met (super)snelzoekboek	programmeertip	88
Het einde in zicht ?	programmeertips	89
P2000gg-bestand in Viditel	ledeninformatie	96
Tekstverwerking met de P2000 (3)	programmeertips	97
De fiscus en de P2000	fiscale tip	107
Familiebestanden afdrukken	programmeertips	108
Automatisch toevoegen van BASIC-regels	programmeertip	115
Programma in de knoop ? Uit de knoop !	hardware-tip	119
RAM-disk	programmeertip	121
Informatica 2000	boekbespreking	124

Dit is een samenvatting van de Nieuwsbrieven nummers 8, 9, 10 en 11 die de P2000gg in de jaren 1984 en 1985 heeft uitgegeven. Met uitzondering van Nieuwsbrief 11, die aan alle leden van de P2000gg is toegestuurd, zijn alle Nieuwsbrieven uitverkocht. Het leek de redactie niet zo'n goed idee deze Nieuwsbrieven opnieuw te laten drukken omdat sommige artikelen achterhaald zijn, bij voorbeeld omdat ze betrekking hadden op zaken die toen actueel waren of omdat de inhoud inmiddels op een andere manier is gepubliceerd. Dat geldt bij voorbeeld voor het overzicht van geheugenadressen dat in Nieuwsbrief 8 is gepubliceerd. Bij de PTC is inmiddels het P2000 Adresboekje verschenen, dat in vergelijking met het overzicht in Nieuwsbrief 8 aanzienlijk uitgebreid en verbeterd is.

Omdat er in deze Nieuwsbrieven toch veel artikelen zijn gepubliceerd waarvan de inhoud weliswaar geen eeuwigheidswaarde heeft, maar nog steeds actueel en interessant is, hebben wij besloten, evenals van de nummers 1...7, een samenvatting te maken van de nummers 8...11.

We hebben van de gelegenheid gebruik gemaakt enkele foutjes te corrigeren. Dat wil niet zeggen dat er geen foutjes meer in deze samenvatting staan. Het blijkt vrijwel onmogelijk te zijn artikelen met programma's of delen daarvan foutloos te publiceren. Er hoeft maar ergens een dipswitch verkeerd te staan en de printer zet hier en daar verkeerde tekentjes neer. Maar we hebben ons best gedaan het aantal foutjes tot het uiterste te beperken.

Veel plezier met deze samenvatting.

Son, juli 1985
Rob Geutskens
Voorzitter P2000gg

Hieronder drukken we integraal een artikel af van A. G. Nederlof die drie mogelijkheden geeft om de P2000 te veranderen:

- 1 geheugenuitbreiding door het aanbrengen van andere RAM-IC's
- 2 het inbouwen van een luidspreker
- 3 het tevoorschijn halen van een samengesteld videosignaal, dat kan worden gebruikt om een kleurenmonitor op de P2000 aan te sluiten die geen RGB-, maar een video-ingang heeft

We nemen zonder meer aan dat de heer Nederlof deze wijzigingen grondig heeft geprobeerd en dat ze goed werken als u stipt de aanwijzingen opvolgt. Toch hebben we gearzeld om dit artikel te publiceren. Bij alle drie de wijzigingen, vooral bij de geheugenuitbreiding, moet u ingrijpen in de 'hardware' van de computer. Dat k n tot gevolg hebben dat u schade toebrengt aan de P2000, vooral als u niet veel ervaring hebt met dit soort handelingen. Daarbij komt dat u andere uitbreidingsmogelijkheden, zoals het aansluiten van 'floppy disk'-eenheden, kunt blokkeren.

We hebben echter besloten het artikel toch, met dank aan de auteur, te publiceren. U kunt dan zelf beslissen of u deze ingrepen aandurft. Publicatie betekent niet dat de redactie van de Nieuwsbrief aansprakelijk kan worden gesteld voor de gevolgen van de ingrepen. *Zie pag. 10!*

GEHEUGEN UITBREIDING TOT 42K RAM VOOR MINDER DAN F100,-!!

De beschikbare 16K byte RAM in onze P2000 wordt opgebouwd uit 8 maal 16K bit RAM ic's (de 4115). Hoewel deze ten tijde van het ontwerpen van de P2000 de beste keus waren, zijn nu reeds enige tijd de 4164, oftewel 64K bit ic's in de handel. De gestaag zakkende prijs van dit ic maakt het nu mogelijk voor een relatief gering bedrag onze oude geheugen ic's hierdoor te vervangen. Dit is mogelijk doordat de nieuwe ic's voor een groot gedeelte pin-compatibel zijn (het heeft een adreslijn extra nodig en een enkelvoudige 5V voeding) en doordat ze veel minder stroom gebruiken dan het oude type. Aangezien dit type RAM ic's met gemultiplexte adreslijnen werkt, moeten de benodigde extra adreslijnen A14 en A15 samen de nieuwe adreslijn verzorgen om de volle 64K (in principe) aan te kunnen spreken. Gelukkig is meneer Philips zo vriendelijk geweest om zo 'n benodigde multiplexer port vrij te laten. Verder moet de P2000 er door middel van zijn adres decoder erop attent gemaakt worden dat hij het extra geheugen niet op het extension bord moet zoeken maar op de plaats van de originele 16K. Helaas is maar een deel van het nieuw ingebrachte geheugen adresseerbaar, (de 22K voor basic, monitor en video gaan af van het totaal beschikbare adresbereik van 64K), maar er wordt aan gewerkt om dat probleem te verhelpen. Dit alles houdt in dat we werkelijk alleen maar de geheugen ic's hoeven te vervangen (allen gemonteerd op sokkel), er een nieuw geprogrammeerde adres decoder moeten insteken, en een aantal nieuwe verbindingen moeten maken. De geheugen ic's heb ik voor 9,90DM bij de firma MSP (Florsheimerstrasse 5, 6234 HattersheimZ, Westduitsland) besteld en een decoder ic moet iets van 1,- kosten. Alles bij elkaar is men dus minder dan honderd gulden kwijt voor een volledig opgevuld geheugen!!

GEHEUGENINDELING

De geheugenindeling wordt bepaald door de adres decoder (de 62s123, een 32*8 ROM ic), welke de hoogste 5 adreslijne (A11-A15) decodeert volgens het opgeslagen programma en dan als resultaat 8 signalen afgeeft die de diverse binnen de P2000 bestaande geheugenblokken aktiveren. In tab. 1 is afgebeeld welke signalen op welke pennen de adresdecoder afgeeft.

TABEL 1

Door de adress decoder bestuurdde signalen

+	-----		signaal		pin		bit		-----	+
	MBEN	-	N		1		1		-----	
	RAMS1				2		2		-----	
	VIDS	-	N		3		3		-----	
	ROMS1	-	N		4		4		-----	
	ROMS2	-	N		5		5		-----	
	CASS1	-	N		6		6		-----	
	CASS2	-	N		7		7		-----	
	RAMS2				9		8		-----	

In tab 2 is het opgeslagen "programma" zoals dat origineel door Philips geleverd wordt weergegeven.

TABEL 2

Origineel programma van de adres decoder

+	-----			-----			-----			-----			-----			-----			-----		+	
	geheugenbereik			signaal output			type geheugen			-----			-----			-----			-----			
	hex		dec		hex		binair		-----		-----			-----			-----			-----		
	0000-		0- 2 k		74		01110100		monitor ROM		-----			-----			-----			-----		
	07FF-		2- 4 k		74 *		01110100		monitor ROM		-----			-----			-----			-----		
	1000-		4-12 k		50		01011100		basic ROM		-----			-----			-----			-----		
	3000-		12-20 k		30		00111100		basic ROM		-----			-----			-----			-----		
	5000-		20-22 k		79		01111001		video RAM		-----			-----			-----			-----		
	57FF-		22-24 k		7D		01111101		niet gebruikt		-----			-----			-----			-----		
	6000-		24-40 k		7E		01111110		basic RAM		-----			-----			-----			-----		
	A000-		40-64 k		FD		11111101		extension RAM		-----			-----			-----			-----		
	* voor de modellen met 2*2716 ROM gelat nier			de volgende codering:			-----			-----			-----			-----			-----			
	07FF-		2- 4 k		6c		01101100		monitor ROM		-----			-----			-----			-----		

Neem nota van het feit dat er twee modellen zijn, een waarbij het 4K monitor ROM over twee 2716 verdeeld is, en het andere, meer voorkomende model waarbij dit geheugen in een 2632 opgeslagen is en waarbij men dus een vrije sokkel op het CPU-bord aantreft. Gaan we nu een geheugenblok uitbreiden, door b.v. de basic RAM van 16K tot 64K op te voeren, dan moeten ook meer geheugen adressen in die richting leiden. Dit doen we door b.v. de adressen die normaal naar het extension bord geleid worden nu naar het cpu bord RAM te voeren. Ook kan de niet gebruikte adresruimte tussen 5800-6000H hiervoor gebruikt worden. Deze zijn dan weliswaar niet door basic aanspreekbaar maar kunnen dienst doen om een machinetaal routinetje in te plaatsen.

TABEL 3

Te wijzigen waarden voor 42k vrije RAM

geheugenbereik		signaal output		type geheugen
hex	dec	hex	binair	
57ff-	22-24 k	7E	01111110	nieuwe RAM
A000-	40-64 k	7E	01111110	nieuwe RAM

BOUWHANDLEIDING GEHEUGEN-UIBREIDING

Benodigdheden:

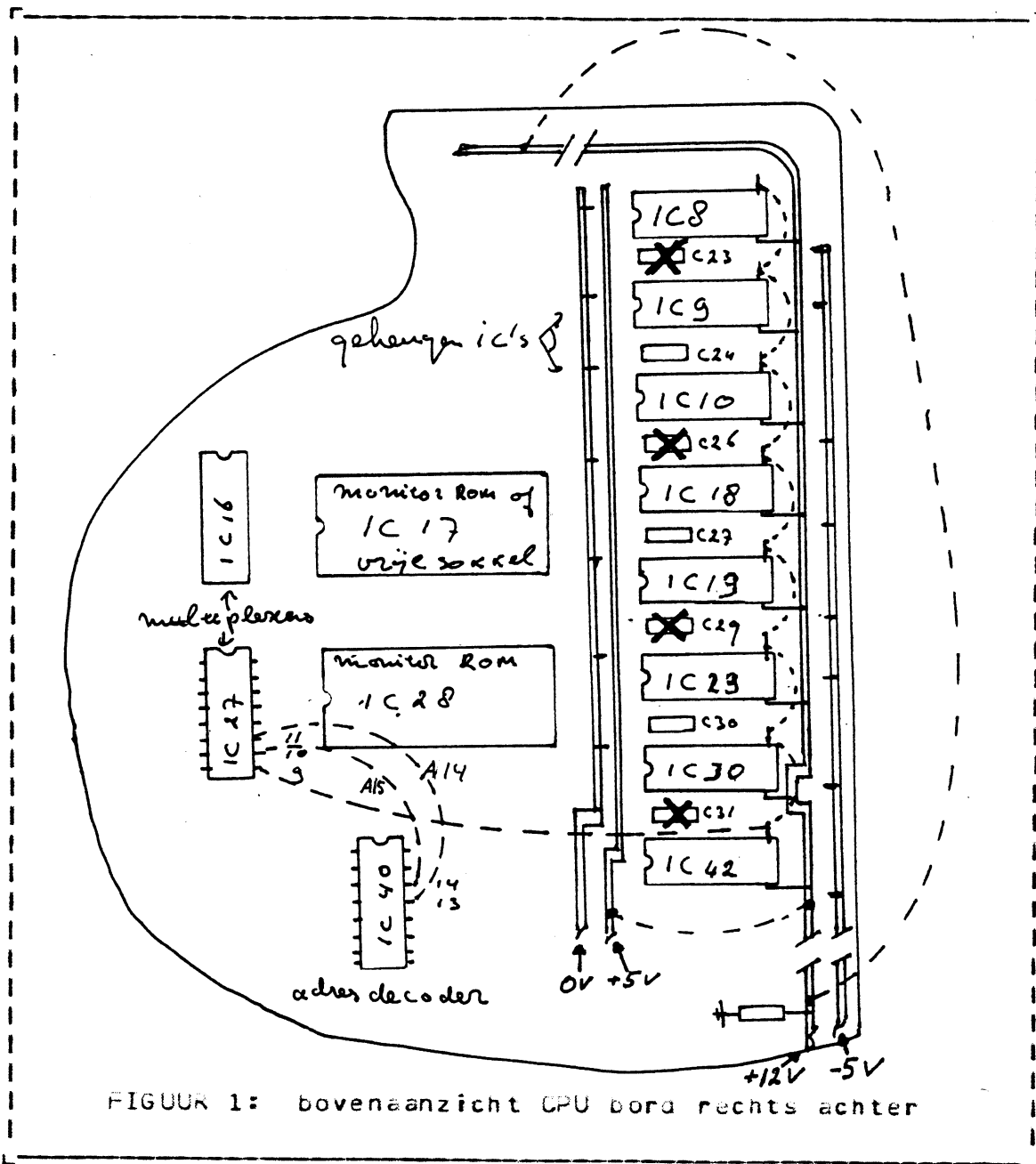
1. 8 * 4154 of vergelijkbare types 64k dyn. RAM,
2. Een nieuw geprogrammeerde adresdecoder,
3. Wat fijn wire wrap draad om de nieuwe verbindingen te leggen.

Te volgen werkwijze:

1. Schroef de P2000 open en demonter het CPU bord.
2. Verwijder de 4116's (8*) en de adres decoder (zie fig. 1)
3. Verwijder de 4 aangekruiste condensatoren (nrs 23,26,29,31).
4. Verbreek op de drie AANGEGEVEN plaatsen de +12V en -5V leidingen.
5. Verbindt de uiteinden van de +12V leiding
6. Verbind het verbroken gedeelte van de oude -5V leiding (dat NAAR de geheugen ic's gaat) met de +5V leiding

7. Verbind de adreslijnen A14, A15 van de adres decoder met de multiplexer als volgt:

adresdecoder	<=>	multiplexer
13	<=>	11
14	<=>	10



8. Monteer nu de nieuwe geheugen ic's in hun sokkels maar buig pin 9 haaks om zodat deze buiten de sokkel valt
9. Soldeer vervolgens een draad over al deze 8 pinnen nr 9 (dit is de nieuwe adreslijn) en verbind deze draad met pin 9 van de decoder (bovenzijde bord). Deze verbinding is het makkelijkst te leggen door de wirewrap draad met isolatie en al eenmaal om iedere ic-

poot te wikkelen en vervolgens vast te solderen. De isolatie smelt dan weg.

10. Steek de nieuwe adres decoder in de sokkel.
11. Ter controle kunnen de volgende testen worden uitgevoerd:
 - a) De Z-80 CPU moet met de multiplexer de volgende verbindingen hebben:

Z-80	<=>	multiplexer	
5	<=>	10	(A15)
4	<=>	11	(A14)
 - b) Controleer of verbrekingen werkelijk geen contact maken
 - c) Controleer of de nieuwe verbindingen alleen de bedoelde punten verbinden!!
12. Monteer het CPU bord en sluit de stroom en monitor aan. Als alles goed verlopen is krijgt u op het beeldscherm
39542 bytes vrij. Nog 2K extra kunt u in het adresbereik van 5800-6000 vinden.

2K USER EPROM

De bezitters van het model met de vrije ROM sokkel hebben nog een andere mogelijkheid om de 2k geheugenruimte tussen 5800-6000h te gebruiken.

TABEL 4

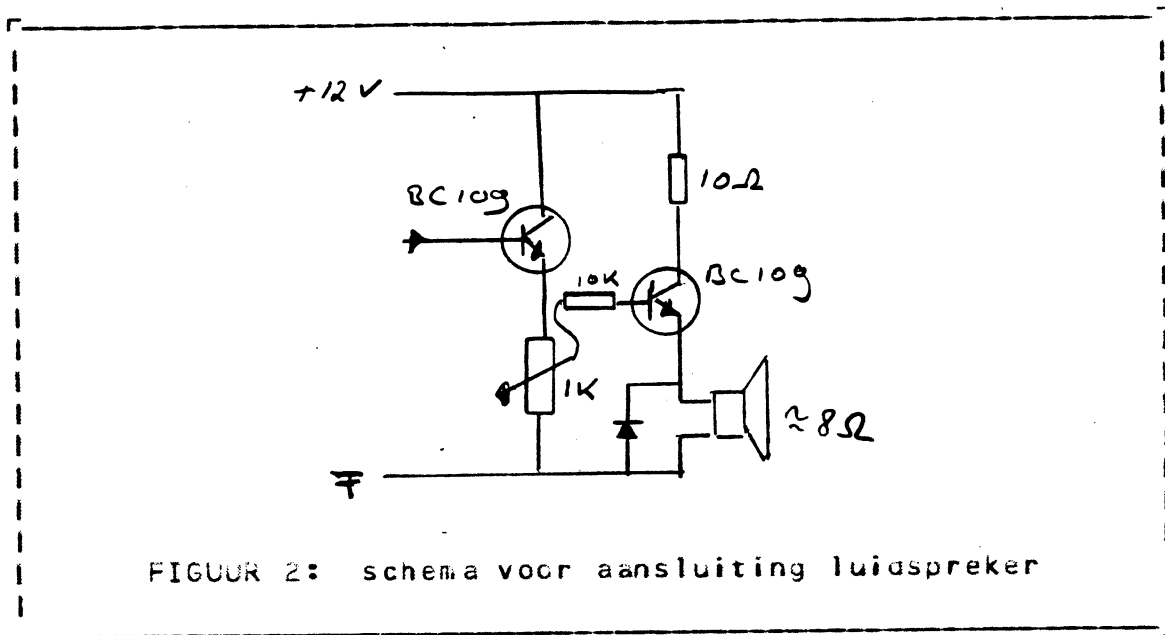
Te wijzigen waarden voor 2k ROM in vrije sokkel

geheugenbereik		signaal output		type geheugen
hex	dec	hex	binair	
57ff-	22-24 k	6c	01101100	ROM in sokkel

Door namelijk hier een (zelf geprogrammeerde) EPROM (2K) in te stoppen en de adresdecoder te herprogrammeren zoals in tab. 4 is aangegeven heeft men een eigen permanent aanwezig stukje machinetaal tot zijn beschikking. Hierin kan men b.v. een parallel interface of een tweeweg seriële interface onderbrengen. Natuurlijk is het ook mogelijk om i.p.v. deze adressen andere niet gebruikte adressen naar deze ic sokkel te leiden door de decoder navenant te programmeren.

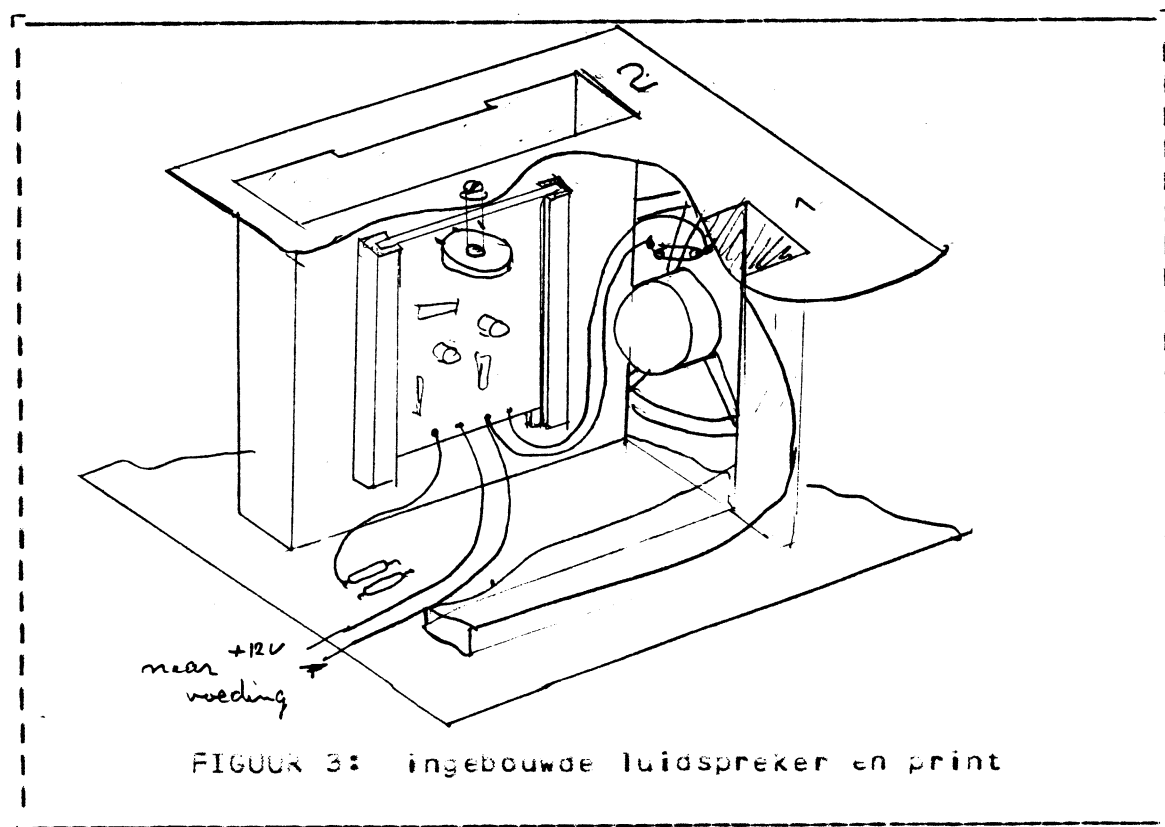
INBOUWEN VAN EEN LUIDSPREKER

Voor degenen die een monitor hebben (zonder geluid) of degenen die een TV hebben waarvan de geluidskwaliteit veel te wensen over laat, bestaat de mogelijkheid om met een vrij eenvoudig schakelingetje (zie fig 2) een kleine luidspreker in de P2000 in te bouwen.



FIGUUR 2: schema voor aansluiting luidspreker

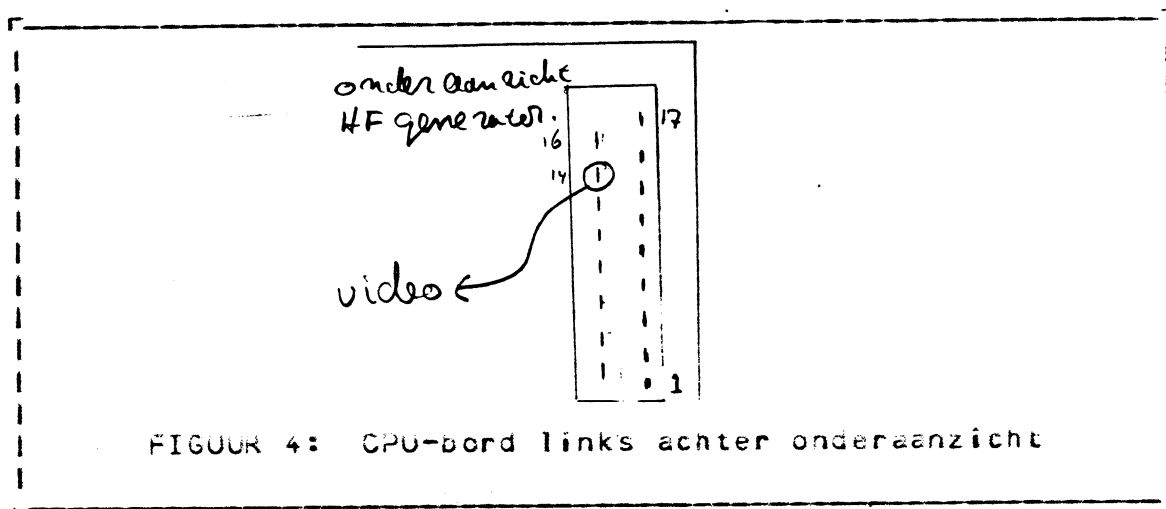
Het printje met de potmeter erop gebouwd kan onder de houder van de twee slots worden ingebouwd (zie fig 3), zodat met een gat in de houder de potmeter bereikbaar is. Ook de luidspreker kan aan de houder vastgemaakt worden. Het voortgebrachte geluid is zeer zuiver.



FIGUUR 3: ingebouwde luidspreker en print

KLEUREN VIDEOSIGNAAL

De P2000 heeft geen kleuren video uitgang. Dit is lastig als je een kleurenbuis hebt met wel een video ingang maar geen RGB plug. Het blijkt echter dat binnen de HF generator wel degelijk zo'n signaal aangemaakt wordt.



FIGUUR 4: CPU-bord links achter onderaanzicht

Deze komt naar buiten via pin 14 aan de onderzijde van het CPU-bord (fig 4) en is rechtstreeks aan te sluiten op een video ingang en geeft daarmee dus een perfect beeld (maar natuurlijk geen geluid, zie boven).

TOT SLOT

Alle hier boven beschreven schakelingen zijn door mij ontworpen en getest. Mocht je moeilijkheden hebben bij het aanbrengen van een van deze zaken dan kun je me bellen of schrijven.

VEEL SUCCES.

Arthur Nederlof

VOORDAT U BEGINT AAN DEZE UITBREIDINGEN:

zie het artikel Oeps op pag. 59 van deze samenvatting !!!

Na de P2000T, de P2000M en de P2500 wordt de Philips-stal nu verrijkt met een nieuw raspaard: de P2000C, een zeer complete machine die in drie standaarduitvoeringen wordt geleverd en vooral mikt op de wat meer professionele toepassingen. De P2000C heeft een los toetsenbord, dat als deksel kan worden gebruikt voor de hoofdkast. Met opgeklapt deksel is de computer gemakkelijk mee te nemen; Philips noemt hem dan ook 'mobiel', wat iets minder gemakkelijk klinkt dan 'draagbaar'.

Op het toetsenbord na is alle hardware ondergebracht in de hoofdkast. En dat is niet niks: de microcomputer zelf, twee schijfeneenheden, een monochrome 9"-monitor en de voeding voor dat alles. Er is zelfs nog een vak voor het opbergen van enkele diskettes.

De P2000C heeft als microprocessor een Z80A, die met een klokfrequentie van 4 MHz werkt. Er is een tweede Z80A die de besturing van het beeldscherm voor zijn rekening neemt, zodat de eerste Z80A zijn handen vrij houdt voor het echte computerwerk. Daardoor is de computer erg snel.

Standaard heeft de machine een gebruikersgeheugen van 64K, uit te breiden tot maar liefst 320K. Daarbij komt nog eens een apart videogeheugen van 32K, dat het mogelijk maakt de $256 \times 252 = 64\ 512$ beeldpunten van het scherm afzonderlijk te besturen, bij voorbeeld om zeer fijne grafieken te tekenen.

De computer heeft drie aansluitingen voor communicatie met de buitenwereld: een paralleluitgang (Centronics), een seriële RS232-uitgang (bekend van het T-model) en een aansluiting voor twee extra 'disk drives'. Bovendien heeft de P2000C een zogenaamde SASI-controller aan boord voor het aansluiten van vaste schijven. Daarmee kunnen gigantische hoeveelheden gegevens worden opgeslagen.

Er zijn drie modellen:

- P2010-1 64K RAM (uit te breiden tot 320K), 32K video-RAM, 2 'floppy disk drives' à 160K.
- P2010-2 Als de P2010-1, maar uitgebreid met een IEEE-bus waarmee allerhande instrumenten kunnen worden bestuurd en uitgelezen. Deze uitvoering is vooral bedoeld voor technisch-wetenschappelijke toepassingen.
- P2012 Als de P2010-1, maar met 'drives' die de diskettes aan beide kanten met 80 sporen kunnen beschrijven (in plaats van 40 sporen aan één kant). Daarmee is de 'on-line'-opslagcapaciteit 640K per diskette, in totaal dus 1,28M (om precies te zijn: 1 310 720 bytes).

Programmatuur

Elke computer is zo sterk als zijn 'software'. De P2000C (dus alle drie de modellen) wordt standaard geleverd met het besturingssysteem CP/M. Dit 'operating system' geeft toegang tot een enorme bibliotheek van toepassingsprogramma's. Een Nederlandse handleiding hoort er ook bij. Er kunnen 'gebundelde' programmapakketten worden meegeleverd, bij voorbeeld voor allerhande kantoortoepassingen.

Verder is er keus uit:

- * P-system (UCSD) - een menu-georiënteerd besturingssysteem, waarvoor onder de naam Profi-On een hele serie programma's is ontwikkeld.
- * BASIC RTS (Run Time System) - onder CP/M.
- * BASIC DS (Development System) - voor het ontwikkelen van BASIC-programma's onder CP/M.
- * Advanced BASIC Interpreter - eveneens onder CP/M.
- * PASCAL-compiler (- draait onder P-system (UCSD)
- * FORTRAN-compiler - idem

Prijzen

De prijs van de P2000C ligt, afhankelijk van het model en de meegeleverde programmatuur, tussen f 6500,- en f 7950,-. Wie had dat vijf jaar geleden durven dromen ?

Er zijn ontzettend veel manieren om een teken op papier af te drukken. Schrijfmachinefabrikanten over de hele wereld hebben zich beijverd om al die manieren in praktijk te brengen. De fabrikanten van microcomputers hebben zich daarentegen uitgesloofd computers te maken waarop geen enkele printer zonder meer kan worden aangesloten.

Wie de vergissing heeft gemaakt zomaar een printer te kopen in de veronderstelling tekstverwerking te kunnen bedrijven, zal het bovenstaande kunnen bevestigen. Vooral als men eisen stelt aan die teksten. De meeste printers slagen er wel in om in combinatie met bij voorbeeld de P2000 wat letters op papier te krijgen, maar wie de puntjes op de i wil hebben zal meestal worden teleurgesteld. Die zal ontdekken dat hij met geen mogelijkheid de Franse accènten en de tréma afgedrukt krijgt; dat hij van die 128 verschillende schriftsoorten er maar één kan gebruiken; dat het niet mogelijk is de linkerkantlijn in te stellen of dat hij om de haverklap een "Printer error" krijgt.

Ik wil in een aantal artikelen wat informatie geven over tekstverwerking met de P2000. Niet omdat ik de wijsheid in pacht heb, maar omdat ik in mijn beroep dagelijks met tekstverwerking te maken heb en redelijk op de hoogte ben van de eisen die aan een goede tekstverwerker worden gesteld. Om te bewijzen dat het kan, is deze tekst op de P2000 geschreven, ook al was het makkelijker geweest de professionele tekstverwerker te gebruiken die onder handbereik staat.

Uiteraard houd ik mij warm aanbevolen voor suggesties.

Iets over printers

In de inleiding heb ik gezegd dat er ontzettend veel manieren zijn om letters op het papier te krijgen. Wie dat niet gelooft moet maar eens een kijkje gaan nemen in het schrift- en schrijfmachinemuseum in Tilburg.

De oudste manier grijpt terug op de ontdekking door Laurens Janszoon Coster uit Haarlem (of Gutenberg uit Mainz) die, zoals bekend, het drukken heeft uitgevonden. Daarbij maakt men gebruik van tekens die verhoogd zijn aangebracht op..... noem maar op: hamertjes ('types'), een cilinder, een bol, een schijf. Wie het genoemde museum bezoekt zal tot zijn verbazing ontdekken dat al die manieren al in het begintijdperk van de schrijfmachine zijn uitgedacht en uitgeprobeerd. De bekende bolkopschrijfmachine waarmee IBM aan het einde van de zestiger jaren op de markt kwam berustte dan ook op oude, en inmiddels vergeten principes. De verdienste van de Amerikaanse computerfabrikant is niet dat hij een revolutionaire manier heeft bedacht om snel en met hoge kwaliteit tekens af te drukken, maar dat hij een oud principe nieuw leven heeft ingeblazen door gebruik te maken van moderne technieken (tussen haakjes: deze tekst is afgedrukt door zo'n tot printer omgebouwde IBM-machine).

Wat later kwamen de machines met margrieten, waarbij de tekens zijn aangebracht op een schijf met tongen. Deze "daisy wheel printers" kunnen sneller zijn dan machines met een golfbal.

Machines die met voorgevormde tekens werken, die in een keer in hun geheel worden afgedrukt, noemt men "impact printers".

Impact printers zijn meestal 'gewone' schrijfmachines waaraan wat elektromagneten en elektronica zijn toegevoegd, om ze met een computer te kunnen besturen. De meeste hebben een toetsenbord, zodat ze ook met de vingers kunnen worden betoetst. Ze hebben doorgaans een uitstekende afdrukkwaliteit. Een nadeel is dat de meeste niet erg snel zijn. Machines met bolletjes typen 15 tot 20 tekens per seconde; machines met schijven komen tot 40 of 50 tekens per seconde, maar dan houdt het zo ongeveer wel op.

Matrixprinters

Dat ligt anders bij de tweede categorie printers, de mozaïek of matrixprinters. Deze zijn ontwikkeld omdat men een machine zocht die veel sneller de grote hoeveelheden computergegevens kon afdrukken dan de traditionele impact printers, en die toch betaalbaar zou zijn (er zijn impact printers waarbij alle tekens zijn aangebracht op een ketting of een trommel, die razendsnel in de rondte gaat; deze kunnen honderden regels per minuut afdrukken, maar kosten een vermogen).

Een matrixprinter bouwt elk teken op uit een aantal puntjes. Bij de eerste typen had je waarlijk geen loep nodig om dat te zien, maar zelfs bij de moderste matrixprinters kun je nog goed zien dat het om spijkerschrift gaat. Een aantal artikelen in deze Nieuwsbrief is afgedrukt met zo'n matrixprinter; om precies te zijn: de EPSON FX-80. Deze is ruim tienmaal zo snel als de IBM-machine: 160 tekens per seconde.

Matrixprinters zijn per definitie elektronische machines. Ze zijn uitgerust met een kop die snel langs het papier beweegt. In de schrijfkop bevinden zich 7, 8, 9 en soms nog meer dunne naalden boven elkaar, die exact op de juiste momenten naar buiten schieten om een enkel puntje op het papier te zetten. Het spreekt vanzelf dat dit allemaal zeer goed 'getimed' moet gebeuren. Een regel van 80 tekens, die in een halve seconde wordt afgedrukt, bestaat uit enkele duizenden puntjes, die allemaal precies op de juiste plaats moeten staan. Alleen een elektronisch systeem is in staat dat perfect te organiseren. Matrixprinters hebben, zover ik weet, nooit een toetsenbord. Ze zijn dus niet als gewone schrijfmachine te gebruiken. Ze moeten altijd door een computer worden bestuurd.

U zult zelf moeten kiezen

Voordat u een printer aanschaft, zult u de vraag moeten beantwoorden wat u met die printer wilt doen. Wilt u er uitsluitend programma's mee uitdraaien, wilt u snel grote hoeveelheden gegevens afdrukken (bestanden, boekhoudgegevens en dergelijke), of wilt u er correspondentie mee voeren?

Om u te helpen bij die keus zullen we eerst de belangrijkste eigenschappen van de twee in aanmerking komende printertypen op een rijtje zetten.

Voordelen impact printers

- Uitstekende afdrukkwaliteit. Machines met een bolletje scoren iets hoger dan machines met een letterschijf.
- Bolletjes en schijven kunnen worden verwisseld, waardoor in principe keus bestaat uit verschillende lettertypen.
- Veel impact printers hebben een toetsenbord en zijn dus ook als gewone schrijfmachine te gebruiken.

Nadelen impact printers

- De afdruksnelheid is beperkt tot 10 à 50 tekens per seconde.
- Een goede impact printer is nogal prijzig, maar dat lijkt te veranderen (zie de voordelen van matrixprinters).
- Ze hebben meer bewegende delen dan een matrixprinter en zijn daardoor kwetsbaarder.
- Het aantal verschillende tekens is beperkt tot 88 of 96 voor een bolletje en 96 of 104 (soms) voor een letterschijf.
- Veelal ontbreekt een aantal tekens die u nodig hebt om programma's uit te draaien. Die kan ik hier dan ook niet laten zien, dus ik zal ze moeten omschrijven: hekje, aapje, groter dan en kleiner dan.

Nu zijn natuurlijk de voordelen van de matrixprinter de nadelen van de impactprinter en omgekeerd, maar we willen er toch iets meer over zeggen.

Voordelen matrixprinter

- Hoge afdruksnelheid; doorgaans in de orde van 80 tekens per seconde, soms 160 t/s. Om snelheid te winnen beweegt het schrijfmechanisme beurtelings van links naar rechts en van rechts naar links (dat scheelt in de teruglooptijd); maar moderne letterschijfprinters doen dat ook.
- Soms kan de snelheid worden teruggebracht om een betere afdrukkwaliteit te krijgen, zij het dat die voorlopig nog achterblijft bij die van impact printers.
- Matrixprinters kunnen doorgaans meer verschillende tekens afdrukken dan impact printers; niet alleen de 'computer'-tekens (aapje, hekje e.d.), maar ook grafische tekens zoals de Viewdata-blokjes. Dat hangt af van wat de karaktergenerator van de printer in huis heeft.
- Bij sommige nieuwe matrixprinters kan de gebruiker zelf karakters definiëren, bij voorbeeld Griekse letters voor technische teksten (omega, mu e.d.). U zou zelfs Cyrillische tekens kunnen definiëren om Russisch te kunnen afdrukken.
- Met behulp van de 'Bit Image Mode' kan in principe elk puntje van een pagina afzonderlijk worden aangesproken. Daarmee is het mogelijk afbeeldingen en grafieken af te drukken. Dat vereist wel enige stuurmanskunst om de printer goed te besturen.
- Matrixprinters hebben weinig bewegende delen. De schrijfkop kan meestal gemakkelijk worden vervangen als hij versleten is.
- Een goede matrixprinter was tot voor kort goedkoper dan een goede impactprinter, hoewel de prijzen elkaar beginnen te naderen. De Hermes Top-Tronic 15, een machine met letterwiel die door leden van de P2C2 geschikt is gemaakt voor de P2000, komt voor circa f 1500,-, en dat ook zo ongeveer de prijs van een ordentelijke matrixprinter.

Nadelen matrixprinters

- Het blijft spijkerschrift. De eerste matrixprinters, waarvan je de sporen nog wel eens vindt, hadden te weinig afdruknaaldjes en kenden dan ook geen 'true descenders' (dat zijn de pootjes van de p, de q, de j en de g die onder de regel horen uit te steken, maar dat niet deden. Let daarop als u een matrixprinter koopt).
- Het aantal lettertypen (dus niet het aantal tekens per lettertype) is beperkt.
- Een matrixprinter is vrijwel nooit als schrijfmachine te gebruiken doordat hij meestal geen toetsenbord heeft.

Nog een paar dingen om op te letten

Als u een weloverwogen keus hebt gemaakt tussen een impact en een matrixprinter, kunt u nog niet naar de dealer stappen om er een te kopen. De machine zal aan enkele voorwaarden moeten voldoen als u hem met de P2000 wilt gebruiken. Om te beginnen heeft de P2000 in standaarduitvoering alleen een seriële uitgang (de V24- of RS-232-connector met 25 pennen aan de achterkant), en zijn de meeste printers uitgerust met een zogenaamde Centronics-ingang, een parallelingang die de bitjes met acht tegelijk wenst te ontvangen (de V24-uitgang stuurt de bitjes serieel ofwel sequentieel, da wil zeggen na elkaar).

U kunt nu twee dingen doen. U kunt de printer laten uitrusten met een V24-interface. Meestal kan dat wel, maar het kost geld; enkele honderden gulden meestal.

U kunt ook een paralleluitgang op uw P2000 maken, bij voorbeeld door een module in sleuf 2 te steken (zie het overzicht van uitbreidingsmogelijkheden in deze Nieuwsbrief). Dat heeft het voordeel dat u de V24-uitgang vrij houdt voor andere dingen (Viditel bij voorbeeld), en het nadeel dat u uw programma's zo moet inrichten dat u de printer via die paralleluitgang kunt besturen. Ik moet bekennen dat ik niet zo gauw zou weten hoe dat precies moet, maar dat hoor ik nog wel. In elk geval kan het wel, al kost dit ook enkele honderden gulden.

Als dit u te veel moeite is, kunt u natuurlijk de veilige weg bewandelen en een van de door Philips aangeboden printers aanschaffen; de matrixprinter of de letterwielprinter. Die kunt u zonder meer aansluiten op uw P2000, zonder u te bekommeren om in- en uitgangen en interfaces. U bent dan zeker niet veel duurder uit, maar u bent wel beperkt in uw keus.

Al met al hebben we het tot nu toe alleen maar over printers gehad, maar dat zijn noodzakelijke attributen als u tekst zwart op wit wilt afdrukken. In het volgende artikel zal ik ingaan op de manier waarop u teksten kunt invoeren. U zult ontdekken dat u daarvoor geen dure extra's nodig hebt, maar dat u ook met de BASIC-NL heel aardig tekstverwerking kunt bedrijven.

Rob Geutskens

Regelmatig worden Rob en ik aan de telefoon geconfronteerd met terugkerende vragen van P2000-gebruikers. In dit artikeltje wil ik enkele van deze vragen eruit lichten en aldus trachten te beantwoorden.

- Kan ik met de P2000 80 tekens op een regel printen?

Ja, dat kan. Hoewel uw beeldscherm breedte in het algemeen 40 tekens breed is, en u bij de "print-scherm"-opdracht dus 40 tekens op een regel zult krijgen is het mogelijk om met een "lprint"-opdracht 80 tekens op een regel te printen. Afhankelijk van uw printer kunt u de P2000 instellen tot maximaal 256 tekens per regel.

NB: Met het "Familiegeheugen" is het niet mogelijk meer dan 40 tekens per regel te printen.

- Welke printers kan ik aan de P2000 aansluiten?

Het antwoord op deze vraag vindt u in deze Nieuwsbrief in het artikel over tekstverwerking met de P2000.

U zult in het algemeen de minste problemen hebben als u gebruik maakt van een Philips P2123-printer of een Epson-printer met een serieel interface.

- Kan ik ook een externe (audio-) cassetterecorder aan de P2000 aansluiten?

U kunt een externe (audio-) cassetterecorder aansluiten aan de P2000 door gebruik te maken van het Basicode-vertaalprogramma en de Basicode-aansluitkabel. Deze aansluitkabel vindt u terug in het artikel over uitbreidingen voor de P2000 onder het hoofd AVO-techniek.

- In onze krant stond dat Viditel 500 gulden entreegeld vraagt. Waarom distribueert Philips de programma's dan niet via een goedkoper medium?

U leest waarschijnlijk een Brabantse krant. Het genoemde entreegeld is namelijk onjuist. Om Viditel te kunnen bekijken hoeft u enkel het maandelijkse abonnement van fl. 11,90 te betalen. De PTT geeft u dan geheel gratis een modem dat u met de aansluitkabel uit de doos van de P2000 aan de P2000 kunt aansluiten. Als u contact maakt met Viditel betaalt u naast het maandelijkse abonnementsgeld alleen telefoonkosten en (afhankelijk van uw woonplaats en de tijd waarop u belt) een gering bedrag voor toegangstijd.

- Kan ik ook game-paddles aansluiten aan de P2000?

Ja, dat kan. U kunt game-paddles kopen bij Sanecal (zie artikel over uitbreidingen voor de P2000).

Hierbij is een kritische kanttekening op zijn plaats:

Momenteel is bij P2C2 een interface in ontwikkeling waarbij de Philips-game-paddles aan de P2000 aangesloten kunnen worden. Een en ander ziet er erg veelbelovend uit en zal vermoedelijk een nieuwe standaard voor spel-programma's inluiden (waarmee de huidige game-paddles niet werken).

- Kunt u mij adressen van leden uit mijn stad of dorp doorgeven?

Ja, dat kan, maar in verband met de privacy van de leden doen we dat niet. Als u contacten met P2000-gebruikers uit uw regio wilt leggen kunt u (als u lid bent van de P2000gg) een contact-advertentie laten opnemen in de Nieuwsbrief.

- Hoe kom ik aan lege cassettes voor de P2000?

De kwaliteitscontrole door Philips op de cassettes is erg streng. Mede hierdoor is onlangs een grote partij cassettes afgekeurd. Het gevolg daarvan is dat Philips (tijdelijk) leveringsproblemen heeft m.b.t. cassettes voor de P2000. Afwachten luidt het devies.

- Ik heb enkele weken geleden programma's besteld bij de P2000gg maar tot op heden heb ik nog niets ontvangen. Wanneer kan ik die programma's verwachten?

Mede door eerder genoemde problemen van Philips kan ook de P2000gg slechts mondjesmaat aan cassettes komen. Zodra er weer nieuwe bandjes binnenkomen worden deze door Ton Hilgersom beschreven en verzonden.

We doen ons best om uw programma's zo snel mogelijk te leveren. Vergeet niet dat wij hobbyisten zijn en in onze vrije tijd aan uw wensen proberen te voldoen.

- Ik ben van beroep en zoek daarvoor speciale programma's. Kunt u mij daaraan helpen?

Nee. Wij zijn een vereniging van, voor en door hobbyisten. Voor professionele toepassingen verwijs ik u gaarne naar Philips PTIS, Maanweg 156 in Den Haag, tel (070) 76 27 66, waar u een dealerlijst kunt aanvragen van dealers die al wat langer de P2000 verkopen en er wat meer ervaring mee hebben.

- Kan ik met de P2000 ook externe apparaten schakelen.

Ja, daarvoor is een interface verkrijgbaar bij Sanecal.

Daarnaast is het mogelijk om voor onderwijstoepassingen met de P2000 een Philips Videorecorder te besturen (alleen nieuwe types). Meer informatie daaromtrent kunt u krijgen bij de coördinator hardware-documentatie van de P2000-onderwijswerkgroep, Ed Hendrich, Kuiperbergstraat 34, 5628 DK EINDHOVEN.

- Ik heb het programma en dat werkt niet goed.
Kunt u mij vertellen wat ik moet veranderen?

Het is via de telefoon nogal lastig om even te bekijken wat er precies mis gaat met een bepaald programma. Als echter alle vragenstellers programma-listings zouden gaan opsturen zouden Rob en ik overstelpt worden met papier. U begrijpt dat we u hier niet mee kunnen helpen. U kunt het beste contact zoeken met gebruikers uit uw omgeving en aan hun uw probleem voorleggen. NB Programma's van de P2000gg (A-cassettes) en van Philips (uit Viditel) zijn uitvoerig getest op goede werking; als u daarmee problemen heeft ligt het waarschijnlijk aan uzelf of aan uw P2000. Neem eens contact op met een service-dienst van Philips of met uw dealer.

- Kan ik met de P2000 ook tekeningen maken, zoals bijvoorbeeld in Viditel?

De P2000 heeft (nog) geen zgn high-resolution. Dat betekent dat u met de P2000 maximaal 80 maal 72 blokjes op uw beeldscherm kunt tekenen. Dat u hiermee toch tot fraaie resultaten kunt komen bewijzen de plaatjes in Viditel. Om functies e.d. te tekenen is dit oplossend vermogen inderdaad te gering. Diverse mensen zijn momenteel iets aan het ontwikkelen waarmee u wel high-resolution op de P2000 krijgt. Afwachten luidt ook hier het devies.

Tot zover deze bloemlezing van vragen over de P2000 en de P2000-gebruikersgroep. U kunt de P2000-gebruikersgroep voor vragen telefonisch als volgt bereiken:

Woensdagavond van 18.30 u tot 22.30 u bij

Rob Geutskens, tel (04990) 74290.

Dinsdag- en vrijdagavond van 19.30 u tot 23.30 u bij

Peter Janssens, tel (04780) 10851.

Peter Janssens

DEEL 1.

OVER VARIABELEN.

Communicatie is in de omgang met andere mensen zeer belangrijk. Om die communicatie te voeren staan ons veel soorten communicatie en veel hulpmiddelen ter beschikking.

Een van de zaken die bij communicatie steeds weer worden gebruikt zijn BEGRIPPEN. Enkele begrippen zijn: krant, stoel en koelkast. De genoemde voorbeelden zijn verzamelnamen voor gelijksoortige dingen. Zo'n verzamelnaam geeft je een idee over welk soort ding het gaat. Over de preciese inhoud van de artikelen in de krant of de inhoud van de koelkast is nog niets duidelijk.

Met een beetje fantasie is BASIC programmeren ook communicatie te noemen. Een vaste (constante) tekst op het beeldscherm afdrukken gaat bv met PRINT "PHILIPS P2000T". We kunnen zo'n stuk tekst ook een naam geven. Lees maar verder.

VARIABELEN.

Een variabele is een verzamelnaam voor stukken tekst of getallen in een computer. De inhoud van een variabele kan wisselen (varieëren). We kennen in BASIC twee soorten variabelen.

1. TEKST of STRING VARIABELEN.

Deze dienen om lettertekens op te bergen en later natuurlijk weer te gebruiken. De inhoud van twee tekstvariabelen kan zeer verschillen net als de inhoud van twee willekeurige koelkasten. Tussen aanhalings tekens volgen enkele voorbeelden van tekst variabelen:

```
"De gelukkige wereld over 80 jaar."  
"NAAM :"  
"31 maart 1985"  
"19% BTW"
```

Door een tekst (=reeks lettertekens) een naam te geven is deze makkelijk te bewaren en weer terug te roepen. Dat opbergen of toekennen gaat als volgt:

```
DATUM$="31 maart 1985"  
HBTW$="19% BTW"
```

Het \$-teken geeft aan dat het gaat om een tekst.

Het \$-teken lezen als "string".

Het =-teken lezen "wordt".

Dus eerste voorbeeld lezen als: "datum-string wordt eenendertig maart negentienhonderdvijfentachtig."

Het afdrukken van de inhoud (=toegekende 'waarde') van een variabele gaat met PRINT:


```
PRINT DATUM#
PRINT HBTW#
```

Zinvolle namen gebruiken bij het opbergen is een eerste vereiste. De tekst "19% BTW" mag ook opgeborgen worden onder de naam XY#. Er is dan echter aan de naam niets over de inhoud af te leiden. Om de leesbaarheid en duidelijkheid te bevorderen gebruiken we logische namen.

2. GETAL VARIABELEN.

Bij getalvariabelen is het mogelijk om bv de uitkomst van een berekening onder een naam op te bergen. In tegenstelling tot tekstvariabelen komt er bij getals variabelen geen #-teken of "-teken aan te pas.

Voorbeelden van toekenningen zijn:

```
SOM= 5+18
HBTW= 19
LBTW= 5
```

Afdrukken op het beeldscherm gaat net als bij tekstvariabelen door PRINT:

```
PRINT SOM
```

of tekst en getal variabelen op een regel:

```
PRINT DATUM#;" ";SOM
```

ALGEMEEN.

Variabele namen:

- ** moeten met een letter beginnen.
- ** hebben een letter of cijfer als 2e teken.
- ** zijn minimaal 1 letter lang.
- ** meer dan twee tekens mag maar de computer gebruikt ze niet ter onderscheiding. Dus BTW19# en BTW5# zijn voor de computer gelijk.
- ** geven liefst een aanduiding van hun inhoud.
- ** mogen niet overeen komen (eerste twee tekens) met een basic opdracht PROOST is dus niet goed (vgl PRINT)
- ** mogen als hoofd of kleine letters ingetikt worden. Basic maakt er hoofdletters van.

Tot slot nog enkele tips. Gebruik niet te veel variabelen.

Reserveer bv X, Y, Z voor gebruik als loopindex (bij for - next). Stel ook enkele 'klad' variabelen vast.

Leg, bij grotere programma's, van af het begin een lijst aan met beschrijvingen van variabelen. Doe dit op een los vel papier.

```
bv : HBTW           percentage hoge BTW
      DATUM#       ingetikte datum
      SOM          uitkomst optelling
```

William van Male.

Bijna iedereen die zich een thuiscomputer aanschaft begint met programmeren in BASIC. De B in BASIC staat immers voor Beginners. Dat wekt de indruk dat BASIC geen taal is voor gevorderden. Wil je meetellen in computerland, dan moet je na een poosje toch minstens overstappen op PASCAL of Assembler.

Dat ook gevorderden veel meer met BASIC kunnen doen dan velen denken, bewijst Dr. Dirk J. Kroon in zijn boek "P2000 - BASIC-notities", dat binnenkort bij Uitgeverij Educaboek in Culemborg zal verschijnen. In zijn voorwoord schrijft de auteur: "Maar tussen basisschool en universiteit ligt zeker een gebied waar deze notities een BASIC-programmeur van dienst kunnen zijn." En zo is het ook. Een Operating System schrijven in BASIC kan niet, maar dat met een goede interpreter zoals de BASIC-NL een heleboel mogelijk is, spreekt uit elke bladzijde van dit boek.

BASIC-notities in vogelvlucht

Met de inhoud van "BASIC-notities" zouden we een lange rij Nieuwsbrieven kunnen vullen. Daarom hier in het kort een overzicht van de vele nuttige tips die u met dit boek in huis kunt halen.

In het hoofdstuk "Zuinig programmeren" geeft Kroon een aantal tips om geheugenruimte te besparen, bij voorbeeld door data (in de betekenis: dag, maand en jaar) in één integer op te slaan, door een geraffineerd gebruik te maken van ASCII-tekens die niet via het toetsenbord kunnen worden ingegeven en door een beter gebruik te maken van de verwaarloosde relaties 'waar' en 'niet waar'.

De mogelijkheden van DATA-regels worden beschreven in het gelijknamige hoofdstuk. Wat dacht u van een eenvoudige tekstverwerker, en het opslaan van bestanden in DATA-regels ?

In het hoofdstuk over de toetsenbordbuffer kunt u vernemen hoe u kunt constateren dat tijdens het lopen van een programma een toets ingedrukt is geweest, en welke toets dat was. Ook wordt beschreven hoe het programma een opdracht in de toetsenbordbuffer kan plaatsen.

"Rommelen met regelnummers" beschrijft een eenvoudige hernummeraar, de manier waarop u een programma 'onLISTbaar' kunt maken en het kopiëren van programma's.

Fouten zijn niet alleen om van te leren, maar ook om nuttig te gebruiken in uw programma. U kunt bij voorbeeld de STOP-toets gebruiken om een programma bewust in de fout te sturen, en die fout dan elegant op te vangen om uw doel te bereiken. Dit hoofdstuk beschrijft ook hoe u cassette- en printerfouten kunt ondervangen.

Hoe u de dingen op een rijtje kunt krijgen, beschrijft de auteur in hoofdstuk 7: "Sorteren". Hier leert u onder meer hoe u namen op achternaam kunt sorteren, zonder de 'voor'-letters achter de 'achter'-naam te zetten. Ook de snelle sorteermethode volgens Shell-Metzner komt aan de orde.

In "Van het toetsenbord" belijdt Kroon gemotiveerd zijn voorkeur voor LINEINPUT boven INPUT en beschrijft hij eigengemaakte invoerroutines, die gebruik maken van de handige INP("")-routine.

Hoofdstuk 9 handelt helemaal over de cassetterecorder die de P2000T siert. Het beschrijft een methode om de naam van een programma of een bestand op meer dan één letter te testen, en een routine die controleert of het klepje van de recorder open en dicht is gegaan.

Wat u allemaal met strings kunt doen vindt u in het hoofdstuk "Strings en wat daarmee kan". Voorbeelden: getallen opslaan alsof het strings zijn en, omgekeerd, het opbergen van strings in numerieke arrays, met als voordeel dat u ze op cassette kunt wegschrijven met CSAVE*. In dit hoofdstuk komt ook de miskende instructie INSTR ter sprake, waarmee de plaats van A\$ in B\$ kan worden opgespoord. Wist u trouwens dat uw bankrekeningnummer zo is samengesteld, dat de computer van de bank kan controleren of u een vergissing hebt gemaakt? De P2000T kan dat natuurlijk ook.

"Functies vormen een van de schoonste en best gestructureerde mogelijkheden van de P2000-BASIC", schrijft Kroon in de inleiding van het hoofdstuk "Functies", waarin onder meer de functie DEF FN aan de orde komt. Deze functies kunnen ook 'in elkaar' worden aangeroepen en ze zijn een ideaal middel om uw programma's kort en overzichtelijk te maken.

Hoofdstuk 12 bevat twee notities over arrays. De eerste beschrijft een omleidingsmanoeuvre die voorkomt dat een array, dat vooraan op de band staat, bij het starten van de P2000T wordt ingelezen alsof het een programma is, want dat leidt natuurlijk tot niets. De tweede notitie is handig als uw array groter groeit dan u had voorzien en u het in tweeën wilt knippen. Ook hier komt een miskende BASIC-opdracht ter sprake: VARPTR.

"Mengelwerk" is de titel van het dertiende hoofdstuk. Het beschrijft de voorkeur van de auteur om steeds dezelfde letters aan bepaalde variabelen te geven, zodat u niet steeds uw programma hoeft af te grazen om te zien met welk type variabele u te doen hebt. In dit hoofdstuk ook een programma om de ingebouwde klok van de P2000 te gebruiken, de mogelijkheid niet alleen op het scherm te schrijven maar er ook van te lezen en het advies om "niet als dank ..." voor 't aangenaam verpozen de computer te laten zitten met de schillen en de dozen.

Veel van het geleerde wordt in praktijk gebracht in het laatste hoofdstuk: "Modelprogramma's", waarin onder meer een ledenbestand en een kasboek aan de orde komen.

Voorbeeld: een klok

Als voorbeeld van de lekkernijen die Kroon in zijn "P2000 BASIC-notities" opdient, zullen we hier met toestemming van de auteur de notitie over de ingebouwde klok van de P2000 overnemen.

De P2000 heeft een ingebouwde klok, die in ongeveer 20 minuten rondloopt. Hoe kunnen we hiermee een timer maken?

De klok vinden we op de geheugenplaatsen &H6010 en &H6011 en de "waarde" van de klok zouden we kunnen vinden met:

```
W = PEEK(&H6010)+256*PEEK(&H6011)
```

Toch is het niet zeker dat dit altijd goed gaat. De geheugenplaats &H6010 wordt namelijk 50 maal per seconde met 1 opgehoogd. Is de waarde 255 bereikt, dan wordt &H6010 op 0 gezet en &H6011 met 1 opgehoogd.

Als we nu de klok zouden berekenen juist op het moment dat dit zou gebeuren, dan meten we misschien wel fout. We lezen de klok dus als volgt uit:

```
20 HB=PEEK(&H6011):LB=PEEK(&H6010):IF HB<>PEEK(&H6011)THEN 20
   ELSE W=LB+256*HB
```

Vindt tijdens het uitvoeren van regel 20 een verandering van het hoge byte plaats, dan wordt opnieuw berekend.

Kijk nu even naar het onderstaande programma.

Als de berekende "waarde" van de klok, W, kleiner is dan de vorige waarde, WV, dan betekent dit dat de klok een keer "rond" is geweest. In dat geval verhogen we een teller T met 1. Op regel 40 berekenen we de juiste tijd, we slaan de waarde W op in WV en op regel 50 wordt de tijd in uren, minuten en seconden op het scherm getoond. Kijk even naar de PRINT USING die hier gebruikt is. PRINT USING maakt immers gebruik van een string waarin bepaalde karakters door cijfers worden vervangen. Deze string is hier:

```
##:##:##
```

zodat de tijd verschijnt als:

```
12:17:44
```

Dan nu het volledige programma:

```
5 REM timer
10 T=0:WV=0:POKE&H6010,0:POKE&H6011,0:REM timer op nul
20 HB=PEEK(&H6011):LB=PEEK(&H6010):IF PEEK(&H6011)<>HB THEN 20
   ELSE W=(LB+256*HB)/50
30 IF W<WV THEN T=T+1:REM klok rond geweest
40 WV=W:W=W+T*1331:REM aantal rondgangen erbij tellen
50 PRINT CHR$(4)CHR$(10)CHR$(10)CHR$(141)CHR$(2)USING"##:##:##";
   INT(W/3600);(INT(W/60-60*INT(W/3600)));W-60*INT(W/60):GOTO 20
```

Toen mijn vrouw en ik deze winter een administratie programma gingen schrijven, waarin we onze eigen specifieke wensen konden verwerken stuitten we op het volgende probleem. Namelijk "het wegzetten van de stringsave routine's in het BASIC programma", op een wijze, waarop het een minimum aan ruimte inneemt en bovendien niet eerst weggezet behoeft te worden met READ, DATA en POKE statements, want dit kost veel tijd en wachten moeten we al genoeg. De oplossing bleek na enige malen m'n oor te luisteren te hebben gelegd vrij eenvoudig.

Tenminste als je weet wat je moet doen.

Als we het programma "STRINGSAVE" zoals het aangeboden wordt bekijken dan zien we in de beschrijving dat het eigenlijk overal in het RAM geheugen kan staan, zolang we maar aangeven waar het begint. Wat is er dan mooier om deze routine gebruiksklaar achter het BASIC programma te zetten. Zo neemt het namelijk z'n minimale ruimte in en is direct aan te roepen, mits we de juiste verbinding leggen met BASIC.

Laten we eerst maar proberen de machinetaal achter BASIC te zetten om daarna de verbinding te maken met BASIC.

Om te beginnen moet het programma STRINGSAVE geladen worden, want we laten de computer het werk doen. Na het inladen DELETE'n we alles wat we niet nodig hebben, met andere woorden, de regels 930-940-950-990-999-1000-1001-1002 en 1003 blijven over, want deze hebben we nodig.

Op de adressen &H6405 en &H6406 staat waar het BASIC programma eindigt, dit is een adres van een geheugenplaats. Van wat er op deze twee plaatsen staat maken we gebruik, om zowel de routine's op de juiste plaats te zetten, als de verbinding met BASIC te leggen. Maar eerst moet er plaats gemaakt worden achter BASIC.

Dit doen we als volgt:

```
Type in.. POKE&H6406,PEEK(&H6406)+1 ..en sluit af met de return toets.
```

Hierna schrijven we er 1 programma regel bij.

```
En wel.. 30 P=PEEK(&H6405)+256*PEEK(&H6406)-256 ..en sluit af met de return toets.
```

Als er nu een RUN gegeven wordt, zet de computer de machinetaal routine uit de regels 1000-1001-1002 en 1003 achter het basic gedeelte, waarbij in regel 20 eerst bepaald wordt waar de eerste DATA byte komt. Na de RUN kunnen de regels 930 tot en met 1003 ge DELETE worden, ze zijn namelijk overbodig geworden.

Om de koppeling met BASIC te complementeren typen we de volgende regels bij.

```
20 KL=PEEK(&H60DD):KH=PEEK(&H60DE)
40 P%=VAL("&H"+HEX$(P))
50 PL=PEEK(VARPTR(P%)):PH=PEEK(VARPTR(P%)+1)
60 POKE&H60DD,PL:POKE&H60DE,PH
```

Voor wat nu in de P2000 staat verdient het aanbeveling om op cassette te bewaren, want het is de STRINGSAVE basis voor elk nieuw te maken BASIC programma, wat we hierin typen. Het voordeel van deze constructie is dat er vrij in het te programmeren BASIC programma ge EDIT kan worden. De routine schuift vanzelf mee en bij iedere RUN wordt opnieuw de juiste koppeling gemaakt. Ook het wegschrijven naar cassette en het copieren na een RUN kan zonder risico van verlies van de routine gedaan worden.

Een nadeel is dat het niet zo makkelijk achter een bestaand programma gekoppeld kan worden. Hierbij kan men beter eerst het afgeslankte en eventueel hernummerde originele STRINGSAVE programma achter het BASIC programma koppelen en daarna de eerder beschreven handelingen uitvoeren.

Voor de mensen die het nu direct willen proberen kunnen dat doen door het volgende bij wijze van proef in te typen.

```

10 CLEAR 100:DIMA$(1)
100 A$(1)="Probeersel"
110 CSAVE*A#$"Proef 1"
120 A$(1)="":PRINT"Probeersel is nu geSAVEd en A$(1) is
    leeg"
130 CLOAD*A#$"Proef 1"
140 PRINT A$(1)
1000 CLEAR 50:POKE&H60DD,KL:POKE&H60DE,KH
1010 END

```

Regel 1000 is een nette afsluiting van het programma en dient uitgevoerd te worden voordat met het programma afsluit.

Tot slot wil ik de Louis Naus bedanken voor zijn telefonische uitleg want aan hem dank ik deze kennis.

SUCCES

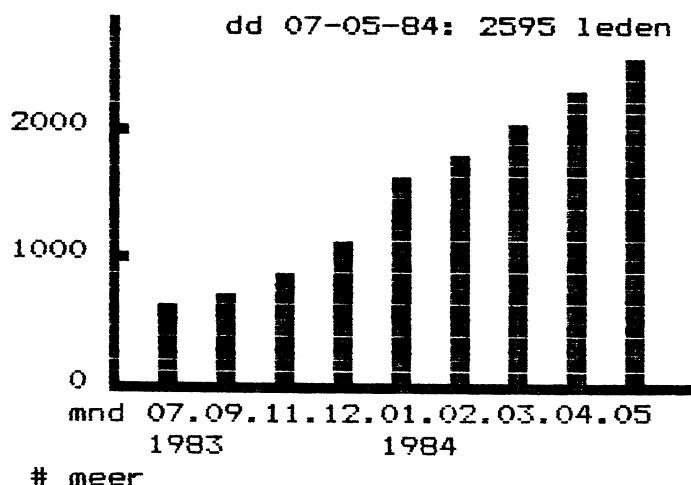
Arie Stolze

Philips 5158217a Oc

HCC-P2000 

(c) P2000gg
Peter
Janssens

Statistiek van het ledental



In allerlei media wordt regelmatig gepubliceerd over "operating systemen" en aanverwante zaken. Kreten als MS-DOS, CP/M, UCSD, UNIX en noem ze maar op zijn dan niet van de lucht. Deze "operating systemen", zeg maar "besturingssystemen", hebben in het algemeen een aantal eigenschappen gemeen:

- Ze zijn allemaal op het gebruik van flexibele schijven (floppy's) gericht.
- Bestanden en programma's van het ene besturingssysteem zijn niet zonder meer uitwisselbaar met andere systemen.
- Ze zijn altijd zeer nadrukkelijk aanwezig op de microcomputer, waarbij termen als "opstarten" of "booten" (dit heeft niets met water te maken) hun gewag doen. (voor de liefhebber nog enkele andere termen: "dir", "pip", "stat", "format", "copy", "sys".....)

Moeilijk? Nee, hoor; dit artikel tracht het eenvoudig te houden.

Uw eigen P2000 (T) heeft ook zo'n besturingssysteem. Als gebruiker merkt u daar veel minder van dan bij andere computers omdat dit besturingssysteem sterk ge-integreerd is met de rest van de P2000 en mede daardoor veel gemakkelijker te bedienen is. Stop maar eens een Basic-bandje in de recorder en druk op RESET. Een aardig voorbeeld van een functie van het P2000-besturingssysteem.

Blijkbaar is een onderdeel van het besturingssysteem de aansturing van de cassetterecorder. Alle functies van dit besturingssysteem zijn toegankelijk voor de programmeur, en alle beginnen ze op startadres &H18.

Dit artikel tracht aan de hand van twee eenvoudige voorbeelden u te laten zien hoe u deze functies geheel zelf vanuit Basic kunt gebruiken, dus met gebruik van zo weinig mogelijk machinetaal.

Een klein stukje machinetaal is echter onvermijdelijk:

7E	LD A, (HL) ;	haal het funktienummer op
CD 18 00	CALL 18H ;	aanroep cassette-besturing
77	LD (HL), A ;	geef foutcode terug
C9	RET ;	terug naar basic

Hoe u deze machinetaal in het geheugen kunt zetten vindt u in voorbeeld 1, regel 20 t/m 40.

Mogelijke opdrachten (functies) die u aan de cassetteroutine kunt geven:

opdracht	betekenis
0	initialiseer cassetterecorder (gebeurt steeds bij het aanzetten van de P2000)
1	spoel terug naar het begin van de band
2	spoel N blokken vooruit (de waarde van N moet u eerst op geheugenadres &H604F POKE-n, zie "tape-header")
3	spoel N blokken terug (als bij opdracht 2, zie voorbeeld 2, regel 160 en 170)
4	schrijf "einde band" (d.w.z. de P2000 weet dan later dat vanaf dit punt het bandje leeg is)
5	schrijf een stuk geheugen naar de cassette (startadres staat op &H6030 en &H6031, aantal bytes staat op &H6032 en &H6033, alsmede op &H6034 en &H6035, zie "tape-header")
6	lees een stuk geheugen in van de band (als bij opdracht 5)
7	status van de cassetterecorder (wel/geen cassette, wel/geen stopje; werkt niet met de machinetaal uit het voorbeeld)

Beschrijving van de POKE en PEEK-adressen van de "tape-header":

adres (&H)	betekenis
6030 - 6031	startadres waar de informatie naartoe moet c.q. vandaan komt. (bij opdracht 5 en 6)
6032 - 6033	lengte van de file (als boven)
6034 - 6035	aantal geldige bytes van dit blok (als boven)
6036 - 603D	eerste 8 tekens van de naam
6041	type van de file (B = basic, P = mach.-taal, enz.)
604F	bloknummer in de file, of aantal blokken dat vooruit of achteruit gespoeld moet worden

Voor meer informatie betreffende de "tape-header"-adressen verwijs ik gaarne naar het "overzicht adressen" van Joep Willemsen in Nieuwsbrief 8 (adressen &H6030 t/m &H604F).

De informatie uit de "tape-header"-adressen wordt door het besturingssysteem met ieder blok op de cassette gezet bij het schrijven op de cassette, en wordt bij elke lees-actie tesamen met de inhoud van het blok weer ingelezen. Zo kunt u met PEEK(&H6036) de ASCII-waarde van de eerste letter van de file-naam opvragen, en met PEEK(&H6041) kunt u zien van welk type applicatieprogramma de file is (vergelijk regel 130 van voorbeeld 1).

De volgende 2 voorbeelden verduidelijken wellicht het een en ander. U leert er het meeste van door het eens zelf te proberen. Gebruik echter voor uw experimenten alleen LEGE bandjes, of bandjes die gewist mogen worden.


```

10 REM Voorbeeld 1: Inlezen van een plaatje van "Peter's Plaatjes Program"
20 CLEAR50,&H9AFF: REM reserveer geheugenruimte voor machinetaal
30 FORI=&H9B00TO&H9B05:READA$:POKEI,VAL("&H"+A$):NEXT:REM POKE machinetaal
40 DATA 7e,cd,18,00,77,c9:REM machinetaal (zie tekst)
50 DEFUSR=&H9B00: REM startadres van machinetaalroutine
60 PRINTCHR$(12)"Terugspoelen"
70 FCZ=1:ERZ=USR(FCZ): REM spoel naar begin van de band
80 IFERZ(>)THEN300: REM cassettefout
90 PRINT"En nu gaan we lezen"
100 GOSUB400: REM vul de cassetteheader met benodigde getallen
110 FCZ=6:ERZ=USR(FCZ): REM lees gegevens
120 IFERZ(>)THEN300: REM cassettefout
130 IFPEEK(&H6041)<>64THEN100: REM Geen plaatje van PPP; dus verder zoeken
140 REM Nu hebben we een plaatje ingelezen. Even op het scherm zetten:
150 PRINTCHR$(2):K=&H5000
160 FORI=0TO21:FORJ=0TO39
170 POKEK,PEEK(&H9C00+I*40+J):K=K+1
180 NEXTJ:K=K+40:NEXTI
190 PRINTCHR$(4)CHR$(22)CHR$(1);
200 END
300 REM "Afhandeling" cassettefout
310 PRINT"Cassettefout "CHR$(ERZ)
320 END
400 REM Het vullen van de cassetteheader
410 POKE&H6030,&H00:POKE&H6031,&H9C:REM het plaatje moet naar &H9C00
420 POKE&H6032,&H00:POKE&H6033,&H04:REM Lengte van PPP-plaatjes is &H400 bytes
(= 1 Kb)
430 POKE&H6034,&H00:POKE&H6035,&H04:REM Aantal geldige bytes is &h400 (= 1 Kb)
440 RETURN

```

```

10 REM Voorbeeld 2: Dit voorbeeld bewaart een plaatje op het scherm als file
20 REM van 2Kb op tape en leest het daarna weer in.
30 REM Het bandje wordt eerst gewist, dus DOE DIT ALLEEN MET EEN LEEG BANDJE!!
40 CLEAR50,&H9EFF:RESTORE:FORI=&H9F00TO&H9F05:READA$:POKEI,VAL("&h"+A$):NEXT:DEF
USR=&H9F00:DATA 7e,cd,18,00,77,c9
50 PRINT"Deze tekst wordt op het bandje gezet!"
60 FCZ=1:ERZ=USR(FCZ): REM spoel terug naar begin
70 IFERZ(>)THEN500: REM cassettefout
80 FCZ=4:ERZ=USR(FCZ): REM wis bandje
90 IFERZ(>)THEN500: REM cassettefout
100 POKE&H6030,&H00:POKE&H6031,&H50: REM startadres v.h. scherm is &h5000
110 POKE&H6032,&H00:POKE&H6033,&H08: REM aantal bytes is &h800 (= 2Kb)
120 POKE&H6034,&H00:POKE&H6035,&H08: REM als boven
130 FCZ=5:ERZ=USR(FCZ): REM bewaar op tape
140 IFERZ(>)THEN500: REM cassettefout
150 PRINTCHR$(12)"twee blokken terugspoelen"
160 POKE&H604F,2: REM 2 blokken
170 FCZ=3:ERZ=USR(FCZ): REM spoel n blokken terug
180 IFERZ(>)THEN500: REM cassettefout
190 POKE&H6030,&H00:POKE&H6031,&H50: REM startadres is weer &h5000
200 POKE&H6032,&H00:POKE&H6033,&H08: REM aantal bytes idem
210 POKE&H6034,&H00:POKE&H6035,&H08: REM als boven
220 REM in feite zijn regels 190 t/m 210 overbodig.
230 REM De cassetteroutine verandert deze waarden ALLEEN bij functie 6
240 FCZ=6:ERZ=USR(FCZ): REM lees van tape
250 IFERZ(>)THEN500: REM cassettefout
260 PRINTCHR$(4)CHR$(22)CHR$(1)"Klaar"
270 END
500 REM Afhandeling cassettefouten
510 PRINT"Cassettefout "CHR$(ERZ)
530 END

```

Uit de vragen die ik op woensdagavond krijg, als ik telefoonwacht heb, blijkt dat veel mensen de P2000 willen gebruiken voor bedrijfsmatige toepassingen, bij voorbeeld de financiële administratie, voorraadbeheer, voortgangscontrole en dergelijke. Blijkbaar zijn er talloze zelfstandigen en kleine en middelgrote ondernemers die de tekenen des tijds verstaan en op bescheiden schaal met automatisering willen beginnen.

Hierbij rijzen twee vragen:

- 1 Is de P2000T bruikbaar voor professionele toepassingen ?
- 2 Is er programmatuur die de P2000T geschikt maakt voor dat doel ?

Beide vragen kunnen met 'ja' worden beantwoord. De eerste vraag omdat de P2000 in vergelijking met veel andere thuiscomputers uiterst degelijk blijkt te zijn. Hij kan zonder bezwaar de hele werkdag ingeschakeld blijven, de ingebouwde minicassetterecorder is een bijzonder praktisch opslagmiddel en het toetsenbord is van professionele allure.

Als dit niet zo was, zou geen sterveling erover piekeren programma's voor bedrijfstoepassingen van de P2000T te ontwikkelen. Uit het feit dat er verscheidene professionele programma's zijn ontwikkeld mag u dus afleiden dat de P2000T geschikt is voor dat doel. En daarmee zijn beide vragen bevestigend beantwoord.

In dit artikel zullen we een korte beschrijving geven van de professionele programma's die door AKG Microsystemen in Amsterdam en Job van Broekhuijze in Ridderkerk zijn ontwikkeld. Er zijn meer ontwikkelaars van professionele programmatuur, zoals Philips PTIS en Miniware, maar die programma's zijn mij op dit moment niet bekend en ik weet ook niet of ze op het T-model draaien. Als dat zo is, kunnen ze later nog wel eens aan de orde komen. Voorlopig is het doel aan te tonen dat met de P2000T wel degelijk bedrijfsmatig kan worden gewerkt.

Voordat we die programma's onder de loep nemen, eerst nog enkele kanttekeningen. Met 'professioneel' bedoel ik dat de programma's zijn bedoeld voor kleine en middelgrote ondernemers, zelfstandigen, verenigingen en dergelijke die een vakkundige administratie willen voeren. Het woord houdt geen oordeel in over de kwaliteit van bij voorbeeld de programma's die in Viditel zijn opgenomen en die door de P2000gg op cassette worden aangeboden. Die kunnen wel degelijk professionele kwaliteiten hebben, maar ze zijn niet in eerste instantie bedoeld voor bedrijfsmatig gebruik.

In de tweede plaats heeft het ontwikkelen van deze professionele programma's de makers veel tijd en geld gekost. Je zou kunnen zeggen dat ze bedrijfsmatig zijn ontwikkeld. Ze worden dan ook niet gratis of voor een geringe vergoeding ter beschikking gesteld, maar ze kosten globaal enkele honderden guldens. Bovendien kunt u ze alleen kopen als u bereid bent een contract te tekenen, dat bepalingen bevat over het gebruik van de programmatuur, of op andere wijze verklaart geen misbruik van de programmaturu te zullen maken. Die bepalingen komen er kort en goed op neer dat u geen kopieën van de programma's mag maken en doorverkopen. Voor wie niet bekend is met de gang van zaken op dit gebied: dat is bij professionele programma's de gewoonste zaak van de wereld. Zonder die bepaling zou de maker veel geld moeten toe-

leggen op zijn produkt; hij zou slechts enkele kopieën kunnen verkopen. Daarmee is niemand gediend, want het zou betekenen dat het ontwikkelen van een professioneel programma alleen maar geld kost, en niets oplevert. Wie zou er in zo'n situatie nog over peinzen professionele programma's te ontwikkelen ?

Het contract bevat ook bepalingen die in uw voordeel zijn. Blijken er foutjes in het programma te zitten, dan worden die kosteloos hersteld. En hebt u een ongelukje, waardoor u uw programma niet meer kunt gebruiken, dan krijgt u gratis of tegen geringe kosten een nieuw programma.

Financieel programma WIEWAT

Het financiële programma WIEWAT is ontwikkeld door AKG Microsystemen. Het programma is menu-georiënteerd. Op het scherm vindt u dus steeds aanwijzingen over hoe u verder moet handelen. Dat neemt niet weg dat een uitvoerige handleiding, die niets aan het toeval over laat, wordt meegeleverd. Ook als u, zoals ik, nooit helemaal het verschil tussen debet en credit hebt kunnen doorgronden, kunt u met dit programma een boekhouding voeren die de toets der kritiek kan doorstaan.

De makers hebben hun best gedaan zoveel mogelijk boekhoudgegevens in zo weinig mogelijk bytes te persen. Voor een boeking hebben ze dan ook niet meer dan 8 bytes nodig. In feite zijn dat twee of drie boekingen omdat altijd debet en credit wordt geboekt (2 boekingen) en omdat er veelal sprake is van BTW (derde boeking). Dit betekent dat u alle boekhoudgegevens van het hele jaar, of bij omvangrijke administraties van een kwartaal of een maand, steeds allemaal in het geheugen van uw P2000 laadt, de nodige mutaties uitvoert en de hele zaak weer op cassette wegschrijft. Dit heeft twee voordelen. In de eerste plaats hoeven niet telkens gegevens van cassette te worden geladen en weggeschreven, zodat het programma zeer snel is. U hebt alle gegevens onmiddellijk ter beschikking. In de tweede plaats hoeft de cassette maar twee keer te worden gebruikt: bij het inlezen van de boekhouding en bij het wegschrijven daarvan met de door u ingevoerde mutaties. Dat verhoogt de bedrijfszekerheid. Verdeelt u het jaar in perioden, dan kunt u de saldi meenemen naar de volgende periode.

Als u veel boekingen hebt kunt u wel voor alle zekerheid af en toe het gemuteerde bestand op cassette schrijven, en dan verder gaan. Het is ook mogelijk, en het verdient zelfs aanbeveling, daarvoor twee of drie cassettes te gebruiken en die te laten rouleren. De kans dat er iets mis gaat is niet groot, maar als dat gebeurt bent u in het ergste geval de laatste mutaties kwijt.

In de kleinste uitvoering van WIEWAT, waarvoor u een P2000T met 32K vrij geheugen nodig hebt, kunt u zelf maximaal 255 verschillende grootboekrekeningen specificeren. Die kunt u onderverdelen in ten hoogste 31 subadministraties, bij voorbeeld winkel en werkplaats, kostenplaatsen, filialen, artikelgroepen, budgetten en dergelijke. Verzonden en ontvangen facturen kunt u in etappes boeken. U boekt ze op de datum dat ze worden verzonden of ontvangen en later, als de betaling binnen is of als u zelf betaald hebt, maakt u de boeking compleet. De kleinste uitvoering van WIEWAT biedt bij 32K geheugen plaats aan 600 à 700 openstaande posten (circa 1300 bij 40K). Fouten kunt u verbeteren zonder tegenboeking. Dat houdt de zaak overzichtelijk.

De grootboekrekeningen zijn bij voorbeeld in drie groepen te verdelen:

- 1 Kostenrekeningen en dergelijke, waaruit de verlies- en winstrekening wordt samengesteld (1...40).
- 2 Rekeningen waaruit de balans wordt opgesteld, zoals inventaris, machines en dergelijke (41...110).
- 3 Debiteuren en crediteuren (111.255).

Bij het boeken wordt telkens gevraagd naar de nummers van de grootboekrekeningen. Omdat u die niet allemaal uit uw hoofd zult kennen, kunt u met behulp van de codetoets spieken en dan weer teruggaan naar het menu waar u was gebleven.

De kracht van een financieel programma ligt niet zozeer in de manier waarop u de boekingen invoert, als wel in de mogelijkheden om uit die boekingen allerhande overzichten samen te stellen. Wat dat betreft is WIEWAT een zeer krachtig programma. U kunt op elk moment overzichten maken van de geldrekeningen (giro, bank, kas), waarbij het saldo wordt berekend. Hetzelfde geldt voor het journaal (een chronologisch overzicht van alle boekingen) en voor elke grootboekrekening afzonderlijk. Bovendien stelt de computer de eindbalans (of de tussenbalans) en de verlies- en winstrekening voor u samen.

Ook verstrekt hij de gegevens die u nodig hebt om de aangifte voor de omzetbelasting te doen. Bij al deze overzichten hebt u de keus uit zichtbaar maken op het scherm en afdrukken door middel van de printer.

WIEWAT is op de groei gemaakt. Als de zaken erg goed gaat kunt u er mee doorgroeien tot de P3500.

Het is jammer dat dit programma zowel op het scherm als op de printer eigenlijk uitsluitend met hoofdletters werkt. Al is het dan een financieel programma, daarom hoeven het nog niet allemaal kapitalen te zijn. Onderzoekingen hebben aangetoond dat kleine letters ('onderkast' in drukkerstaal) minder vermoeiend zijn om te lezen. De ANWB heeft bij voorbeeld niet voor niets voor de wegwijzers 'kleine' letters gekozen.

Bij nader inzien blijken kleine letters met WIEWAT wel degelijk mogelijk te zijn. Maar het sorteren werkt dan niet goed meer omdat in de ASCII-tabel de hoofdletter Z voor de kleine letter a komt.

Boekhoudprogramma JVB

Het eerste dat opvalt bij het boekhoudprogramma van Job van Broekhuijze is dat het totaal anders van opzet is dan WIEWAT. Dit houdt geen oordeel in over de kwaliteit. Blijkbaar zijn er vele wegen die naar het boekhoudparadijs leiden.

Dit boekhoudprogramma werkt onder een aangepaste versie van BASIC UK, die BASIC BR heet. Na het opstarten verschijnt niet het vertrouwde beeld op het scherm, maar de naam van de auteur. Hierna kan het boekhoudprogramma worden geladen met START. Eén van de veranderingen in de interpreter betreft de toetscodetabel, die in RAM wordt gezet. Daardoor kunnen op eenvoudige wijze functies aan bepaalde toetsen worden toegekend.

Voordat met boekhouden kan worden begonnen moeten eerst de grootboekrekeningen worden gedefinieerd. Het maximaantal rekeningen bedraagt 64. Daarin zijn de debiteuren- en crediteurenrekeningen niet begrepen; dat kunnen er nog eens maximaal 998 zijn. De rekeningnummers bestaan uit vier cijfers en zijn verdeeld in vier groepen:

vanaf 1000 voor debiteuren
vanaf 2000 voor crediteuren
vanaf 3000 voor exploitatierekeningen (in- en verkoop, kosten, verlies en winst e.d.)
vanaf 5000 voor 'onpersoonlijke' rekeningen (bank, giro, kas, balans, voorraad, inventaris e.d.)

Daarnaast kunnen nog rekeningen worden geopend die niet in één van deze vier groepen thuis horen, zoals premieheffing en persoonlijke verplichtingen. Enkele rekeningnummers, zoals BTW, liggen tevoren vast.

Na het invoeren van de grootboekrekeningen typt u het woord STOP in (dus niet met de stoptoets, zoals de handleiding suggereert) en worden de grootboekrekeningen numeriek gesorteerd. Er kan nu een nieuwe data-cassette worden geopend, waarop de boekingen worden weggeschreven. In het eerste blok dat op cassette wordt geschreven staan de rekeningnummers met de bijbehorende omschrijving en de saldi.

Nu kunnen we gaan boeken. Na enige handelingen, waarbij het scherm de weg wijst, verschijnt op het scherm een menu dat kan worden ingevuld. Dit vermeldt achtereenvolgens:

- boekingsnummer
- datum
- notanummer
- omschrijving
- rekeningnummer
- tegenrekening
- bedrag D/C
- BTW% code

Het boekingsnummer wordt door de computer ingevuld; het is steeds 1 hoger dan het laatste boekingsnummer. De datum moet u zelf invullen, maar bij elke volgende boeking kunt u de datum onveranderd overnemen door de terugtoets in te drukken.

Na de omschrijving moet u twee rekeningnummers invoeren: een debet- en een credit-rekening. Op beide wordt hetzelfde bedrag geboekt, tenzij BTW moet worden berekend. Dan wordt het door u ingevoerde bedrag gesplitst in drie bedragen: inclusief BTW, exclusief BTW en de BTW zelf. Van alle drie de grootboekrekeningen verschijnen de saldi onder in beeld, zowel voor als na de mutatie. Tijdens het invoeren houdt de computer dus van alle grootboekrekeningen de saldi bij.

Elke boeking wordt onmiddellijk afgedrukt door de printer. Als die niet is aangesloten of ingeschakeld, werkt het programma niet. Dat kan zijn voordelen hebben als er iets mis gaat omdat u dan alle mutaties op papier hebt.

Het boekhoudprogramma van Job van Broekhuijze schrijft alle boekingen in blokken van 35 boekingen weg op de datacassette. Alleen de saldi van de grootboekrekeningen blijven in het geheugen aanwezig, zodat ze bij elke boeking verhoogd of verlaagd kunnen worden. Voordeel van deze aanpak is dat het geheugen niet hoeft te worden uitgebreid (16K is genoeg). Om het wachten op de cassetterecorder te bekorten is het programma zo gemaakt dat niet onnodig wordt teruggespoeld. Per cassettekant kunnen 4200 boekingen worden vastgelegd. Het totale aantal boekingen is dus onbeperkt, als u tenminste kans ziet lege cassettes op de kop te tikken. Natuurlijk verdient het aanbeveling 'back-up'-

cassettes te maken voor het geval een datacassette in het ongereede raakt. En die kans is niet helemaal uitgesloten, gezien het feit dat de cassette telkens wordt gebruikt om de mutaties vast te leggen.

Alle boekingen worden ingevoerd in het journaal. Aan de hand daarvan maakt de computer op elk gewenst moment een overzicht van het grootboek, de balans, de verlies- en winstrekening en overzichten van debiteuren- en crediteurenrekeningen.

Bij elke boeking kunt u een codegetal ingeven, waardoor het mogelijk is de posten uit te splitsen naar filialen, kostenplaatsen en dergelijke. Uiteraard wordt ook de BTW keurig uitgesplitst. De BTW wordt niet ingevoerd door middel van een code voor hoog, laag of nultarief, maar door het invoeren van het werkelijke percentage (0, 5 of 19%). Verandert het tarief, dan hoeft uw programma dus niet te worden gewijzigd.

Het programma werkt wel vriendelijk, al zult u een uurtje of wat nodig hebben om voetangels en klemmen te leren ontwijken. Drukt u bij voorbeeld op de regeltoets waar dat niet hoeft (na het invoeren van het grootboeknummer, als de computer zelf de bijbehorende omschrijving ter controle opzoekt), dan wordt de volgende regel van het menu overgeslagen of gevuld met nullen. Aan het einde van het menu is er echter een ontsnappingsmogelijkheid naar het begin van het scherm, zodat u fouten kunt herstellen voordat de boeking effectief wordt. Ook later kunt u boekingen terughalen en verbeteren.

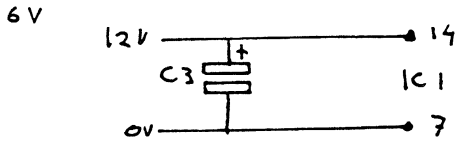
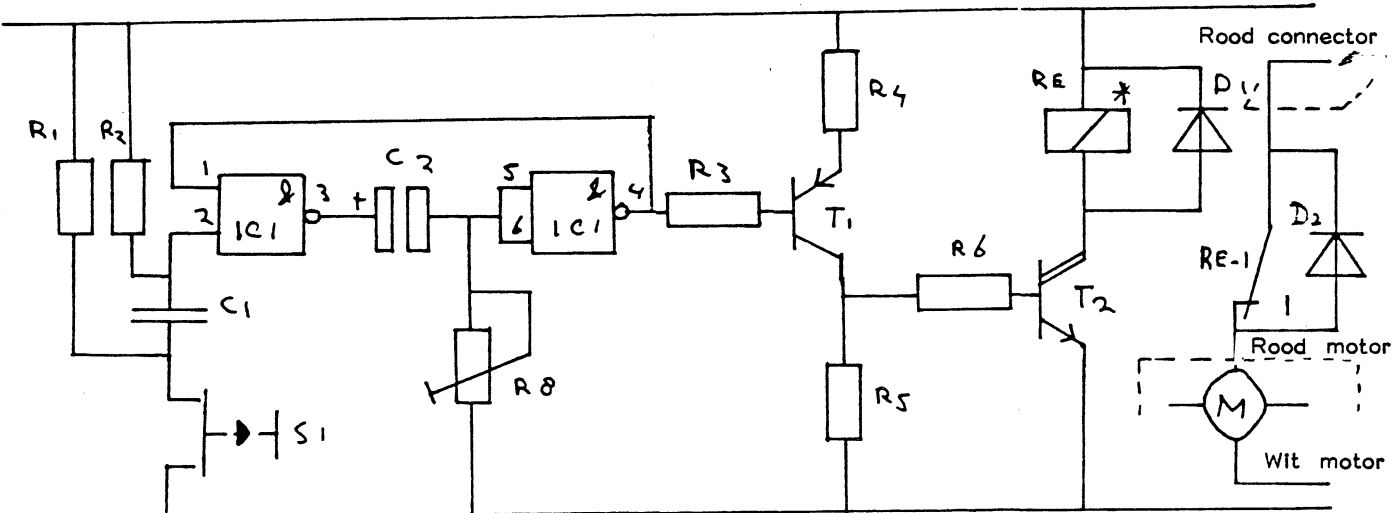
Na enig oefenen werkt het programma redelijk gebruiksvriendelijk. Het gebruikt zowel hoofd- als kleine letters en oogt daardoor ook wat vriendelijker dan een programma met louter kapitalen. En mocht u problemen hebben, dan kunt u onmiddellijk de telefoon grijpen. Ik heb geen scherm kunnen ontdekken waar de naam van de maker met zijn telefoonnummer niet pontificaal op de eerste regel staat.

Andere professionele programma's

Zowel AKG en Job van Broekhuijze als andere leveranciers hebben meer programma's voor professioneel gebruik in huis, zoals voorraadprogramma's (dat van AKG heet WATWAAR), die in grote lijnen op dezelfde manier werken als de financiële programma's die we hier hebben beschreven. Ze komen duidelijk uit hetzelfde nest. We zullen ze hier niet verder beschrijven omdat de bedoeling van dit verhaal is te laten zien dat de P2000T wel degelijk geschikt is voor bedrijfsmatige toepassingen. Informatie over die andere professionele programma's en over de prijzen kunt u vinden in de advertenties elders in deze Nieuwsbrief.

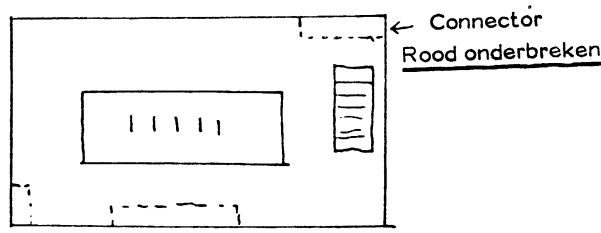
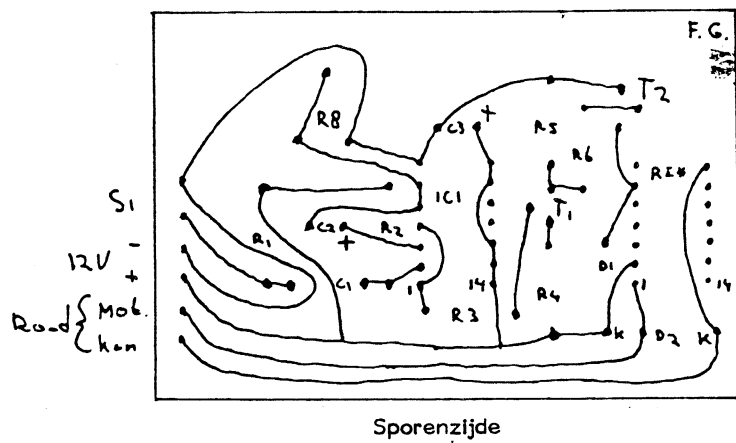
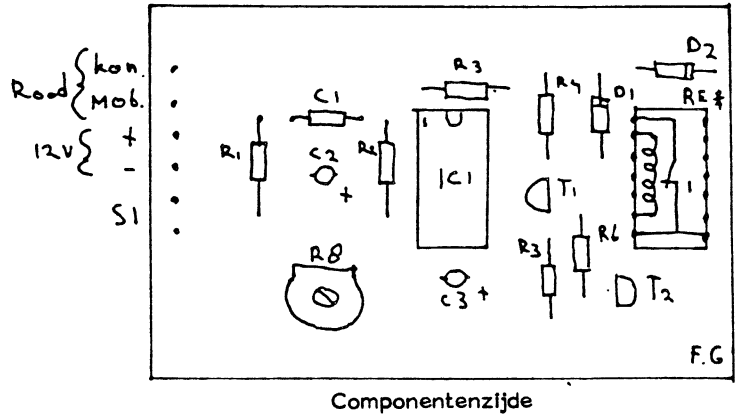
Rob Geutskens

12V+



Stuklijst

- R1, R2 = 10 kΩ
- R3 = 220 kΩ
- R4, R5 = 12 kΩ
- R6 = 100 kΩ
- R8 = 1 MΩ
- (instelpotmeter)
- C1 = 100 nF
- C2, C3 = 10 μF/16 V
- (tantaal)
- T1 = BC 557
- T2 = BC 517
- IC1 = CD 4011
- D1 = 1N 4148
- D2 = 1N 4001
- S1 = Drukknop
- RE* = DIL-relais 10...12 V met wisselcontact



Bovenzijde cassetterecorde

Bijgaande terugspoelautomaat is ontwikkeld door F. Geeraert, Schuitvlotstraat 46 in Groede. De maker heeft de terugspoelautomaat al enige tijd in gebruik en hij blijkt uitstekend te voldoen. "Het is een ware verademing dat de cassette niet iedere keer moet terugspoelen", schrijft hij.

Misschien ten overvloede moeten we hier bij aantekenen dat de schakeling door de redactie van de Nieuwbrief niet is nagebouwd en getest en dat wij dus ook geen inlichtingen kunnen geven over de verkrijgbaarheid van de onderdelen en de werking van de schakeling.

Beschrijving van de schakeling

"Deze schakeling bespaart een hoop tijd in het gebruik met de digitale cassetterecorder, alhoewel wij P2000-gebruikers niet mogen klagen over de wachttijden bij laden en saven van programma's. Het onzinnige terugspoelen bij saven en laden en ook bij de inhoudsopgave is nu verleden tijd. Wanneer men nu een programma wil laden, dat aan het eind van de cassette staat en de cassetteband staat in het midden, dan hoeft men maar CLOAD"n" in te tikken, return te geven en daarna op schakelaar S1 drukken, en de cassetterecorder zal automatisch vooruit spoelen. Dus geen gePOKE meer.

Dit geldt ook voor het saven en de inhoudsopgave. Bij wissen gaat het ook, maar pas hierbij op, want de cassette moet dan op de juiste plaats staan.

De schakeling bestaat uit een 'one shot' (IC1) die gedurende 0...5 seconden laag is op zijn uitgang na indrukken van S1. Daardoor wordt T1 opengestuurd, die op zijn beurt weer T2 openstuurt en het relais zal opkomen. Hierdoor wordt de motor stilgezet. De P2000 denkt nu dat hij aan het begin van de band staat en zal vooruit gaan spoelen. De tijd dat RE-1 onderbroken moet zijn om het geheel goed te laten functioneren kan men instellen met R8. In de meeste gevallen zal de middenstand een goed resultaat geven.

Men mag de tijd niet te kort instellen daar anders de P2000 niet zal reageren en gewoon zal blijven terugspoelen.

Voor het relais kan men een DIL-type kiezen maar ieder ander relais kan ook worden gebruikt. Hou wel rekening met een maximum collectorstroom van 400 mA en dat het relais 12 V moet zijn. Gebruik eventueel een weerstand in serie met het relais."

In de vorige Nieuwsbrief hebben we stilgestaan bij de voor- en nadelen van de twee soorten printers die er zijn: 'impact' en matrixprinters. In dit artikel zullen we het hebben over het aansluiten van printers op de P2000, want dat geeft nog wel eens problemen.

Stuurcodes naar de printer

De P2000 heeft in standaarduitvoering alleen een seriële uitgang, V24- of RS232-uitgang geheten. Dat is dus de enige mogelijkheid om een printer aan te sluiten. Stuurt de P2000 een 'karakter' naar de printer, dan gebeuren er een aantal dingen. In de eerste plaats wordt het karakter door de BASIC-interpretator bekeken en gewogen. Is het een normale ASCII-code tussen &H20 (32 = spatie) en &H7F (127 = ■), dan gaat het karakter in principe linea recta door naar de printer. In principe, want de P2000 gebruikt voor sommige karakters, zoals # en £, een andere ASCII-waarde dan de meeste printers. Daarom kijkt de P2000 eerst in de printervertaaltabel. BASIC NL gebruikt hiervoor dezelfde tabel als de schermroutines; het beginadres van de vertaaltabel staat op &H6092 en &H6093 (hier staat &H1A21, dus de vertaaltabel begint op &H1A21).

Deze gang van zaken verhindert dikwijls het gebruik van speciale codes die nodig zijn om de printer te instrueren, bij voorbeeld over het te gebruiken lettertype, de instelling van kantlijn en regelbreedte enzovoort. Wilt u bij voorbeeld de linkerkantlijn op 12 instellen met `LPRINT CHR$(27);"M";CHR$(12)`, dan draait de printer alleen het papier door. [`CHR$(12)` ziet de P2000 als 'formfeed'.]

Om dit soort gebeurtenissen te voorkomen kunnen de printeroutines 'transparant' worden gemaakt. Dat gebeurt door op adres &H60A2 een '4' te `POKE`. Rob van der Heij gaf in Philitel de volgende subroutine, die bovendien na afloop van de instructie aan de printer de 'oude' waarde weer op &H60A2 zet:

```
100 X=PEEK(&H60A2):POKE &H60A2,4
110 LPRINT CHR$(27);PR$;
120 POKE &H60A2,X
130 RETURN
```

Let op de puntkomma achter `PR$`; laat u die weg, dan begint de printer telkens op een nieuwe regel na het uitvoeren van deze subroutine. De string `PR$` kunt u nu wegschrijven door met `GOSUB 100` deze routine aan te roepen. Het instellen van de linkerkantlijn op positie 25 gebeurt bij de printer die Rob gebruikt als volgt:

```
PR$="M"+CHR$(25):GOSUB 100
```

Bij mijn printer (de FX-80) gebeurt dat op soortgelijke manier:

```
PR$="l"+CHR$(25):GOSUB 100
```

Hierin is 'l' de kleine letter L (van 'linkerkantlijn'; dus geen cijfer één). 25 is de positie van de linkerkantlijn; hier kan natuurlijk elke andere waarde tussen 0 en 255 worden ingevuld.

`CHR$(27)` in regel 110 is de 'ESCAPE'-code, die de printer vertelt dat wat er op volgt niet moet worden afgedrukt, maar een instructie is.

De regelbreedte

Sommige printers houden zelf de regellengte bij. Die staat bij het inschakelen meestal op 80 karakters per regel, maar u kunt de regelbreedte met een soortgelijke instructie als hierboven veranderen (bij de FX-80 met LPRINT CHR\$(27);"Q";CHR\$(n), waarin n de regelbreedte is). Wie dit probeert kan tot de frustrerende ontdekking komen dat sommige regels, waarin instructies staan, te kort worden. Dat komt doordat de P2000 zo eigenwijs is zelf het aantal karakters per regel bij te houden (op &H6253) en, in tegenstelling tot de printer, ook de stuurcodes mee te tellen. Wilt u "Dit is Belangrijk." afdrukken, dan kan dat bij de FX-80 met:

```
LPRINT"Dit is ";CHR$(27);"G";CHR$(27);"-" ;CHR$(1);"Belangrijk." ;CHR$(27);"-" ;CHR$(0);CHR$(27);"H";
```

Voert u na de laatste puntkomma in: ":PRINT PEEK(&H6253)" dan kunt u meteen de kolomteller van de P2000 uitlezen. Die geeft in dit geval 22 aan. De P2000 heeft dus vier stuurcodes meegeteld.

Bij gebruik van de subroutine gaat dit als volgt:

```
10 LPRINT"Dit is ";:FR#="G":GOSUB100:FR#="-"+CHR$(1):GOSUB 100:LPRINT"Belangrijk." ;:FR#="-"+CHR$(0):GOSUB 100:FR#="H":GOSUB 100
20 PRINTPEEK(&H6253):END
100 X=PEEK(&H60A2):POKE &H60A2,4
110 LPRINT CHR$(27);FR#;
120 POKE &H60A2,X
130 RETURN
```

Dit is Belangrijk.

De codes "G" en "H" betekenen dubbel afdrukken en stoppen daarmee. In regel 20 leest de P2000 de kolomteller uit. Die blijkt op 18 te staan. Dit wil zeggen dat nu alleen de echte karakters en niet de stuurcodes zijn geteld.

Dit is een tweede voordeel van het gebruik van de genoemde subroutine: de P2000 telt de stuurcodes niet mee (de printer deed dat toch al niet). Zou de P2000 dat wel doen, zoals bij normale LPRINT-opdrachten, dan genereert hij zelf CR en LF, voordat de regel vol is.

Het probleem van de wachttijden

De seriële uitgang van de P2000 heeft nog enkele nadelen. De BASIC-routines, die tekst naar de printer sturen, maken gebruik van enkele Monitor-routines en die laatste hebben een paar ingebouwde wachtlopen. Voordat de P2000 een karakter naar de printer stuurt kijkt hij even of de printer aan staat en na het versturen van het karakter wacht hij wat langer, 10 tot 12 seconden. Kan hij binnen die periode geen nieuw karakter aan de printer kwijt, dan krijgt u een foutmelding en wordt het programma afgebroken.

Bij veel printers geeft dit laatste problemen, vooral als de afdrukeenheid voorzien is van een buffer. Heeft de printer geen buffer, dan accepteert hij de karakters stuk voor stuk. Het afdrukken van één karakter kost altijd minder dan 10 à 12 seconden, dus voordat de P2000 op tilt slaat kan hij het volgende karakter kwijt.

Heeft de printer een zeer grote buffer, dan kan de hele tekst die u wilt afdrukken daar in één keer in worden opgeborgen. Ook dan zijn er geen problemen.

Die problemen komen pas als de printer een niet te grote buffer heeft en bovendien tamelijk traag afdrukt. Ik bezit bij voorbeeld een tot printer verbouwde IBM schrijfmachine, die een buffer van 96 karakters heeft en circa 15 tekens per seconde kan afdrukken. Het gebeurt vaak, vooral als ik gebruik maak van de trage tabulator, dat de printer binnen 10 seconden nog geen nieuwe karakters accepteert omdat hij nog niet zo ver is.

Dit probleem kan worden omzeild door zelfgemaakte printerroutines te gebruiken, die een veel langere of zelfs een oneindige wachttijd hebben. Die dus desnoods tot in de eeuwigheid blijven wachten totdat de P2000 weer een paar karaktertjes aan de printer kan slijten. Het gaat nu te ver om deze routines hier te beschrijven. Die houdt u nog te goed.

Een ander probleem van de seriële uitgang is dat de P2000 er maar één heeft. Gebruikt u die voor Viditel, dan kunt u niet tegelijkertijd een printer aansluiten om de plaatjes af te drukken.

Een elegante oplossing hiervoor is het gebruik van een Centronics-interface ofwel parallel-I/O-module in de tweede sleuf van de P2000.

Paralleluitgang voor de P2000

Er zijn verschillende leveranciers van een parallel-I/O-module. Voor zover ik weet zijn dat Miniware en Sanecal (zie het overzicht van toeters en bellen in Nieuwsbrief nr. 8). Omdat ik toevallig de modules van Miniware tot mijn beschikking heb (de M2005 en de M2003; de laatste heeft ook nog een extra seriële in- en uitgang) zal ik daar wat meer over vertellen. I/O-modules van andere herkomst zullen niet wezenlijk verschillen, ook niet in het gebruik.

Zo'n module heeft drie voordelen: u houdt de V24-uitgang van de P2000 vrij voor andere zaken (bij voorbeeld Viditel), u kunt een printer gebruiken die alleen een Centronics-interface heeft en u hoeft uw printer dus niet speciaal te laten uitrusten met een seriële interface, en de parallel-I/O-module gebruikt zijn eigen printerroutines, waarvan u de wachttijd, die in acht wordt genomen totdat de printer klaar is om de volgende karakters te ontvangen, binnen zekere grenzen in de hand hebt.

Na het aanbrengen van de module moet u de printerroutines 'omleiden'. Dat kan door op de adressen &H60C6...60C8 het beginadres van de nieuwe printerroutines te zetten. Normaal staat hier C3, 12, 1A (JP 1A12); dit moet worden &H9F00, &HCFO0 of &HFF00, afhankelijk van de geheugenomvang van uw P2000. Het navolgende programma van Miniware zoekt dit voor u uit en zet de printerroutines boven in het geheugen. U moet dit programma steeds als eerste laden, als u de paralleluitgang voor de printer wilt gebruiken. Nadat het geRUND is, kan het worden overschreven door een ander programma.

```

2 REM *****
3 REM * CENTRONICS PRINTER DRIVER VOOR MINIWARE MODULES M2003/M2005 *
4 REM *****
5 REM afhankelijk van de geheugengrootte wordt de driver geladen op:
6 REM 16 kB: &H9F00
7 REM 32 kB: &HCFO0
8 REM 48 kB: &HFF00
10 POKE&H60C7,0:ON PEEK (&H605C) GOTO 20,30,40
20 CLEAR 100,&H9F00-1:ST=&H9F00:POKE &H60C8,&H9F:GOTO 50
30 CLEAR 100,&HCFO0-1:ST=&HCFO0:POKE &H60C8,&HCF:GOTO 50
40 CLEAR 100,&HFF00-1:ST=&HFF00:POKE &H60C8,&HFF:GOTO 50
50 FOR I=0 to &H33:READ A$:POKE ST+I,VAL("&H"+A$):NEXT
100 DATA C5:      REM   PUSH BC      Zet BC op de stack
110 DATA 4F:      REM   LD C,A       Zet het te printen karakter in C
120 DATA C5:      REM   PUSH BC      Zet de te printen waarde op stack
130 DATA D5:      REM   PUSH DE      Zet DE op stack
140 DATA 06,02:   REM   LD B,02      Laadt B met de waarde voor time out
150 DATA 37:      $01 REM   SCF        Zet de carryflag
160 DATA DB,47:   REM   IN 47        Lees de status van de printer
170 DATA CB,4F:   REM   BIT 1,A      Test of de printer klaar is
180 DATA 20,04:   REM   JR NZ,$03    Printer aan, dan verder naar $03
190 DATA DB,47:   REM   IN 47        Lees de status nogmaals
195 DATA CB,4F:   REM   BIT 1,A      Test nogmaals of printer klaar is
200 DATA 28,07:   $03 REM   JR Z,$02    Printerstatus ok, dan naar $02
210 DATA 13:      REM   INC DE       Verhoog time out teller met 1
220 DATA 7A:      REM   LD A,D       Zet D in A voor testen eindwaarde
230 DATA B3:      REM   OR E         Vergelijk A en E en zet vlaggen
240 DATA 20,EE:   REM   JR NZ,$01    Waarde nog niet bereikt dan opnieuw
250 DATA 10,EC:   REM   DJNZ,$01     Verlaag waarde B met 1
260 DATA 3F:      $02 REM   CCF        Complementeer carry flag
270 DATA 38,13:   REM   JR C,$04     Time out bereikt en printer nog
                                         niet klaar, dan verder naar $04

290 DATA F3:      REM   DI           Zet interrupts uit
300 DATA 37:      REM   SCF          Zet carry flag
310 DATA DB,47:   REM   IN 47        Lees status printer
320 DATA CB,67:   REM   BIT 4,A      Test of printer klaar is voor
                                         ontvangen van het karakter

340 DATA 28,0B:   REM   JRZ,$04     Printer niet klaar dan naar $04
345 DATA B7:      REM   OR A         Zet vlaggenregister
350 DATA 79:      REM   LD A,C       Zet karakter terug in A
360 DATA D3,49:   REM   OUT 49       Zet strobe uit
370 DATA D3,46:   REM   OUT 46       Zet data in Centronics register
380 DATA D3,48:   REM   OUT 48       Zet strobe aan
390 DATA D3,49:   REM   OUT 49       Zet strobe weer terug
400 DATA FB:      $04 REM   EI         Enable interrupts
410 DATA D1:      REM   POP DE       Herstel oude waarde van DE
420 DATA C1:      REM   POP BC       Dummy-instructie voor stackgrootte
430 DATA C1:      REM   POP BC       Herstel oude waarde van BC
440 DATA C9:      REM   RET          Terug naar BASIC-programma

```

Als u geen zin hebt om dit allemaal in te typen, dan kunt u de REM-regels 2...8 en alle commentaar in de regels 100...440 weglaten. Dan kunnen alle DATA in één regel worden opgenomen, gescheiden door komma's. Dat ziet er dan zo uit:

```

10 POKE &H60C7,0:DNPEEK(&H60C5)GOTO 20,3
0,40
20 CLEAR 100,&H9F00-1:ST=&H9F00:POKE&H60
CB,&H9F:GOTO50
30 CLEAR 100,&HCFO0-1:ST=&HCFO0:POKE&H60
CB,&HCF:GOTO50
40 CLEAR 100,&HFF00-1:ST=&HFF00:POKE&H60
CB,&HFF:GOTO50
50 FORI=0 TO &H33:READ A#:POKE ST+I,VAL(
"&H"+A#):NEXT
100 DATA C5,4F,C5,D5,06,02,37,DB,47,CB,4
F,20,04,DB,47,CB,4F,28,07,13,7A,B3,20,EE
,10,EC,3F,38,13,F3,37,DB,47,CB,67,28,0B,
B7,79,D3,49,D3,46,D3,48,D3,49,FB,D1,C1,C
1,C9

```

Het kan echter nog korter:

```

20 CLEAR 100,&H6F00+PEEK(&H605C)*&H3000:
POKE&H60C7,1:POKE&H60CB,PEEK(&H63B9)
50 FORI=1 TO &H34:READ A#:POKEPEEK(&H63B
9)*256+I,VAL("&H"+A#):NEXT
100 DATA C5,4F,C5,D5,06,02,37,DB,47,CB,4
F,20,04,DB,47,CB,4F,28,07,13,7A,B3,20,EE
,10,EC,3F,38,13,F3,37,DB,47,CB,67,28,0B,
B7,79,D3,49,D3,46,D3,48,D3,49,FB,D1,C1,C
1,C9

```

Omdat dit programma in RAM staat, kunt u het gemakkelijk veranderen. U kunt bij voorbeeld de wachttijd wijzigen door in regel 140 van het volledige programma register B tot een hogere waarde te laden. De maximale waarde is &HFF, en de computer wacht dan een aantal minuten voordat hij de foutmelding "geen printer" geeft. U moet wel een vrese-lijk trage printer hebben, wil dat niet genoeg zijn. Regel 140 wordt dan: 140 DATA 06,XX: waarin XX de waarde 01...FF kan hebben.

Opmerking: een Centronics-interface lost de problemen, die wij aan het begin van dit artikel hebben beschreven, niet op. Wilt u stuurcodes naar de printer sturen zonder dat de P2000 zich daar mee bemoeit, dan zult u net als bij gebruik van de V24-interface de subroutine moeten gebruiken.

Volgende aflevering

Dit verhaal is niet in overeenstemming met wat ik in de vorige Nieuws-brief heb aangekondigd. Voordat u aan tekstverwerking kunt beginnen zult u echter eerst een printer op uw P2000 moeten aansluiten, die doet wat u van hem verlangt.

Maar in de volgende aflevering zullen we het echt hebben over het invoeren van tekst, zowel met BASIC-programma's als met de speciale tekstverwerkingsmodules die voor de P2000 verkrijgbaar zijn. Verder hoop ik enkele BASIC-programma's te geven waarmee teksten, ingevoerd met het Familiegeheugen of de Philips Tekstbewerkingsmodule onder BASIC NL kunnen worden afgedrukt, waarbij u zelf kunt kiezen hoe u die teksten afgedrukt wilt hebben.

Rob Geutskens

"Ik ben maar een gewone jongen" Marcel Bruins
"Ik ben leek, en ik wens dit te blijven" Erica Bruins

Na in een eerdere Nieuwsbrief een verhaal over Klaas Robers nu een verhaal over Erica en Marcel Bruins; twee mensen aan wie de P2000gg veel te danken heeft.

Marcel is geboren in Hengelo op 21 januari 1951. Toen al was hij een stevige rekel; hij woog 8 pond! In zijn eerste levensjaren was hij een echte belhamel. Hij deed altijd dingen die hij niet mocht doen, maar die hij wel wou doen en dus ook deed.

Na de lagere school ging hij naar de LTS waarvan hij na drie en een half jaar door onenigheid met een docent werd verwijderd.

"Vader zat met de handen in het haar. Hij wist niet wat hij met me aan moest; zo'n knul groeit op voor galg en rad..."

Nadat hij 2 dagen had gewerkt als puntlasser in een productiebedrijf ging hij bij zijn vader in de garage werken. Marcel heeft in die tijd gestudeerd in Voorschoten en behaalde zijn BOVAG-diploma en het middenstandsdiploma.

In de periode dat hij bij zijn vader werkte heeft hij Erica leren kennen. De kennismaking verliep op een nogal ongebruikelijke manier: Tijdens een diner wilde Marcel de menu-kaart van Erica afpakken met een fikse diskussie als gevolg. Erica: "Ik pakte hem bij z'n haar en heb hem nooit meer losgelaten."

Erica is op 29 december 1952 geboren in Den Haag. Ze heeft de lagere school in Lochem doorlopen, vervolgens in Almelo de INAS-opleiding, waarna ze op verschillende plaatsen in Nederland stage heeft gelopen. Daarna werkte ze in Hengelo in de verpleging. In die tijd leerde ze Marcel kennen.

Terwijl Marcel nog steeds in vader's garage werkte is Erica gaan werken als inkoopster/verkoopster in een winkel voor positie- en babykleding.

Op 19 september 1972 kregen Erica en Marcel een inval; Ze hebben zich op die dag pardoes verloofd. Kort daarna kwam er in Hengelo een woning voor hen vrij, en op 22 november 1973 zijn ze getrouwd.

In 1974 ging Marcel bij Akzo in Delfzijl werken, waar hij een baan in de proces-industrie had. Na 3 jaar werd echter het bedrijf opgeheven, waardoor hij werd overgeplaatst naar MCN. Hier beleefde hij zijn eerste kennismaking met computers. Dat betrof zgn. PLC's (Process Logic Control); computers speciaal voor proces-besturing.

In deze tijd hebben Erica en Marcel een pleegzoon gehad. Deze zoon, Pieter (11 jaar), heeft 14 maanden bij hun gewoond. Ze hebben voor hem veel liefde en geduld op moeten brengen, wat echter helaas niet baatte. "Wij willen graag voor iemand zorgen. Het is fijn om liefde te geven en te ontvangen, maar het was een erg moeilijke tijd. We hebben er erg veel van geleerd."

Na 2 jaar bij MCN kon Marcel bij Vredestein in Enschede team-leider (over 32 man) worden. Ook daar werkte men met PLC's, doch deze waren veel moderner. Na 2 jaar kon hij bij een andere firma wachtchef worden. Hij zou er de leiding krijgen over een hele afdeling, maar werd in zijn proeftijd ontslagen. "Dat was financieel een heel behoorlijke verbetering. Maar de eerste maandagmorgen dat ik er 2 uur vroeger moest beginnen heb ik me verslapen. Ach ja, ik had net 5 dagen mijn P2000. Daarmee was ik 'snachts tot een uur of twee, drie bezig en 'smorgens om 6 uur moest ik er zijn en om 9 uur kwam ik aanrennen. 'sMiddags om 3 uur kon ik opstappen, toen hoefde het niet meer."

Een tijd van 20 maanden werkloosheid brak aan waarin veel tijd aan de P2000 en de P2000gg werd besteed. Sinds 1 oktober 1982 werkt Marcel bij een P2000-dealer in Midden-Limburg.

De hobby's.

Marcel heeft veel hobby's gehad, doch steeds maar EEN tegelijk. Hij bouwde zelf een modelboot met afstandsbesturing waar hij veel plezier aan heeft beleefd. Daarna leefde hij zich uit in de fotografie, waarvoor Erica poseerde. De volgende hobby was een aquarium, dat ze nu nog steeds hebben. Weer later kocht hij een elektronisch orgel waar vooral Erica vele uren plezier aan heeft beleefd. Marcel vond al na een week dat hij geen talent had ("Ik heb te dikke vingers"), maar Erica heeft een half jaar lessen gevolgd. Marcel legde zich toen meer op de electronica waar hij altijd al erg in geïnteresseerd was. Als logisch gevolg daarvan ontstond zijn huidige hobby: De P2000.

Erica's hobby's zijn tekenen, schilderen en borduren. Kortom hobby's waar ze haar creatieve geest in op kan laten gaan. Verder houdt ze van lezen, doch het belangrijkste voor haar is het hebben van contact met mensen. Toen Marcel geen tijd meer had voor de gebruikersgroep nam ze vele werkzaamheden van hem over, waarbij het contact met de andere gebruikers haar voornaamste drijfveer was.

De P2000.

Marcel heeft 6 maanden nodig gehad om te beslissen welke computer hij zou kopen. Van meet af aan wist hij dat hij met een goede computer moest starten. Desondanks maakte hij bijna een verkeerde keuze.....

Tijdens de HCC-dagen in het Turfschip in Breda zou Marcel een computer kopen. Er was daar een aanbieding van een ITT-computer. Snel overleg was geboden en daarom belde hij met Erica om te overleggen. Nadat ze samen hadden besloten dat de aanschaf oke was liep hij terug om te konstateren dat de ITT inmiddels verkocht was! "...en toen heb ik gelijk de P2000 gekocht omdat ik de normale prijs van de ITT veel te hoog vond. Mijn eisen waren kleur en tenminste 16K ram, en voor mij was de P2000 dus de beste keuze...."

Toen Marcel de P2000 kocht was hij volkomen leek op computer-gebied. Hij ging naar de HCC-bijeenkomsten op de TH Twente teneinde daar wat meer informatie over de P2000 te krijgen. Met hulp van Roelof Beverdam die al veel ervaring had met de Exidy zette Marcel zijn eerste schreden in (thuis-) computerland. Op de TH was bovendien een vrij toegankelijke bibliotheek waar Marcel veel tijd heeft doorgebracht met het lezen van computerboeken. "... en soms stond ik daar te staren naar de grote computers waar de studenten gebruik van mochten maken, maar daar durfde ik nooit aan. Daar durfde ik nooit een toets aan te raken...."

Op een gegeven moment vroeg Roelof (toen voorzitter NASCOM-gg) waarom Marcel geen gebruikersgroep oprichtte. Hij werd hierin verder aangespoord door Rein Heesterman (toen voorzitter OSI-gg) maar daartoe voelde Marcel zich eigenlijk niet geroepen. Uiteindelijk plaatste hij een advertentie in de HCC-Nieuwsbrief om de interesse in een gebruikersgroep voor de P2000 te polsen. Op de eerste bijeenkomst van de P2000 gebruikersgroep waren ca. 45 mensen aanwezig, waaronder enkele mensen van Philips. Op deze bijeenkomst werd het eerste bestuur benoemd. Marcel werd tegen wil en dank tot voorzitter gebombardeerd, iets wat hij zelf niet zag zitten omdat hij zichzelf maar een "gewone jongen" vond. Na de eerste bestuursverkiezingen had de P2000gg nog geen penningmeester. "Ik keek toen op de ledenlijst en zag dat Frans Moes accountant was. Die moest dus penningmeester worden waartoe ik hem vrij snel omgepraat had."

Het bestuur van de gg was een erg prettige groep waarvan Marcel veel steun voor zijn activiteiten van kreeg. In de tijd dat de P2000gg nog zo'n 200 leden had kende hij iedereen persoonlijk. Door de groei van de laatste tijd is dat sterk vervaagd wat Marcel erg betreurt (Erica knikt instemmend). "Computer-hobby-isten zijn altijd erg enthousiast. Ik vind het leuk om eens een praatje met de mensen te kunnen maken en bijvoorbeeld wat te filosoferen over de toekomst."

Toen Marcel bij de P2000-dealer ging werken heeft hij het voorzitterschap eraan gegeven. "Je kunt dat niet combineren. Je moet je OF 100% voor de hobbyist inzetten OF er je werk van maken. Een combinatie zou alleen maar ten nadele van de leden uitwerken." Erica heeft toen het contact met de leden van hem overgenomen, maar hoewel ze wel het een en ander weet over de P2000 werkt ze er zelf niet mee. "Ik vertegenwoordigde de leek in de P2000gg, en ik wens(-te) leek te blijven."

Na de prijsverlaging van augustus 1983 ging het echter allemaal veel te hard en werd het werk voor de P2000gg meer dan een dagtaak. "Ik kreeg op een dag soms wel 60 telefoontjes van mensen die door Philips naar mij werden doorverwezen omdat Philips zelf het antwoord op de vragen niet wist. Het werd me gewoon teveel. Nadat voor vervanging was gezorgd ben ik er dan ook helemaal mee gestopt. Ik krijg nog steeds regelmatig telefoontjes van mensen met vragen over de P2000 en soms vind ik dat best leuk."

Anita Renkien.

Assembleurs opgelet. Er is door R. K. Eijnhoven, de maker van het assembler-programma op cassette A212, een assembler ontwikkeld die in een ROM-module moet worden ondergebracht. Deze assembler (met monitor) komt dus in de plaats van de BASIC-interpreter in sleuf 1 van de P2000. De assembler biedt in grote lijnen dezelfde mogelijkheden als de cassette-versie, maar heeft een paar extra's. Hieronder een korte beschrijving.

Deze assembler in ROM-pack is verkrijgbaar bij:

AVO Techniek
Postbus 67
Ginderover 29
5590 AB Heeze (NB)
Tel.: (04907) 2285

Wat heeft de assembler in ROM te bieden ?

Het voornaamste verschil met de cassette-assembler is natuurlijk dat de ROM-assembler in de plaats komt van de BASIC-module, en dus geen extra geheugenruimte vraagt. U houdt dus veel meer RAM over voor tekst, labels, het geassembleerde programma en dergelijke. Daardoor kunt u deze assembler, in tegenstelling tot de cassette-versie, ook op een 16K-machine gebruiken.

Een ander belangrijk verschil is de werking van de RESET-knop die, zoals u weet, meestal heel gevaarlijk is. Als u daar op drukt, bent u alles kwijt wat in het geheugen van de P2000 staat. Bij de ROM-assembler werkt de RESET-knop echter niet destructief. Hebt u een machinetaalprogramma gemaakt met een fout erin, waardoor het vastloopt als u het laat werken, dan is een druk op de RESET-toets voldoende om de zaak weer los te maken, zonder dat de geheugeninhoud verloren gaat. U kunt dus verder gaan waar u gebleven was en de fout herstellen, tenzij uw programma de boekhouding of de tekst-file van de assembler heeft vernield.

Bij programma's in machinetaal is het vaak praktisch gebruik te maken van bepaalde routines, die in BASIC NL aanwezig zijn. Dat geldt met name voor routines die een toets uitlezen en de ASCII-code bepalen, die een karakter op het scherm brengen of naar de printer sturen en die een file van cassette laden of daarop wegschrijven. Daarom zijn zes labels gedefinieerd, namelijk COUT (&H104D), CHIN (&H104D), CLOA (&H1059), CSAV (&H105C), PRIN (&H105F) en ASCI (&H1814). De vijf routines en de ASCII-vertaaltabel zitten bij de ROM-assembler op dezelfde geheugenadressen als bij BASIC NL. Dit betekent dat u machinetaalprogramma's, die onder BASIC NL moeten draaien, kunt testen onder de ROM-assembler, en dat u toch gebruik kunt maken van de BASIC-routines.

Extra opdrachten

De ROM-assembler herkent twee opdrachten, die de cassette-versie niet herkent. Dat zijn GLOBAL en LOCAL. Deze opdrachten zijn bedoeld om relocatie-informatie te genereren. De LOCAL-adressen kunnen worden gereloceed; GLOBAL-adressen (bij voorbeeld adressen van het beeldscherm en van de boekhouding) blijven bij relocatie ongewijzigd.

Iedereen die al wat langer met de P2000T werkt kent het verschijnsel van cassette errors zoals "leesfout" en "niet gevonden" of meer van dat 'fraais'.

Normaal is dit niet zo erg, want je weet gewoon dat bewegende delen, wat een draaiende cassette is, slijten.

Maar onlangs ging het toch wel erg fout met de recorder. Het begint met 1 of 2 maal per avond een leesfout, wat opzich niet zo erg is, maar het komt elke avond terug en het neemt steeds ergere vormen aan.

Eerst maak je de recorder schoon en denk je het euvel verholpen te hebben. Maar helaas, de vreugde is van korte duur en het begint opnieuw en groeit het besef dat er iets echt mis is met de recorder.

Nu zijn er verschillende manieren om van dit probleem af te komen, zoals een noodkreet op het prikbord van PHILITEL, waarin dan allerlei mogelijkheden geopperd worden maar voor jou niet de goede. Je kan ook de P2000 naar de service dienst van Philips brengen wat waarschijnlijk een nieuwe recorder kost. (Dit kost ongeveer 500 gulden.)

Voor een ieder die de voornoemde verschijnselen tegen komt en voor de beslissing staat wat te doen.., heb ik het volgende advies. Haal alle stekers en ook de rompack uit de P2000 en keer hem om. Verwijder de schroeven waarmee de bovenkap vast zit en draai het geheel weer terug. Til de kap eraf en ga zitten.

Nu de zwarte vierkante ring met de resetknop losmaken en opzij leggen. De resetknopdraadjes kunnen blijven zitten.

In deze situatie is de cassettehouder waarin de weergave- en wiskop vastzitten vrij gekomen en zijn de 5 draadjes waar het in dit geval omgaat goed zichtbaar. Deze 5 draadjes worden naar links afgevoerd en eindigen in een connectortje op de print onder de recorder.

Vlak naast de koppen zitten ze vast met een nylon bindbandje.

Dit bandje zorgt ervoor dat de draadjes niet aan de koppen trekken wanneer de houder open en dicht gaat.

Dit bandje kan echter ook de oorzaak zijn van bovengenoemde storing.

Het bandje moet nu voorzichtig losgeknipt worden zonder de draden te beschadigen zodat je kunt controleren of ze op deze kwetsbare plek soms al niet gebroken zijn.

Bij ondergetekende was het zwarte draadje in zijn geheel doormidden gebroken en daardoor bovengenoemde ellende veroorzaakte.

Wie nu denkt bij Philips een nieuw draadsetje te kopen komt bedrogen uit. Dit onderdeel heeft bij Philips geen onderdeelnummer en komt zodoende niet voor in het magazijn. Er zijn nu twee mogelijkheden:

1. Je koopt een nieuwe recorder.
2. Je soldeert zelf maar een nieuw draadje in.

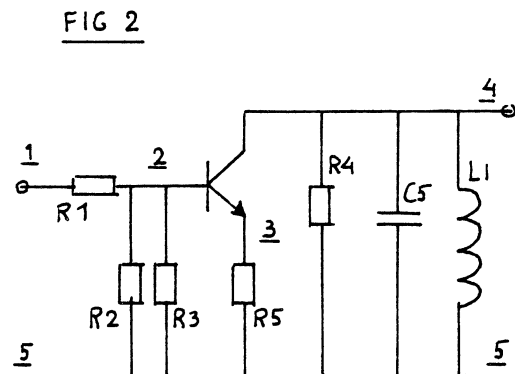
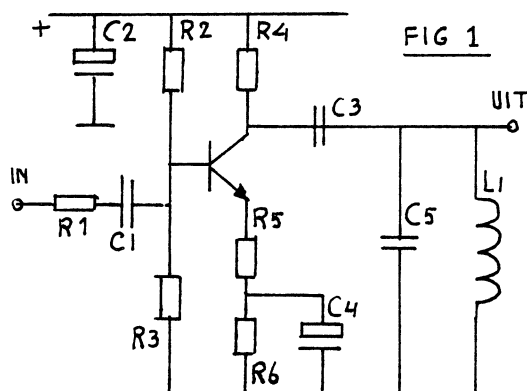
Het laatste zal duidelijk faboriet zijn want het kost praktisch alleen een beetje moeite en een draadje van 10 cm.

Indien deze reparatie of controle noodzakelijk mocht zijn neem er dan de tijd voor en doe voorzichtig. Vergeet vooral niet de spanning eraf te halen en bij montage van de recorder ook het nieuwe bandje niet vergeten. Gebruik eventueel het oude draadje om de andere nee vast te zetten.

Dit zijn nu eens geen spelletjesprogramma's maar rekenprogramma's, bedoeld voor hen die zich met electronica bezighouden.

"NETWERK-ANALYSE" is een universeel programma voor het berekenen van frequentie-karakteristieken en versterking van willekeurige analoge schakelingen; het berekent echter niet de elementen van de schakeling.

Aangezien het om wisselspanningen gaat, moeten eerst in het bestaande schema koppel- en ontkoppelcondensatoren worden kortgesloten. Dit kan indien de impedantie bij de laagst weer te geven frequentie klein is ten opzichte van de ermee in serie of parallel geschakelde onderdelen.



C1 en C3 in fig 1 zijn koppelcondensatoren, C2 en C4 zijn ontkoppelcondensatoren. Het wisselstroomschema komt er dan uit te zien zoals in fig 2.

In dit schema worden alle knooppunten genummerd, beginnend bij 1 aan de ingang; het hoogste cijfer moet voor in- en uitgang gemeenschappelijk zijn, dit is dus meestal aarde. Er mogen geen cijfers worden overgeslagen.

Het programma vraagt nu het soort onderdeel, tussen welke knooppunten, en de waarde. Na invoer van het netwerk wordt gevraagd de uitgang, start en stop frequentie en stapgrootte. Er is een editing mogelijkheid voor fout ingevoerde waarden. Hierna volgt de berekening.

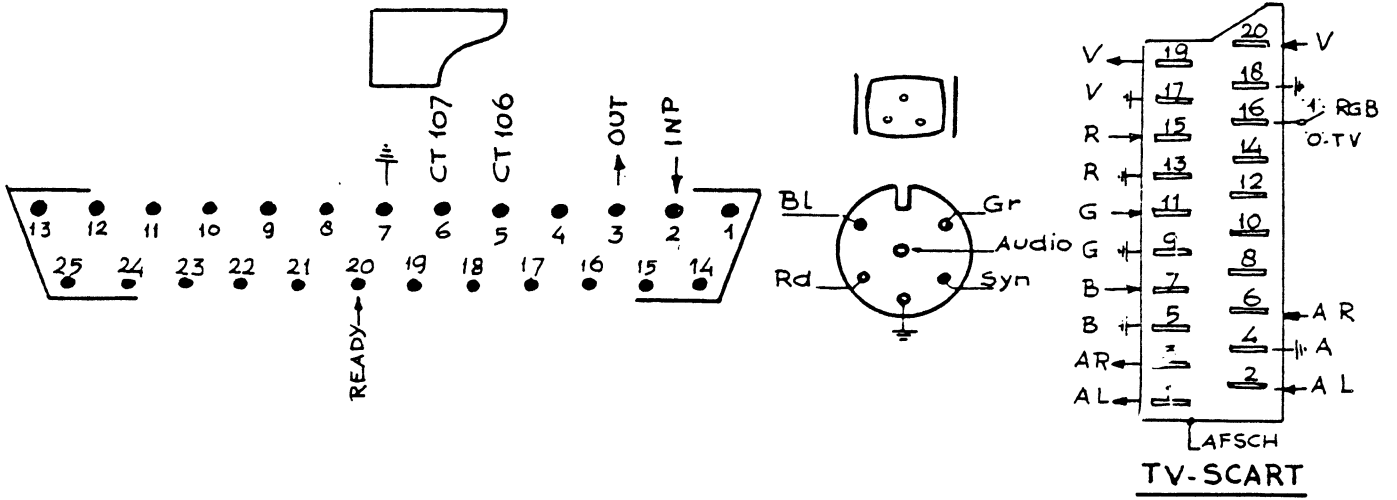
Het maximum aantal knooppunten bedraagt voor een 32 K machine ca. 15. Bij 16 K is het aantal beperkt; door verwijderen van de uitleg (Delete 5000-5920) zijn 6 a 7 knooppunten in te voeren.

De berekeningssnelheid loopt sterk terug bij hoger knooppuntgetal. Zo vraagt een 9-pools actief filter met 15 knooppunten ca. 3 minuten per frequentie; buiten het groot aantal berekeningen dat moet worden uitgevoerd komt dit door de vereiste dubbele precisie van de matrixen.

Het resultaat kan worden uitgeprint.

Het programma "FILTER-KARAKTERISTIEKEN" is speciaal bedoeld voor passieve laag-, band- en hoog- doorlatende filters. Het berekent naast de demping ook de ingangsreflectie en de groeplooptijd als functie van de frequentie.

RGB = RGB 0.7V75Ω
 A = Audio 0.5V
 V = Video 1V75Ω



- Aarde ● 1 ● Aarde
- D0 ● 3 ● 4 ● D1
- D2 ● 5 ● 6 ● D3
- D4 ● 7 ● 8 ● D5
- D6 ● 9 ● 10 ● D7
- A0 ● 11 ● 12 ● A1
- A2 ● 13 ● 14 ● A3
- A4 ● 15 ● 16 ● A5
- A6 ● 17 ● 18 ● A7
- A8 ● 19 ● 20 ● A9
- A10 ● 21 ● 22 ● A11
- A12 ● 23 ● 24 ● A13
- A14 ● 25 ● 26 ● RAMS 2
- MRQ ● 27 ● 28 ● RD
- INT ● 29 ● 30 ● WR
- IOREQ ● 31 ● 32 ●
- M1 ● 33 ● 34 ● RES
- Lock ● 35 ● 36 ● RFSM
- DEW/R ● 37 ● 38 ● 2.5 MHz
- 5 MHz ● 39 ● 40 ● T4M

EXT.

- +5V b 1 a NMI
- D1 2 D0
- D3 3 D2
- D5 4 D4
- D7 5 D6
- A1 6 A0
- A3 7 A2
- A5 8 A4
- A7 9 A6
- A9 10 RES
- A11 11 M1
- A12 12 IOREQ
- 13 CARSEL2
- MRQ 14 +12V 13 LFinp
- Aarde b 15 a Aarde
- WR 14 -12V

SLOT 2

SLOT 1

In navolging van het gelijkkluidende artikel uit Nieuwsbrief 8 hierbij weer enkele vragen met de bijbehorende antwoorden.

- Iedere keer als ik op RESET heb gedrukt krijg ik met "?RND(1)" het zelfde getal. Is daar wat aan te doen?

Ja. De random-waarde wordt in sterke mate bepaald door de inhoud van geheugenadres &h6238. Door daar steeds een andere start-waarde in te zetten krijgt u steeds een ander resultaat. In een programma kunt u dit probleem oplossen door vooraan in het programma de opdracht "POKE&H6238,PEEK(&H6010)" op te nemen.

- Ik heb me 4 weken gelden aangemeld als lid van de P2000gg, maar ik heb tot op heden niets gehoord. Krijg ik bericht van de P2000gg?

Ja, u krijgt zo spoedig mogelijk bericht van ons. Vergeet echter niet dat wij onze activiteiten in onze vrije tijd doen en dat het daarom niet mogelijk is om elke aanmelding individueel te behandelen. De gemiddelde tijd die er overheen gaat voor u een reactie ontvangt is ca 4 weken, maar uitschieters tot 6 weken zijn zeer wel mogelijk. Heb a.u.b. geduld; we doen ons best.

- Heeft de P2000gg ook een clubblad?

Ja, u zit er immers momenteel in te lezen!

- Ik heb al vaker gewerkt met het bewaren van arrays op cassette, maar tot nu toe is het me nooit gelukt om letters of strings te bewaren.

U kunt met behulp van het programma "Stringsave" (cassette A213) ook letters of strings op cassette bewaren. In dit programma staat verder beschreven hoe u de daarvoor benodigde programma-opdrachten in uw programma moet opnemen.

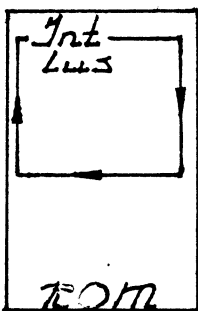
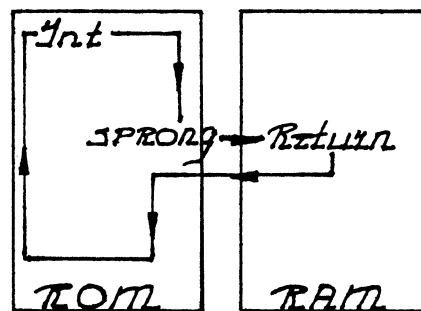
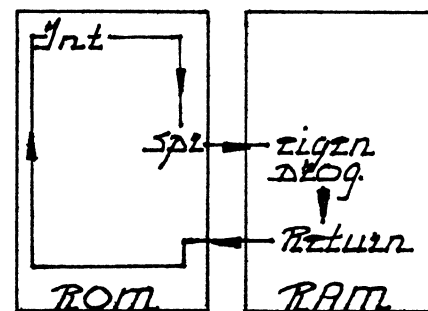
- Ik heb de avonturenprogramma's "Schatzoeken" en "Piraat", maar ik kom er niet uit. Is er wel een uitweg?

Ja, er is een uitweg in deze programma's, maar ik heb ze zelf nog nooit gevonden. Het zou ook niet leuk zijn om de clou te verraden, want daarna is het programma immers geen avontuur meer.

- Is er een catalogus van alle programma's die ik bij de P2000gg kan bestellen?

Ja, u kunt bij de radiohandelaar gratis een folder afhalen, getiteld "Programma's voor de P2000T thuiscomputer". Daarin staan alle bandjes van de P2000gg vermeld.

De Interpreter van de P2000 is het gedeelte dat de BASIC-opdrachten uitvoert. Als we dus de BASIC-opdrachten willen wijzigen of nieuwe opdrachten willen toevoegen, moeten we ingrijpen in de Interpreter. De Interpreter is opgeslagen in het ROM-geheugen, hetgeen wil zeggen dat er alleen uit gelezen kan worden terwijl er niets veranderd of bijgevoegd kan worden. Het lijkt dus onmogelijk, en zo was het ook met de oude BASIC Interpreter. Echter met de invoer van de nieuwe BASIC-NL zijn er extra "sprongpunten" ingelast, waardoor het toch mogelijk wordt. Bij zo'n sprongpunt aangekomen, springt de computer even vanuit ROM naar een adres in RAM; meestal staat hier dan "RETURN" zodat de computer meteen weer terug springt naar ROM. Het RAM-adres kunt u wijzigen: u "knipt" de spronglus door en voegt daar een eigen programma in (zie figuur 1).

OUDE BASICBASIC-NLBASIC-NL uitbreiding

Omdat de Interpreter een machinetaal programma is, moet ook het toegevoegde programma machinetaal zijn: het kan dus niet in "BASIC". Voor dit invoegen van programma's zullen we een viertal voorbeelden behandelen:

- 1) Print de hoeveelheid vrij geheugen met behulp van 1 toets.
- 2) Verplaats de cursor met de cursortoetsen.
- 3) Doorloop een BASIC-programma stap voor stap.
- 4) Nieuw geluidscommando "BOEM"

Er zijn meerdere sprongadressen in BASIC-NL; voor ons doel zullen we er twee gebruiken, nl. 60D0 en 60D9. Maar eerst gaan we wat dieper in op de werking van de Interpreter. De Interpreter bestaat uit twee aparte (gescheiden) lussen. Als de computer op input wacht, dat wil zeggen als u bezig bent met een BASIC-programma in te typen, dan bevindt de computer zich in de eerste lus. Deze heet dan ook de INPUTFASELUS. Als de Inputfase klaar is (dat wil zeggen de BASIC-regel is af en u hebt de Returntoets ingedrukt) en u typt nu "RUN" in, dan springt de computer naar de tweede lus om het ingetypte programma te gaan uitvoeren. Deze lus heet daarom EXECUTIEFASELUS. In beide lussen bevindt zich een bruikbaar sprongadres. De eerste toepassing (=Programma 1) gebruikt het sprongadres van de Inputlus: we willen nl. tijdens het intypen van het programma

weten hoeveel vrij geheugen er nog aanwezig is.

De toepassingen in de Programma's 2,3 en 4 gebruiken het sprongadres van de EXECUTIELUS, want dit zijn toepassingen die tijdens de uitvoer van een BASIC-programma plaats moeten kunnen vinden. Na de terugkeer vanuit de eigen (toegevoegde) Programma's 1,2,3 en 4 moet gezorgd worden dat de Interpreter zijn oude programmaloop ongestoord kan hervatten. Hiervoor is nodig dat de stack en (enkele) registers de juiste waarden bevatten. Zo moet bijvoorbeeld het HL-register het adres bevatten dat een plaats voor de volgende in BASIC uit te voeren opdracht ligt. Het sprongadres in de Inputfase (=3 byte 60D9 t/m 60DB) bevat normaal C3 56 19 = JP 1956, hetgeen een sprong is naar de Keyboardroutine. Deze wacht tot een toets wordt ingedrukt en zet dan vervolgens de toetscode in ASCII in de Accu. De sprongadressen worden als volgt gewijzigd:

Sprongadres	60D9	60DA	60DB	
Normale inhoud	C3	56	19	= JP 1956
Inhoud voor programma 1	C3	50	61	= JP 6150
Sprongadres	60D0	60D1	60D2	
Normale inhoud	00	00	00	
Inhoud voor programma 2	CD	74	61	= CALL 6174
Inhoud voor programma 3	C3	92	61	= CALL 6192
Inhoud voor programma 4	CD	AE	61	= CALL 61AE

BASIC HOOFDPROGRAMMA: "NIEUWE COMMANDO'S"

Als u het BASIC-programma intypt en "RUN" zegt, kunt u direkt over elk van de vier nieuwe commando's beschikken. De allereerste keer worden alle DATA gelezen en via POKE op de juiste plaats weggezet (regel 50,60 en 70). Als het de eerste keer is, wordt dit bijgehouden door een vlag op 6057; dat is een adres dat de eerste keer 0 is, waarna het 1 wordt (regel 45).

De vier programma's worden opgeslagen in een lege kladblokruimte (van 6150-61FF, zie Nieuwsbrief No. 8). Dit is een onderdeel van de BASIC-huishoudruimte dat toch ongebruikt is. Voordeel hiervan is dat ze de gewone BASIC-programma's niet in de weg staan (er is geen extra CLEAR voor nodig); terwijl het dus ook geen vrije geheugenruimte kost.

De programma's zijn "relocatable", dat wil zeggen zonodig kunt u ze toch op een andere plaats opbergen. De afzonderlijke programma's zijn vrij klein doordat gebruik wordt gemaakt van de al aanwezige ROM-routines die het eigenlijke werk doen. Als alle DATA ingelezen zijn kunt u het hele BASIC-programma weggooien (met bijvoorbeeld NEW). Om programma 1 te gebruiken moet u intypen PRINTUSR1(1). Het USR1 programma is een hulp-

routine die in een ander ongebruikt gedeelte staat (6360 t/m 63B3); deze routine vult de 3 byte van het sprongadres in een keer met het nieuwe (gewenste) adres (NB. Dit moet in machinetaal gebeuren. Want zou u dit proberen via 3 achtereenvolgende POKE-opdrachten in BASIC, dan gooit de 1e POKE de executielus in de war die u nog moet gebruiken voor de volgende POKE'S). U kunt de oorspronkelijke (gewone) BASIC-NL toestand herstellen met bijvoorbeeld PRINT USR1(0); de machinetaal programma's blijven dan ongestoord zitten en u kunt ze later naar believen weer oproepen met PRINT USR1 (n) (n= 1,2,3 of 4).

Op regel 70 worden de noten in de noottabel gezet. Deze tabel begint vlak achter de vulroutine; er is plaats voor 18 noten (63A2 - 63B3); de laatste noot moet altijd een 0 zijn.

Hieronder volgt het volledige BASIC-HOOFDPROGRAMMA gevolgd door een aparte bespreking en toelichting van elk van de vier machinetaal programma's. Als er iets niet duidelijk is, en/of als u verbeteringen of uitbreidingen heeft, belt u me dan gerust even op.

Dick van den Berge
Hazerswoude
Tel. 01728-8239

PROGRAMMA: NIEUWE COMMANDO'S

```

030 DEFUSR1 = &H6372 : REM USR1 aanroep voor vulroutine
040 PRINT "1 = FREE MEM":
    PRINT "2 = CURSORBESTURING":
    PRINT "3 = SINGLE STEP":
    PRINT "4 = GELUID":
    PRINT "0 = ALLES UIT"
045 IF PEEK (&H6057)=0 THEN PRINT"evengeduld" : POKE&H6057,1
    ELSE GOTO 80
050 P=&H6150 : N=160 : GOSUB 150 : REM POKE DATA PROGRAMMA
    1 t/m 4
060 P=&H6360 : N=18+48 : GOSUB 150 : REM DATA Sprongadres +
    vulroutine
070 P=&H63A2 : N=10 : GOSUB 150 : REM NOTENTABEL
080 INPUT "kies maar";Z
090 ON Z+1 GOTO 100,110,120,130,140
099 END
100 PRINT USR1(0) : END : REM zet alles UIT (alleen BASIC-NL)
110 PRINT USR1(1) : END : REM zet PROGRAMMA 1 aan
120 PRINT USR1(2) : END : REM zet PROGRAMMA 2 aan
130 PRINT USR1(3) : END : REM zet PROGRAMMA 3 aan
140 PRINT USR1(4) : END : REM zet PROGRAMMA 4 aan
150 FORI=1 TO N : READ A$ : A$="&H"+A$
155 POKE P+I-1, VAL(A$) : NEXTI :RETURN

200 DATA CD,56,19,FE,7,C0,3E,4,32,B6,63,D5,CD,3F,4E,CD,31,38

```



```

210 DATA CD,9E,3E,36,2,11,0,50,1,6,0,ED,B0,EB,36,7,D1,C9
: REM DATA voor PROGRAMMA 1

220 DATA CD,29,0,C8,D8,E5,CD,56,19,FE,10,28,C,FE,11
230 DATA 28,8,FE,12,28,4,FE,13,20,3,CD,86,13,E1,C9
: REM DATA voor PROGRAMMA 2

240 DATA E5,2A,5A,62,CD,9E,3E,36,2,11,0,50,1,6,0,ED,B0
250 DATA EB,36,7,E1,CD,26,0,D0,C3,3,25
: REM DATA voor PROGRAMMA 3

260 DATA E5,7E,FE,0,20,4,23,23,23,23,23,7E,FE,42,28,2,E1,C9
270 DATA C1,E5,23,7E,FE,3A,28,D,FE,0,28,9,FE,CA,20,F2,E1,2B
280 DATA C3,13,28,21,7,4,22,BC,60,21,A2,63,23,7E,CD,86,13
290 DATA 3A,BC,60,B7,20,F5,E1,23,23,23,23,18,BE
: REM DATA voor PROGRAMMA 4

450 DATA 0,0,0,C3,56,19,CD,74,61,C3,92,61,CD,AE,61,C3,50,61
: REM DATA voor de SPRONGADRESSEN


480 DATA 11,D0,60,1,3,0,7E,21,60,63,FE,0,20,B,ED,B0
490 DATA 11,D9,60,1,3,0,ED,B0,C9,9,9,FE,2,28,E,9,FE,3,28,9
495 DATA 9,FE,4,28,4,9,11,D9,60,ED,B0,C9
: REM DATA voor de VULROUTINE

500 DATA B0,D0,B0,B0,D0,B0,D0,B0,D0
: REM NOTENTABEL

```

PROGRAMMA 1 - PRINT VRIJ GEHEUGEN

60D9		C3 50 61	JP 6150	
6150		CD 56 19	CALL 1956	Haal toetscode in ASCII
		FE 07	CP 07	is het SHIFT 9
		C0	RET NZ	nee, terug v. gewone afhandeling
6156		3E 04	LD A,4	type 4 = Single Precision
		32 B6 63	LD (63B6)A	klaar zetten
		D5	PUSH DE	bewaar DE
615C		CD 3F 4E	CALL 4E3F	bereken FRE(0)
		CD 31 38	CALL 3831	zet resultaat in FAC
6162		CD 9E 3E	CALL 3E9E	ASCII, in printbuffer
6165		36 02	LD (HL),02	code voor groen
		11 00 50	LD DE,5000	1e schermregel
		01 06 00	LD BC,0006	6 stuks
6160D		ED B0	LD IR	Verplaats
		EB	EX DE,HL	
6170		36 07	LD (HL),07	afsluiten met code voor WIT
		D1	POP DE	herstel DE
6173		C9	RET	

Op het kleine toetsenbord zit een toets met een cirkeltje (=SHIFT ). Als u hier op drukt gebeurt er niets! U hoort alleen een "tuut". Dit is wel een erg mager resultaat voor zo'n fraaie toets. Programma 1 zorgt er nu voor dat het indrukken van deze toets beloond wordt door het verschijnen van de hoeveelheid vrij geheugen (in groene cijfers linksboven op uw scherm). Hetzelfde resultaat bereikt u met PRINT FRE(0), maar niet zo snel en ongekleurd. Als u gehecht bent aan de "tuut", zult u horen dat deze gelukkig niet verloren is gegaan. Omdat bovendien de toetsen SHIFT 2,4 en 6 dezelfde toetscode (=07) hebben, zijn deze plotseling ook "vrijgeheugen"-toetsen geworden.

PROGRAMMA TOELICHTING + MOGELIJKE WIJZIGINGEN:

- 6156 | A = 4 is de code als u FRE(0) wilt zien (= Single Precision type), als u A = 3 gebruikt krijgt u FRE("") dit is de vrije geheugenruimte in de stringruimte.

- 615C | CALL 4E3F is dezelfde routine die ook BASIC gebruikt voor de berekening van FRE(0). Het resultaat is een SP (Single Precision) getal, dat door CALL 3831 ineen integer wordt omgezet en opgeborgen in de FAC.
Op het scherm komt een getal dat 2 kleiner is, dan u verkrijgt via PRINTFRE(0): dat komt doordat CALL 4E3F uitgevoerd wordt nadat juist PUSHDE gedaan is; hierdoor is de Stack 2 plaatsen groter geworden en de vrije geheugenruimte dus 2 kleiner.

- 6162 | CALL 3E9E Zet dit (Binaire) getal om in ASCII en slaat het op in de Printbuffer op 651C.

- 616D | is een blokmove die het getal (1 kleurcode + 5 cijfers) op het scherm zet op het adres 5000, gegeven in DE; (voor een andere schermplaats: DE aanpassen).

PROGRAMMA 2: CURSORBESTURING TIJDENS BASIC-EXECUTIE

```

60D0 | CD 72 6      CALL 6174
      |-----|
6174 | CD 29 00    CALL 0029    Haal KB Status
      | C8      RET Z      Z = geen toets gebruikt
      | D8      RET C      C = STOP toets gebruikt
      | E5      PUSH HL    Save HL
617A | CD 56 19    CALL 1956    Haal toetscode + omzetten in ASCII
      | FE 10    CP 10      is het ←
      | 28 0C    JR Z 618D  ja, spring
      | FE 11    CP 11      is het ↑
      | 28 08    JR Z 618D
      | FE 12    CP 12      is het ↓
      | 28 04    JR Z 618D
      |-----|

```

	FE 13	CP 13	is het →
618D	20 03	JR NZ 6190	Verplaats cursor in gewenste richting
	CD 86 13	CALL 1386	
6191	E1	POP HL ←	Herstel HL
	C9	RET	

TOELICHTING:

6174	CALL 0029	deze routine	set Z vlag als geen toets gebruikt is.
		deze routine	set C vlag als STOP toets gebruikt is.
			zodat met RET Z of RET C de normale gang gevolgd kan worden.
617A	CALL 1956	deze routine	haalt de toetscode op en zet deze om in de ASCII-waarde in de Accu; vervolgens wordt gekeken of en zoja welke van de 4 pijltjestoetsen gebruikt is.
618D	CALL 1386	zoja, dan verplaatst deze CALL	de cursor in de gewenste richting. Daarna (of ook bij "zo nee") wordt het gewone programma voortgezet via RET

PROGRAMMA 2: CURSORBESTURING TIJDENS PROGRAMMA-UITVOER

Heeft u wel eens op de cursorpijltjestoetsen gedrukt? Ja, wat gebeurde er? O, niks, u hoorde alleen "tuut"! Dit komt omdat deze toetsen in de KEYBOARDROUTINE niet bekeken worden: ze werken alleen als de "EDIT" mode gebruikt wordt. Alleen dan is er een beperkte beweging mogelijk. Het volgende programma brengt hier verandering in. U kunt de cursor met behulp van de pijltjes vrij over het scherm heen en weer bewegen, dit is mogelijk terwijl een ander BASIC-programma in uitvoering is. U kunt de cursor dus gebruiken als een bewegende bal. Of als een achtervolgende raket in een BASIC-spelletje, met de volgende voordelen:

- A) er is een automatische controle op het bereiken van de rand (de cursor gaat niet van het scherm af)
- B) de beweging is snel, nl. de volgende tijdrovende BASIC "IF" statements worden overbodig:

```

IF PEEK (24588)=0 THEN GOTO ....
G = PEEK (24576)
IF G = 0 THEN X = X-1
IF G = 23 THEN X = X+1
IF G = 2 THEN Y = Y+1
IF G = 21 THEN Y = Y-1

```

NB.: U verplaatst inderdaad de cursor zelf, als in het BASIC-programma een PRINT opdracht voorkomt, dan wordt deze uitgevoerd op de plaats waar u de cursor inmiddels naar toe heeft gebracht (en omgekeerd: na een PRINT opdracht springt de cursor terug naar de kantlijn). De plaats van de cursor op dat moment kunt u op de gewone opslagplaats te weten komen:

{ 60B3 → cursorkolom
 { 60B4 → cursorrij

Om te stoppen moet u 2x de STOP-toets indrukken. U kunt een en ander testen met het (korte) programma 10 GOTO 10.

PROGRAMMA 3: SINGLE STEP BASIC

Heeft u wel eens de TRON opdracht gebruikt om de fout in een onwillig BASIC-programma te vinden? Was bij u ook het resultaat dat u onmiddellijk met een scherm vol regelnummers zat?

Waarschijnlijk had u opdat moment ook de volgende twee wensen:

- A) kan ik dat printtempo niet vertragen
- B) kunnen die regelnummertjes niet ergens in een klein hoekje verschijnen, zodat de informatie op het beeldscherm niet overschreven wordt.

En jawel, om met die bekende Nederlander te spreken: "Daar hebben we het volgende opgevonden": programma 3 zorgt er voor dat de aparte opdrachten van een BASIC-programma een voor een worden uitgevoerd. Het regelnummer van de opdracht wordt linksboven op het scherm in groen aangegeven. Het programma wacht dan tot u een willekeurige toets indrukt, pas daarna wordt de volgende opdracht uitgevoerd.

BASIC-opdrachten worden van elkaar gescheiden door de dubbele punt (:) of door een (niet zichtbare) nul aan het einde van de regel. Indien er bijvoorbeeld 3 opdrachten op een regel staan, moet u driemaal een willekeurige toets indrukken (bijvoorbeeld: 10 PRINT "A" : PRINT "B" : PRINT "C"). Linksboven staat dan driemaal regelnummer 10. Wilt u slechts een gedeelte testen, dan kunt u gewoon halverwege beginnen met behulp van bijvoorbeeld: GOTO 400 (in directe mode). Om te stoppen moet u tweemaal de STOP-toets indrukken. De allereerste keer verschijnt als regelnummer linksboven het getal 65535 (dit is het Hexgetal FFFF=-1), dat wil zeggen de waarde waarmee BASIC zelf het regelnummer initialiseert alvorens een programma geRUNned wordt.

PROGRAMMA 3: SINGLE STEP BASIC

60D0	C3 90 61	JP 6192	
6192	E5	PUSH HL	bewaar HL
6193	2A 5A 62	LD HL,(625A)	zet Regelnr. in HL
6196	CD 9E 3E	CALL 3E9E	omz. in ASCII+opslag in Printbuffer

6199	36 02	LD (HL),02	na 651C
619B	11 00 50	LD DE,5000	code van GROEN
619E	01 06 00	LD BC,0006	1e schermregel
61A1	ED B0	LD IR	6 stuks
61A3	EB	EX DE,HL	verplaats
61A4	36 07	LD (HL),07	afsluiten met WIT
61A6	E1	POP HL	herstel HL
61A7	CD 26 00	CALL 0026	wacht op willekeurige toets, test
61AA	D0	RET NC	KB status; set C if "STOP"
61AB	C3 03 25	JP 2503	NC doe volgende BASIC-opdracht
			"STOP"toets is gebruikt: beëindig
			het programma

TOELICHTING:

6193 625A bevat het regelnummer van de regel die momenteel in executie is.

6169- Dit regelnummer wordt "vertaald" in ASCII en op het
61A4 scherm gezet via een routine welke gelijk is aan die in Programma 1 (6162-6170)

61A7 CALL 0026; hier wacht de computer tot u toestemming geeft (=typ willekeurige toets) om door te gaan. Als het BASIC-programma onverhoopt "vast" zit (bijvoorbeeld 10 GOTO 10), dan zit hier ook de noodrem: nl. de check op "STOP"-toets. Het BASIC-programma wordt dan beëindigd via JP 2503.

NB.: Als er teveel (=foutieve) letters op het scherm verschijnen, dan was de Printbuffer achter 651C niet op nul gesteld: u kunt dit alsnog doen, nl. in direct mode, door vanaf 651C een tiental nullen in de buffer te POKEn.

PROGRAMMA 4: GELUIDSCOMMANDO "BOEM"

60D0	CD AC 61	CALL 61AE	
61AE	E5	PUSH HL	bewaar pointer
	7E	LD A,(HL)	
	FE 00	CP 0	test op einde regel
	20 04	JR NZ61B8	
	23	INC HL	passeer regelnummer
	23	INC HL	
	23	INC HL	
	23	INC HL	
61B8	23	INC HL	HL = pointer naar "B"
61B9	7E	LD A,(HL)	
61BA	FE 42	CP 42	test of het "B" is
	28 02	JR Z 61C0	ja: spring
	E1	POP HL	nee: terug naar BASIC
	C9	RET	
61C0	C1	POP BC	clear stack
	E5	PUSH HL	bewaar nieuw pointer

61C2	23	INC HL	sean verder; zoek vergelijking
	7E	LD A, (HL)	
	FE 3A	CP 3A	is het ":" (einde opdracht)
	28 0D	JR Z61D5	is het einde regel
	FE 00	CP 0	
	28 09	JR Z 61D5	is het "=" teken nog niet? zoek verder
	FE CA	CP CA	
	20 F2	JR NZ 61C2	ja! vergelijking gevonden
	E1	POP HL	stel pointer 1 plaats terug
	28	DEC HL	terug naar gewone BASIC
61D2	C3 13 28	JP 2813	
61D5	21 07 04	LD HL,0407	
	22 BC 60	LD (60BC),HL	laad vlag (60BC) en lengte (60BD)
	21 A2 63	LD HL,63A2	pointer voor noottabel
61DE	23	INC HL	volgende noot
	7E	LD A, (HL)	zet gereed in Accu
61E0	CD 86 13	CALL 1386	speel noot
61E3	3A BC 60	LD A, (60BC)	test op laatste noot
	B7	OR A	klaar?
	20 F5	JR NZ 61DE	nee doe volgende noot
61E9	E1	POP HL	ja, pak pointer (naar B)
	23	INC HL	passeer 4 letter van BOEM
	23	INC HL	
	23	INC HL	
	23	INC HL	
61EE	18 BE	JR 61AE	ga terug: er kan nog een BOEM volgen

PROGRAMMA 4: GELUIDSCOMMANDO "BOEM"

Het sprongadres 60D0 leent zich ook voor het invoegen van nieuwe BASIC-commando's. Programma 4 illustreert dit met het geluidseffect "BOEM". U kunt na het laden van programma 4 het volgende BASIC-programma intypen (bijvoorbeeld):

```
100 PRINT "verloren!" : BOEM
110 BOEM : BOEM : END
```

U kunt dan 3x het bijhorende geluidseffect horen. Dit programma is wat langer vanwege de benodigde controles onderweg; er zijn 3 delen te onderscheiden:

- A) 61AE-61B8 check op positie in de regel
- B) 61C2-61D2 check op vergelijking
- C) 61D5-61E7 de eigenlijke geluidsroutine

Op deze plek in de Interpreter (60D0), waar programma 4 begint, is HL een pointer. Deze wijst naar het adres achter de laatst uitgevoerde BASIC-opdracht: HL wijst naar een dubbelepunt (":"), of naar een nul ("0") die de regel afsluit. Deze pointer is essentieel: na afloop moet deze het juiste adres (= achter de volgende BASIC-opdracht) aanwijzen. Deel A test of de pointer een

":" aanwijst, of een eindnul. Is het een eindnul dan wordt HL vier plaatsen verhoogd om het volgende regelnummer te passeren. Vervolgens wordt gekeken of hierna een opdracht staat die met de letter "B" begint. Eerst moet dan nog bekeken worden of er soms een variabele bedoeld is, die ook met een "B" kan beginnen (bijvoorbeeld B= 2).

Deel B doet dit, door de volgende characters af te zoeken of er een gelijkteken ("=") voorkomt. Zo nee, dan is vastgesteld dat het volgende commando met een "B" begint. Omdat er verder geen (normale) BASIC-commando's zijn die met een "B" beginnen, moet dit het nieuwe commando "BOEM" zijn.

Vervolgens wordt dan in deel C het bijhorende geluid gemaakt. Dit gebeurt met dezelfde routine die ook gebruikt wordt in PRINT CHR\$(23) CHR\$(4) CHR\$(N) CHR\$(0) (nl. 61E0 CALL 1386). De lengte (= de duur) van de te gebruiken noten wordt in 60BD opgeslagen: alle noten zijn dus even lang, nl. 04 in 61D5 (een groter getal geeft langere noten). Het adres 60BC moet met 07 gevuld worden, dit is een vlag die door de routine gebruikt wordt om te bepalen of er nog meer noten volgen. Het adres 63A2 is het eerste adres van een tabel waar de noten zelf zijn opgeslagen. U kunt hier zelf eigen noten in opbergen (noten= getallen max. 255); de laatste noot moet 00 zijn. Deze tabel, die ook in het huishoudgedeelte van BASIC zit, heeft ruimte tot aan 63B4. Als deze notentabel (18 stuks) te klein is, kunt u hem nog verplaatsen naar een ander leeg gedeelte: nl. in 642D-648F (onder aanpassing van de pointer in 61DB).

Als de laatste noot gespeeld is (test 60BC op 0 in 61E6) dan wordt de oude pointer 4 plaatsen verhoogd, om de 4 letters van "BOEM" te passeren. Dit betekent dat er geen syntaxcontrole is op de laatste 3 letters van "BOEM". U kunt hier dus ook "BOOM" of "BAMM" van maken. Alle woorden van 4 letters die met "B" beginnen worden geaccepteerd. De eerste letter kan ook nog veranderd worden in B, J, K, Q, Y of Z (in 61BA), omdat er geen BASIC-opdrachten hiermee beginnen. (de andere letters zijn echter toch nog wel toegestaan, als het BASIC-opdrachten betreft die in een vergelijking voorkomen; uitbreiding echter tot 4 letters syntaxcontrole is wel beter)

Leesfouten ?

Door het toepassen van een dikkere cassetteband kunnen bij sommige recorders leesfouten optreden, omdat de recorder te gevoelig is. Dit is met een kleine aanpassing te verhelpen, waarvoor hier de beschrijving volgt.

Maak de computer open door de zeven schroeven aan de onderkant los te draaien en verwijder de kap. Draai nu de vier schroeven los waarmee de recorder op de bodem vast zit en klap de recorder naar voren met de bovenkant boven en de achterkant naar u toe. In de linkerbovenhoek staat R62. Soldeer daar een weerstand van 100 K overheen en zet dan de zaak weer in elkaar.

Aan de hand van mijn artikel in nieuwsbrief 8 over de 42K geheugenuitbreiding hebben een aantal mensen dit projekt uitgevoerd. Daarbij zijn enkele typfouten, onduidelijkheden en problemen aan het licht gekomen.

Eerst maar de typfouten:

- op bladzijde 6 staat bij het 6e punt in de bouwhandleiding: "de oude -5V leiding", dit moet zijn (conform de tekening) "de oude +12V leiding".

- de luidsprekerversterker op bladzijde 9. Aan de connector staat een weerstand van 10 ohm getekend, dit moet 100 ohm, 1 watt zijn.

Dan de onduidelijkheden:

- de adrescodering in de P2000 met behulp van een 8x32 PROM verdeelt het totale (64K) geheugen van de P2000 in blokken van 2K. Dat betekent dat ieder (van de 32) adressen in de PROM naar een stuk geheugen van 2K wijst. In de tabel op bladzijde 5 heb ik dit aangegeven door te zeggen dat bijvoorbeeld tussen 24 en 40K 7E in de PROM geprogrammeerd moet worden. Dit betekent dus dat in adres 12 (24-26K) tot en met adres 19 (38-39K) van het programma 7E moet staan. Het originele PROM bevat dus:

```
74 74 5C 5C 5C 5C 3C 3C 3C 3C 79 7D 7E 7E 7E 7E
```

```
7E 7E 7E 7E FD FD FD FD FD FD FD FD FD FD
```

en het nieuwe PROM bevat dus:

```
74 74 5C 5C 5C 5C 3C 3C 3C 3C 79 7D 7E 7E 7E 7E
```

```
7E 7E 7E 7E 7E 7E 7E 7E 7E 7E 7E 7E 7E 7E
```

Als laatste, maar helaas niet minste, de problemen:

- het is gebleken dat niet alle fabrikanten dezelfde specificaties aanhouden voor de geheugenchip 4164. In het kort: de 4164 is een dynamisch RAM, dat regelmatig "gerefreshed" moet worden, anders verliest het zijn data. Nu wordt dit gedaan door het geheugen met vaste regelmaat een serie adressen aan te bieden, zodat het daar intern zijn geheugen mee kan blijven vasthouden. En daar zit nu precies het probleem.

De Z80 chip biedt voor deze "refresh cycle" een 7-coloms adrescycle (128 verschillende adressen) aan, voldoende voor de "normale" 4164. Nu blijkt echter dat diverse fabrikanten de 4164 uitvoeren met een 8-coloms adrescycle (256 adressen), waardoor deze ongeschikt is voor de Z80 cpu. Diezelfde fabrikanten hebben dan de juiste chip onder de naam 4864 (Mostek, Siemens). Als conclusie kunnen we dus stellen dat u, wanneer u de uitbreiding wilt maken, de datasheets van de betreffende fabrikant eerst moet doorlezen of het een 7-coloms "refresh cycle" betreft, voor u de chip aanschaft.

- de extra 2K RAM of adres 5800-6000. Hiervan heb ik de laatste keer gezegd dat dit geheugenbereik ook gevuld kan worden met RAM. Dit is op zich waar; je kan er heen lezen en schrijven, alleen is er een probleem: als het scherm gecleard wordt, wordt ook dit deel van het geheugen gereset. Het is dus niet echt handig om hier machinetaalroutines in RAM te plaatsen.

De kommunikatiemodule (KM) bestaat uit een kassette voor het tweede slot en een programma, geleverd in een Eprom, of naar wens op tape. De KM is in staat twee seriele input/output-poorten (half duplex) en een parallelle outputpoort simultaan te besturen. Daarnaast wordt een gedeelte van het P2000-geheugen gebruikt als printerbuffer zodat men verder kan werken met de computer terwijl geprint wordt. De parallelle poort heeft de centronics standaard of is direct als interface te gebruiken voor de Brother CE50/60, de Praxis 30/35/40 en de IBM bolkopschrijfmachine.

De data die over de inputpoorten binnenkomt wordt rechtstreeks doorgegeven aan BASIC, alsof het toetsenbord gebruikt wordt. Ook de originele printeruitgang is nog te gebruiken, en combinaties van printer, modem (Viditel) en andere randapparatuur zijn mogelijk. De EPROM (met het programma) past in de vrije IC-voet in de P2000. Deze extra software gaat niet ten koste van de bestaande hoeveelheid vrije geheugenruimte.

De standaard P2000 heeft weinig mogelijkheden om met de buitenwereld te communiceren, namelijk slechts een seriele RS232 poort, en dan nog alleen als uitgang.

Er zijn een aantal gronden om meer te wensen, bijvoorbeeld een parallelle centronics uitgang of een extra seriele tweerichtings RS232 poort:

met een paralleelpoort kunnen goedkope printers gebruikt worden die bovendien sneller aangestuurd kunnen worden.

met een tweeweg seriele poort kan een modem worden aangesloten of direct contact gemaakt worden met elk ander apparaat dat deze standaard voert.

extra poorten maken het mogelijk om meerdere apparaten tegelijkertijd aan de P2000 te koppelen, bijvoorbeeld een Viditelmodem en een printer.

Dit alles en meer is verwezenlijkt in de kommunikatiemodule. Gekozen is voor een oplossing waarbij de hardware zo eenvoudig en veelzijdig mogelijk is, en de rest (het leeuwendeel) van het werk gedaan wordt door de software. Met deze combinatie hebben we het volgende kunnen realiseren:

- 2 RS232 poorten ingang/uitgang,
- en een centronics parallelle uitgang,
- een interface voor de Brother CE50/60 (voor de CE50 dient een interface connector unit van Brother te worden aangeschaft (f 70,--); in de CE60 is dit ingebouwd. Verder heeft u een interfacekabel nodig. U kunt die samen met de KM bestellen (f 75,--; losse stekker f 41,--)),

- een interface voor de Praxis 30/35/40 (interfacekabel zelf te solderen aan de hand van tekening en beschrijving die bij de KM geleverd worden),
- een interface voor de IBM bolkopschrijfmachine (deze moet over een ingangspoort beschikken),
- een interne printerbuffer, in grootte instelbaar,
- een volledige samenwerking tussen BASIC en de KM-software.

Zodra de KM-software geactiveerd is kunnen uw eigen programma's normaal lopen, met dat verschil dat de te printen tekst nu niet meer aangeboden wordt aan de originele printeruitgang, maar tijdelijk wordt opgeslagen in een buffer. Hiermee is bereikt dat uw BASIC-programma's verder kunnen gaan terwijl ondertussen tegelijkertijd de tekst wordt uitgeprint.

De ingangen werken precies zo eenvoudig. Wordt een bepaalde inputpoort geactiveerd dan wordt de data die over die poort wordt aangeboden aan BASIC doorgegeven alsof het toetsenbord gebruikt wordt. Met andere woorden: als bijvoorbeeld twee P2000's aan elkaar zijn gekoppeld (uiteraard beide voorzien van een KM), dan zal een "l1ist" op de ene als resultaat hebben dat het programma overgeseind wordt naar de andere P2000, die dit dan automatisch als nieuw programma laadt. Een andere mogelijkheid is om met behulp van een BASIC-regel (met het input-commando) de P2000 te veranderen in een terminal.

De in- en uitgangen werken los van elkaar, en kunnen tegelijkertijd gebruikt worden (slechts 1 uitgang per keer). Indien gewenst kunnen eigen printroutines gebruikt worden voor speciale schrijfmachines/printers die niet tot de standaardmogelijkheden behoren.

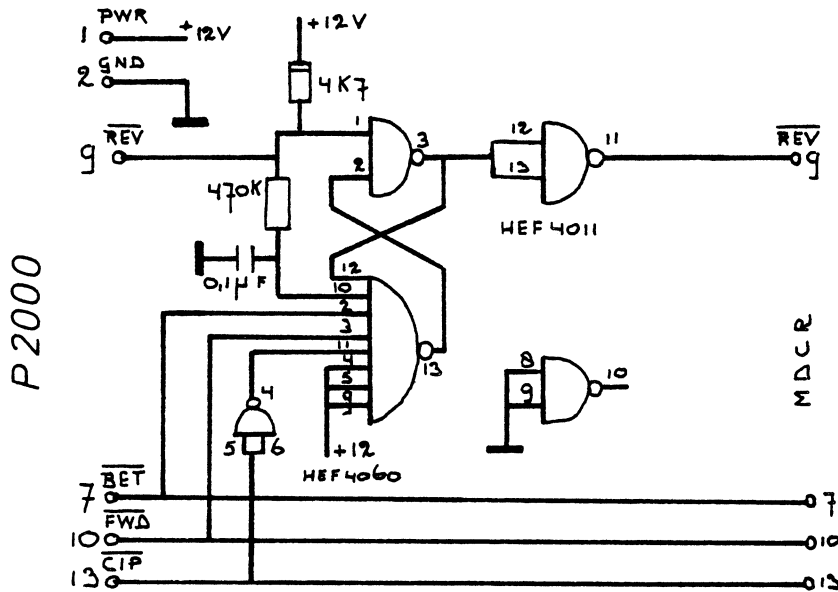
Arthur Nederlof
 Wilhelm Leuschnerstrasse 54
 6083 Biebesheim
 BRD
 tel. 09-4962587619



Dopmoeren op de printerconnector

De connector voor de printer is voorzien van z.g. dopmoeren, waarop de connector kan worden vastgeschroefd. Vele V24 stekers zijn echter niet voorzien van deze bevestiging; in plaats daarvan is de stekker dichtgeschroefd met schroefjes, waarvan de koppen tegen de dopmoeren aankomen. Resultaat is dat de verbinding niet ver genoeg in elkaar gestoken kan worden en in het gebruik los gaat. In dit geval is het aan te bevelen de dopmoeren los te draaien en te vervangen door 2 korte boutjes (M3). Bij oudere uitvoeringen van de P2000 moeten tevens moertjes worden aangebracht, bij latere uitvoeringen van de P2000 behoeft de computer niet eens geopend te worden voor deze eenvoudige ingreep.

Het gebruik van cassette voor de opslag van programma's en gegevens duurt altijd even omdat alle data achter elkaar staan opgeslagen en het zoeken daarin enige tijd kost. Daarbij wordt eerst de band teruggespoeld tot het begin en vervolgens wordt de gevraagde informatie opgezocht. Zoeken en spoelen gaat met dezelfde snelheid, dus wanneer de cassette automatisch steeds zou worden teruggespoeld naar het begin kan het zoeken gemiddeld tot de helft van de tijd worden teruggebracht. Onderstaande schakeling kan hiervoor gebruikt worden:



Deze schakeling kan gebouwd worden op een printje, dat in de minicassetterecorder past. Dus de stekker, die origineel in de cassetterecorder steekt, eruit halen, printje erin en de stekker weer in het printen steken.

BESTURING

De terugspoelbesturing moet door het programma geschieden. Het eenvoudigste is om als een van de eerste instructies van het programma op te nemen:

OUT 16,68

en bovendien iedere keer nadat er iets met de cassette is gedaan. De tape spoelt dan naar het begin terug, terwijl de uitvoering van het programma gewoon doorgaat. De P2000 kan echter op elk moment de besturing van de recorder weer overnemen. In machinetaal is een terugspoelinstructie te geven door bit 3 van outputpoort &H10 op 1 te zetten en direct daarna weer op 0.

WERKING

In de schakeling zit een flip-flop, die door een korte rewindpuls op 1 gezet wordt. Deze 1 houdt de recorder in terugspoelen. Met OUT 16,68 wordt een rewindinstructie gegeven en deze wordt door de toetsenbord interrupt weer uitgezet. Daardoor ontstaat de korte puls. Bij een puls langer dan 1/30 seconde valt de flip-flop weer op 0 als de puls is afgelopen. Daardoor wordt de normale

besturing van de P2000 niet verstoord. De flip-flop valt ook weer op 0 als: de band aan het begin is, er geen cassette (meer) is of als de cassette vooruit moet spoelen.

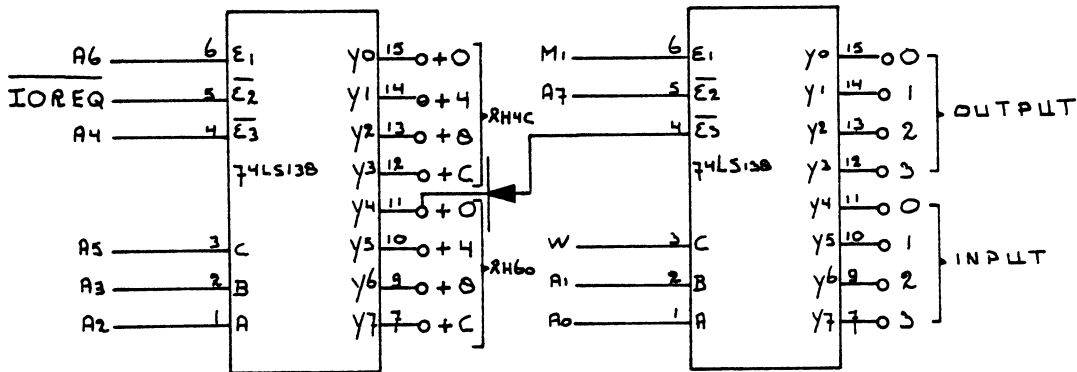
LET OP!! Omdat OUT 16,68 direkt weer door interrupt terug gezet moet worden, werkt dit alleen als de interrupt aan staat!



Uitcoderen van I/O-adres

Bij het gebruik van een print in slot 2 ten behoeve van het aansluiten van randapparatuur, komt men steeds weer voor het ontwerp van een adres-decoder te staan. Aan de ene kant is het aantrekkelijk om alle bits uit te decoderen omdat dan een verkeerde interface-does gewoon niet werkt; aan de andere kant streeft men naar een minimaal aantal IC's.

Onderstaande schakeling combineert beide wensen. Met slechts twee IC's heeft men de beschikking over 4 output-poorten en 4 input-poorten met 4 aaneensluitende poortnummers, die in stappen te leggen zijn in de bereiken; &H40 - &H4F en &H60 - &H6F.



Door ingang E3 van de tweede 74LS138 aan te sluiten op een van de uitgangen Y0 - Y7 van de eerste 74LS138 kan men in stappen van 4 kiezen welke reeks poortnummers men wil gebruiken. De uitgangen Y0-Y3 van de tweede 74LS138 geven een negatieve puls bij een OUT-instructie. Voor output kan men de puls gebruiken om de data in te klokken in bijvoorbeeld een 74LS273. Voor input kan men een 74LS244 buffer aansturen om de externe data op de databus te zetten. Het gebruikte signaal M1 is noodzakelijk om te voorkomen dat de interface ook aangesproken wordt tijdens een interrupt in de modes 0 en 2. BELANGRIJK!! Belast de databus en de adresbus absoluut niet zwaarder dan met een LS-TTL ingang. Indien meer dan 1 output latch in gebruik is, betekent dat het inbouwen van een buffer IC.

Dit artikel is eigenlijk geschreven voor de HCC-Nieuwsbrief. Enkele maanden geleden heeft het bestuur van de P2000gg met het bestuur van de HCC afspraken gemaakt over publikaties van de P2000gg in de HCCNieuwsbrief. Omdat er eigenlijk nooit wat over de P2000 in die Nieuwsbrief staat, hebben we afgesproken dat wij in elk nummer twee pagina's zouden vullen met ongecensureerde wetenswaardigheden over de P2000T. Maar er is weer eens iets mis gegaan bij de HCC. Of het HCC-bestuur had niet de bevoegdheid deze afspraak te maken, of het heeft vergeten de zaak af te stemmen met de hoofdredacteur van het blad. Die voelde zich in elk geval niet gehouden aan de afspraak en deelde lakoniek mee dat hij zelf wel uitmaakt wat er wel en wat er niet in zijn HCC-Nieuwsbrief komt. Om kort te gaan: in het oktobernummer staat weer bijna niets over de P2000, op een artikel van Peter van Asseldonk over het Familiegeheugen na.

Omdat het artikel over wortel trekken met dubbele nauwkeurigheid toch wel interessant is, publiceren we het maar alvast in onze eigen Nieuwsbrief. Het idee is afkomstig van Albert Nijenhuis, en is door mij een beetje uitgebreid en aangevuld.

16 in plaats van 6 cijfers

Net zoals de andere wiskundige functies die in de P2000 aanwezig zijn, wordt de wortel uit een getal, $SQR(X)$, in enkele precisie uitgerekend. U krijgt in dit geval een nauwkeurigheid van 6 cijfers. Het is echter op eenvoudige wijze mogelijk de wortel in dubbele precisie, met een nauwkeurigheid van 16 cijfers, uit te rekenen, namelijk door de variabele met dubbele precisie te kiezen. Als u intoetst: $W\#=SQR(2):PRINT W\#$, dan krijgt u 1.414213538169861. U denkt dan de wortel uit 2 met een nauwkeurigheid van 16 cijfers te hebben berekend. Maar dit is een schijnnaauwkeurigheid (toetst u maar eens in: $W\#=SQR(2):PRINT W\#*W\#$). Een werkelijk nauwkeurige berekening gaat als volgt:

$$WO\#=(SQR(X\#)+X\#/SQR(X\#))/2\# \quad (1)$$

Vult u voor $X\#$ de waarde $2\#$ in, dan vindt u voor $WO\#$ de waarde 1.414213562373095. Dat zijn ook 16 cijfers, maar de staart is anders dan we eerst vonden. Als u nu (1) aanvult met $PRINT W\#*W\#$, dan vindt u met de P2000: 2.000000000000001. De gevonden wortel is nu dus wel nauwkeurig.

Om uit te leggen hoe dit werkt, is een beetje wiskunde nodig. Laten we even aannemen dat de werkelijke wortel die we zoeken, $WO\#$, gelijk is aan $SQR(X\#)$. In formule (1) staat dan:

$$WO\#=(WO\#+X\#/WO\#)/2 \quad (2)$$

Met andere woorden: de onbekende $WO\#$ die we zoeken staat niet alleen voor, maar ook tweemaal achter het is-gelijk-teken, en wel eenmaal boven en eenmaal onder de breukstreep (rekent u formule (2) uit dan vindt u natuurlijk $WO\#=WO\#$; een waarheid als een koe). Dit betekent dat de formule (2) alleen opgaat als $WO\#$ de juiste wortel uit het getal $X\#$ is.

De grap van formule (1) is nu dat we voor $SQR(X\#)$ een geschatte waarde kunnen invullen. Is die geschatte waarde te groot, dan wordt $SQR(X\#)$ te groot en $X\#/SQR(X\#)$ te klein. Dat compenseert elkaar min of meer. Maar belangrijker is dat we een waarde voor $WO\#$ vinden die een betere benadering van de werkelijke wortel is dan we de eerste keer hebben aangenomen. Anders gezegd: $WO\#$ is nauwkeuriger dan de wortel $SQR(X\#)$ die we door de computer hebben laten uitrekenen.

Met formule (1) kunt u, met behulp van een zakrekenmachine (oneerbiedig zakjapanner of hersenprothese geheten), maar natuurlijk ook met uw computer, in een aantal stappen elke willekeurige vierkantswortel berekenen. Laten we als voorbeeld de wortel uit 10 nemen. U kunt even niet erg goed schatten, en u denkt dat de wortel van 10 in de buurt van 2 ligt. U toetst dan in (in de directe stand):

WO#=(2+10/2)/2:PRINT WO#.

U vindt dan WO#=3.5.

3.5 is dus een betere benadering van de vierkantswortel uit 10.

Met dit getal gaat u verder rekenen:

WO#=(3.5+10/3.5)/2:PRINT WO#. Dan vindt u: WO#=3.178571462631226, en dat is weer een betere benadering van de wortel die we zoeken.

De formule werkt dus convergerend. Na een aantal stappen vindt u voor WO# de waarde die u de laatste keer in de formule hebt ingevoerd. Als de gevonden waarde voor WO# niet meer verandert, hebt u de juiste wortel gevonden.

Wat we met formule (1) hebben gedaan is deze procedure eenmaal doorlopen, en daarbij gebruik maken van het feit dat $\text{SQR}(X\#)$ een redelijk nauwkeurige schatting is van de vierkantswortel uit $X\#$.

Hogeremachtswortels

Wat we zojuist hebben geschreven voor vierkantswortels, geldt in zijn algemeenheid voor alle wortels, ook voor hogeremachtswortels.

Noemen we de N-e machts wortel uit een willekeurig getal WO (we laten voor de overzichtelijkheid het hekje # even weg), dan kunnen we schrijven:

$$WO = \frac{N - 1}{N} \left(WO + \frac{X}{(N - 1) * WO^{N-1}} \right)$$

In deze formule kunnen we dus een geschatte waarde voor de N-e machts wortel invullen, en we vinden dan een nauwkeuriger wortel, die we opnieuw kunnen gebruiken om een nog betere benadering te vinden.

Met het hier afgedrukte programma kunnen we met de P2000 (en met elke andere computer die met dubbele precisie kan rekenen) elke willekeurige wortel uit elk willekeurig getal berekenen, met een nauwkeurigheid van 16 cijfers.

In regel 10 definiëren we alle variabelen die met een letter N tot en met Y beginnen als dubbeleprecisie-variabelen. Regel 20 wist het scherm. In de regels 80, 90 en 100 rekenen we voor de overzichtelijkheid wat delen van de formule uit. In regel 110 bepalen we de geschatte waarde van de wortel en die wordt op het scherm weergegeven in regel 120 (VW = Voorlopige Wortel). Dat schatten gebeurt door de N-e machts wortel uit X te schrijven als X tot de macht T=1/N.

Regel 140 rekt de nieuwe voorlopige wortel V tot de macht (N-1) uit en regel 150 bepaalt de noemer van de grote breuk door de in 140 gevonden waarde voor V met S=(N-1) te vermenigvuldigen. Regel 160 is de eigenlijke formule, opgedeeld in de stukken die we eerder hebben berekend.

In regel 170 kijken we of de absolute waarde van het verschil tussen de gevonden wortel W de waarde VW die we eerder hebben gevonden, kleiner is dan 10^{-16} . Is die waarde groter, dan wordt de berekening opnieuw uitgevoerd met de nieuwe, nauwkeuriger wortel. Dat gebeurt net zo lang totdat de wortel niet meer verandert. Is de nieuwe wortel gelijk aan de oude, dan kan geen nauwkeuriger resultaat worden bereikt en wordt in regel 200 de uitkomst van de berekening op het scherm gebracht.

```

10 DEFDBL N-Y
20 PRINT CHR$(12)
30 INPUT"Welk getal";X
40 IF X<0 THEN 20
50 INPUT"Welke wortel";NN
60 N=INT(NN):IF N<2 THEN 10
70 PRINT STRING$(39,"=")
80 S=N-1
90 Q=S/N
100 T=1/N
110 VW=X^T
120 PRINT"Voorl. wortel VW=";VW
130 V=1
140 FOR I=1 TO N-1:V=V*VW:NEXT
150 R=S*V
160 W=((X/R)+VW)*Q
170 IF ABS(W-VW)<1D-16 THEN 180 ELSE VW=W:GOTO 120
180 Y=1:FOR I=1 TO N:Y=Y*W:NEXT
190 PRINT STRING$(39,"=")
200 PRINT" De"N;CHR$(16) "e machts wortel uit"X:PRINT"
    is gelijk aan"W
210 PRINT:PRINT W"tot de macht "N:PRINT "="Y
220 PRINT STRING$(39,"=")
230 END

```

In regel 180 verheffen we de laatst gevonden wortel tot de macht N. Dat moet dan weer het oorspronkelijke getal X opleveren. In regel 210 wordt de uitkomst van die berekening op het scherm gebracht, niet alleen tot lering en vermaak maar ook om te kunnen controleren hoe nauwkeurig de berekening van de wortel is geweest.

Met dit programma kunt u op de P2000 zeer snel de 23ste machts wortel uit 123456787654321 berekenen, als u daar goesting in hebt (dat is 4.09896834126794 om precies te zijn). Aardig is ook de vierkantswortel uit 123456790.1234568. Merk op dat dit getal bestaat uit de cijfers 1 tot en met 0, waarin 8 ontbreekt, en nogmaals 1 tot en met 8, waarin 7 ontbreekt.

Het aardige van het programma is dat de rekennauwkeurigheid van de computer geen rol lijkt te spelen. De computer rekt door totdat de wortel niet meer verandert, en geeft dan het eindresultaat. Maar ik heb het niet kunnen proberen omdat ik nu eenmaal niet beschik over thuiscomputers die minder nauwkeurig rekenen dan de P2000.

Meer informatie over rekenen in dubbele precisie kunt u vinden in het blad "Creative Computing" van december 1983 en januari 1984. Hierin staan twee artikelen van Albert Nijenhuis over SIN, COS, EXP, LOG, ATN en SQR in hoge precisie.

Rob Geutskens

```

Welk getal? 111
Welke wortel? 9
=====
Voorl. wortel VW= 1.68755578994751
Voorl. wortel VW= 1.687555671061703
Voorl. wortel VW= 1.687555671061669
=====
De 9 e machts wortel uit 111
is gelijk aan 1.687555671061669

1.687555671061669 tot de macht 9
= 111
=====

```

Om de gegevens, die u met het Supersnelzoekboek hebt ingevoerd, te kunnen afdrukken, moeten enkele regels aan het programma worden toegevoegd:

```
8620 PRINT CHR$(12)CHR$(13);:GOSUB 9000
8625 PRINT FNP$(24,1) M$"[ opnieuw 00 PRINT E EDIT
verder ]";
8630 I=INP("):IF I=69 OR I=101 THEN 9800 ELSE IF I=16
THEN 8450 ELSE IF I=19 THEN 8520 ELSE IF I=5 THEN PRINT
CHR$(15)CHR$(5);:GOTO 8625 ELSE 8630
65432 IF ERR=80 THEN PRINT CHR$(15) FNP$(24,1) M$ "zet de
printer aan";:RESUME ELSE POKE &H60AD,0:POKE &H60B0,39:
CLEAR 50:END
```

Gebuikt u het gewone snelzoekboek, dan moeten de volgende regels worden toegevoegd:

```
1180 PRINT CHR$(12)CHR$(13);:GOSUB 2100
1190 PRINT FNP$(24,1) M$"[ opnieuw 00 PRINT E EDIT
verder ]";:POKE 22321,91:POKE 22358,93
1205 IF TC=18 THEN PRINT CHR$(15);:IF INP(32) AND 2 THEN
PRINT FNP$(24,1) M$ "zet de printer aan";:GOTO 1205 ELSE
PRINT CHR$(5);:GOTO 1190
```

De rechte haken [en] in de regels 8625 en 1190 geven de toetsen aan die u moet indrukken. Op het scherm en in het programma komen dan pijltjes naar links ← en naar rechts → te staan.

Ton Hilgersom

Voorkeur-kleurencombinaties

Het komt soms voor dat in programma's kleuren gebruikt worden die slecht leesbaar zijn, vooral bij gebruik van een monochrome (eenkleurige) monitor. Daarom hier een tabel met bruikbare combinaties; X = bruikbaar, + verdient de voorkeur omdat die ook op monochrome schermen goed leesbaar zijn.

	Voorgrondkleur nr.:						
achtergrond * kleur	129	130	131	132	133	134	135
129	*	x	+	+		x	+
130	*	+		+	+		
131	*	+		+	+		
132	*	x	x	+		x	+
133	*		x	+	+	x	+
134	*	+		+	+		
135	*	+		+	+		

Wat krijgen we nou ? Hebt u net een P2000 gekocht en moet u in uw eigen Nieuwsbrief lezen hoe u hem weer moet verkopen ?

Dat "verkopen" moet u niet te letterlijk opvatten. Maar u hebt natuurlijk ook de ervaring dat u aan uw omgeving moet verkopen waarom u in vredesnaam een P2000 hebt gekocht. Aan uw vrouw en kinderen, aan uw moeder, aan uw buurman en aan de meewarig kijkende bezitters van Apples, Commodores en Spectra, om maar eens een paar dwarsstraten te noemen. Hoe verkoopt u die miskleun, die onvergeeflijke blunder, in een vlaag van totale verstandsverbijstering begaan ? Waarom hebt u niet gewoon zo'n thuiscomputer gekocht als Maurice de Hond heeft aanbevolen ? Zo'n computer waarmee Vroom & Dreesmann het Nederlandse volk het computerparadijs wil binnenlokken ?

Omdat Maurice de Hond de afgelopen week op een persconferentie nogal wat uitspraken heeft gedaan die P2000-bezitters in het verkeerde keelgat zijn geschoten, zullen we een rollenspel opvoeren waarbij u moet proberen uw P2000 te verkopen aan De Hond. Om het wat moeilijker te maken leggen we hem wat uitspraken in de mond die hij niet gedaan heeft, maar die hij gedaan zou kunnen hebben als je kijkt naar wat hij wèl gezegd heeft. Kortom: waar vindt u de stokken om De Hond te slaan ?

dH voor de P2000 is gewoon te weinig software (in De Volkskrant)

U Flauwekul, Maurice. Er zijn voor de P2000 honderden programma's van uitstekende kwaliteit.

dH 80% van de software is vuilnis, 10% is redelijk, de rest is goed (in De Volkskrant)

U Kolder. U hebt blijkbaar nooit naar het programma-aanbod voor de P2000 gekeken. Je krijgt maar niet zo een programma in Viditel of op A-casette.

dH De P2000 kan internationaal niet concurreren (in Het Parool)

U Dat slaat nergens op. De P2000 is een Nederlandse computer. Of hij in Amerika wordt gebruikt zal me worst wezen. Trouwens, op die persconferentie hebt u geklaagd dat er zo weinig Nederlandstalige software is. Nou is er een Nederlandse computer die de gebruiker zo veel mogelijk in het Nederlands toespreekt, met veel programma's, die bijna zonder uitzondering door Nederlandse programmeurs zijn gemaakt, waar geen woord Engels in voorkomt als het ook in het Nederlands kon, en nou beveelt u V&D een Amerikaanse en een Engelse computer aan.

Ik denk dat er boos opzet in het spel is. U hebt doelbewust twee computers gekozen waarvoor bijna geen goede Nederlandstalige programma's zijn, alleen om daarna te kunnen jammeren dat er geen Nederlandstalige programma's voor zijn, en om vervolgens heel menslievend te roepen dat V&D er wel voor zal zorgen dat die programma's er komen. Voor vele tientallen guldens per stuk. V&D wil niet alleen aan de computers, maar ook aan de programma's verdienen. Zeg maar gewoon dat de P2000 voor V&D niet interessant is omdat iemand die een P2000 koopt nooit meer terugkomt om programma's te kopen. Aan P2000-programma's valt niets te verdienen omdat ze zo goedkoop zijn.

dH Die programma's zijn zo goedkoop, dat kan nooit wat zijn

U Apekool. Philips heeft heel bewust, en heel verstandig, de gebruikersprogramma's vrijgegeven. Voor een paar gulden haal je al een goed programma uit Viditel en voor f 3,50 meer dan een lege cassette kost koop je bij de gebruikersgroep een cassette boordevol programma's. Een paar kwartjes per programma. Kopiëren bij een vriendje is nauwelijks

goedkoper dan een programma uit Viditel halen of bij de gebruikersgroep kopen. Bij ons hoeft de politie geen invallen te doen om krakers en kopieerders te betrappen. V&D kan zijn lol nog op als zijn dure programma's eenmaal gekraakt zijn en er voor elk legaal programma vijftig illegale kopieën in omloop zijn.

dH P2000-programma's zijn zeker niet te beveiligen ?

U Wel degelijk. Er zijn professionele programma's die je niet zo maar kunt kopiëren. Dan moet je van goeden huize komen.

dH De P2000 is een oude, lompe machine

U Dat'ie oud is bewijst alleen maar dat'ie goed is. Anders was er allang een opvolger geweest. En lomp ? Philips heeft alles ingebouwd wat je altijd nodig hebt. Wat je niet altijd nodig hebt kun je eraan hangen of erin steken. Welke ontwerper haalt het in zijn hoofd om de voeding en de recorder buiten boord te hangen; die heb je altijd nodig. En de BASIC-interpret, die je niet altijd nodig hebt, hebben ze ingebouwd. Sluit op de P2000 een monitor of een TV-toestel aan en je kunt aan de slag, zonder eerst de hele tafel vol te zetten met rommel en spaghetti.

dH De P2000 heeft geen fijne graphics

U Fijne graphics zijn leuk voor spelletjes. Maar de P2000 is niet bedoeld als spelcomputer, maar als professionele machine. Je kunt er wel spelletjes mee spelen, heel leuke zelfs, maar als je alleen spelletjes wil spelen moet je één van uw spelletjescomputers kopen. Als je meer wil, kun je met de P2000 jaren vooruit.

U hebt waarschijnlijk nooit goed naar de grafische mogelijkheden van de P2000 gekeken. Die zijn ideaal voor Viditel. Programmaatje erin, modempje eraan en je kunt zo Viditel in. Om een goed en goedkoop programma op te halen bij voorbeeld. Om om een vraag op een prikbord te schrijven als je met een probleem zit. De P2000 heeft verreweg het grootste Telesoftware-bestand. Waarom ? Omdat hij gemaakt is voor Viewdata. Probeer dat eens met een andere computer.

Trouwens, je staat te kijken wat met die beperkte grafische mogelijkheden van de P2000 aan fraai ogende plaatjes gemaakt kunnen worden.

En dan nog wat: de P2000 gebruikt het hele scherm, en hij schrijft prettig leesbare letters wit op zwart, met een behoorlijke tussenruimte om de leesbaarheid te verbeteren. De P2000 heeft geen blauwe rand om het scherm, die andere computers nodig hebben om te verdoezelen dat zij maar de helft van het scherm werkelijk gebruiken.

dH De P2000 is dus geen vlees en geen vis. Het is geen spelcomputer, maar het is ook geen echte professionele machine

U Dat is maar hoe je het bekijkt. Er zijn duizenden mensen die meer willen dan spelletjes spelen die door anderen bedacht en geprogrammeerd zijn, maar die ook geen computer nodig hebben waarmee je de boekhouding van een multinational kunt voeren. De P2000 vult het gat tussen spelcomputers en de machines die je alleen maar voor bloedserieuze dingen kunt gebruiken. Je kunt er een heel behoorlijke administratie op bijhouden. In een handomdraai steek je er een module in waarmee je de P2000 omdoert tot een professionele tekstverwerker, met een professioneel toetsenbord. En even zo makkelijk kun je er het Familiegeheugen in steken, om behoorlijk omvangrijke bestanden bij te houden.

En wat dacht u van de (alweer dat rotwoord) professionele digitale cassetterecorder die de P2000 aan boord heeft; tienmaal zo snel als een audiocassetterecorder en honderdmaal zo gemakkelijk omdat hij helemaal door de computer wordt bestuurd. Met aanwijzingen in het Nederlands. Als je daar eenmaal mee gewerkt hebt, zul je nooit meer kunnen wennen aan het gemodder met een trage, losse audiocassetterecorder.

En dan nog wat. De BASIC van de P2000 rekt beter dan die van bijna alle andere computers, inclusief grote systemen van grote leveranciers. Probeer maar eens een eenvoudige lakmoesproef: het 27 keer achtereenvolgend kwadrateren van het getal 1.000 000 1. Daar moet, in 10 cijfers nauwkeurig, 674 530.4707 uit komen. Dat gaat met het volgende BASIC-programma:

```
10 A#=1.0000001
20 FOR I=1 to 27:A#=A#*A#:NEXT
30 PRINT A#
```

Het resultaat:

Merk en type	Resultaat	Absoluut verschil
IBM PC BASIC	8 850 273.	-.-
Apple-II BASIC	22 723.970 9	-.-
Commodore 64	22 723.970 9	-.-
MSX-BASIC	674 529.413 0	1.0577
CDC Cyber 10-digits FORTRAN	674 530.576 5	0.11
CDC Cyber double prec. FORTRAN	674 530.536 3	0.066
IBM Retain	674 530.452 7	0.018
P2000 24K-BASIC	674 530.475 6	0.0049
P2000 BASIC-NL	674 530.473 1	0.0024
TRS 80	674 530.470 6	0.0001

U ziet, alleen de TRS-80 is nog nauwkeuriger. Ook zo'n ouwe getrouwe.

dH (met een begin van aarzeling) Maar de P2000 is zo d ur

U Maar dan ben je ook in  en keer klaar. Je hoeft er niks meer bij te kopen om vrolijk te kunnen computeren. Alles wat je nodig hebt zit erin. Je krijgt voor honderden guldens programmatuur meegeleverd, als je met de prijzen rekent die V&D hanteert. Ik heb trouwens meer dan 50 programma's. Dat is alleen al voor f 2500,- handel als ik die in het warenhuis had moeten kopen.


dH (in de verdediging gedrongen) Ja, maar ik heb V&D alleen maar computers geadviseerd waar geen programma's voor zijn, zodat ze veel geld kunnen verdienen aan de software die ze zelf laten maken. Het gaat niet om de gebruiker, maar om de verkoper.

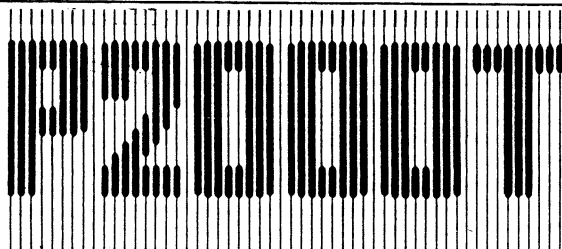
U (in de aanval nu) Volgens Het Parool hebt u gezegd dat je de P2000 maar het beste in de vuilnisbak kunt gooien.

dH Nee, nee. Dat heb ik niet gezegd. Dat hebben die stomme journalisten ervan gemaakt. Ik zie nu wel in dat hij daar gewoon te groot voor is. En te goed.

Hier aangekomen hebt u een goede kans uw P2000 te hebben verkocht. Is het niet letterlijk aan de heer M. d. H., dan wel figuurlijk aan uw vrouw en kinderen, aan uw moeder, aan uw buurman of aan de afgunstig kijkende bezitter van een niet met name genoemde andere computer. Of aan uw alter ego.

Rob Geutskens

I  my



Er zijn verschillende manieren om een programma te laten wachten tot een toets is ingedrukt, om daar vervolgens iets mee te doen. De oude BASIC-UK en de nieuwe BASIC-NL kennen beide de instructies "INPUT" en "LINE INPUT".

"INPUT A" kunt u in een programma gebruiken als u een waarde, dus een getal wilt invoeren. Die waarde kan uit een lange reeks cijfers bestaan. Drukt u per ongeluk een lettertoets in, of een andere toets die geen cijfer invoert, dan krijgt u de foutmelding "Fout! opnieuw".

"INPUT A\$" kunt u op dezelfde manier gebruiken als u een string wilt invoeren, bestaande uit letters, cijfers en leestekens. Voert u nu een numerieke waarde in, dan beschouwt de computer die niet als een getal, maar als een string. Hij kan er dus niet mee rekenen.

"LINE INPUT A\$" kan net als "INPUT A\$" worden gebruikt om een string in te voeren. Het verschil met "INPUT A\$" is dat er niet automatisch een vraagteken op het scherm wordt afgedrukt en dat "LINE INPUT" een lege string aflevert als u alleen de regeltoets ("ENTER") indrukt (bij "INPUT A\$" blijft de oude waarde van A\$ behouden).

INP("")

Een functie die BASIC-UK niet kende, maar BASIC-NL wel is "INP("")". Deze functie heeft tot gevolg dat de computer wacht totdat een toets is ingedrukt. Daarna geeft hij de ASCII-waarde van die toets.

"INP("")" is alleen te gebruiken om één toets te onderzoeken. U kunt de functie dus niet gebruiken om getallen of strings in te voeren. Maar per slot van rekening komt het in programma's vaak voor dat er maar één toets hoeft te worden ingedrukt om het programma in staat te stellen een bepaalde bewerking uit te voeren. Een bekend voorbeeld hiervan is bij voorbeeld "Verder gaan (J/N) ?", waarbij de computer verwacht dat u een J of een N indrukt. Een ander voorbeeld zijn keuzemenu's, waarbij u moet kiezen uit een aantal mogelijkheden door een enkele letter of een enkel cijfer in te toetsen.

Toch heeft deze functie enkele nadelen:

- 1 Hij maakt onderscheid tussen kleine en hoofdletters; toetst u "j" in in plaats van de "J" waarop het programma wacht (of omgekeerd), dan herkent de computer dit niet als "ja".
- 2 Als u de STOP-toets indrukt springt de P2000 uit het programma.
- 3 Enkele toetsen zijn niet "aangesloten" en geven als ASCII-waarde 7.

Het eerste probleem kunt u ondervangen door zowel op "j" als op "J" te controleren, bij voorbeeld:

```
100 A%=INP("")
110 IF A%=74 OR A%=106 THEN 120 ELSE 100
120 PRINT "Ja"
```

Als u uit meer mogelijkheden moet kunnen kiezen en het programma ook nog op zowel kleine als hoofdletters wilt laten reageren, wordt het programma onnodig lang. Dirk Kroon geeft hiervoor in zijn boek "BASIC-notities voor de P2000T" een elegante oplossing:

```
10 A%=INP(""):A$=CHR$(A%)
20 B%=INSTR("JjNnWw",A$)
30 ON B% GOTO 100,100,200,200,300,300:GOTO 10
```

Het tweede probleem, dus het afvangen van de STOP-toets, kunt u in een programma bereiken door "ONERRORGOTO" te gebruiken en in de foutafhandelingsroutine te kijken of de foutmelding nummer 64 heeft (met "IF ERR=64 THEN is de stoptoets ingedrukt"). Zie hiervoor eveneens het boek van Dirk Kroon (pag. 44).

Voor het derde probleem is met "INP("")" geen goede oplossing mogelijk. U kunt natuurlijk in uw programma opnemen: "IF INP(")=7 THEN doe dit of dat". Het programma doet dan dit of dat als u SHIFT + 2, 4 ("INL"), 6 ("OPN") of 8 van het kleine toetsenbord indrukt, maar het maakt geen onderscheid tussen die toetsen omdat ze allemaal de ASCII-waarde 7 geven.

Toetscodes uitlezen

Als u in een programma één toetsindruk wilt uitlezen, dan kunt u dat doen door gebruik te maken van de routine die begint op adres &H0026 (in de monitor). Eerst moet die routine worden gedefinieerd met DEFUSR; daarna kunt u hem in uw programma aanroepen. Dat kan als volgt:

```
10 DEFUSR=&H26
20 T=USR(0):PRINT T
```

In regel 20 ("T=USR(0)") wacht de computer op een toetsindruk. Er doet zich echter een klein probleempje voor. Na terugkeer uit de routine is T=0. Blijkbaar zet de routine de code van de ingedrukte toets ergens weg, waarna hij T gelijk maakt aan de waarde die we aan de USR-functie meegeven. Als u in regel 20 intoetst "T=USR(99)", dan krijgt u T=99-terug.

De toetscode van de laatst ingedrukte toets is echter te vinden op adres &H600D. Als u het programma als volgt uitbreidt, werkt het wel:

```

10 DEF USR=&H26
20 T=USR(0):T=PEEK(&H600D):PRINT T
30 GOTO 10

```

Eén van de aardige dingen van dit programma is dat de toetscode met SHIFT altijd 72 hoger is dan zonder SHIFT. Daarmee kunt u in één instructie bij voorbeeld "j" en "J" onderscheppen. Dat gaat met de functie "MOD". Een voorbeeld: "IF T MOD 72=14 THEN PRINT "Ja" ELSE IF T MOD 72=25 THEN PRINT "Nee".....". Drukt u nu de kleine letter "j" in, dan krijgt u de toetscode 14; drukt u de hoofdletter "J" in, dan is de toetscode 86, en 86 MOD 72 levert weer 14 op, zijnde de "rest" die overblijft als 72 op 86 wordt gedeeld.

(Opmerking: met "INP("")" kunt u dit niet doen, ook al is de ASCII-waarde van een kleine letter altijd 32 hoger dan die van de hoofdletter)

Een andere aardigheid is dat het indrukken van de STOP-toets niet meer tot het afbreken van het programma leidt. Elke keer als u in het laatste programma STOP indrukt, krijgt T de waarde 88 en verder gebeurt er niets. U kunt nu uw programma zo maken dat bij het indrukken van STOP de computer bij voorbeeld vraagt "Wilt u werkelijk stoppen (J/N)", zonder de foutafhandelings-routine aan te roepen ("IF ERR=64 THEN"). Ook het volgende stukje programma kan bruikbaar zijn:

```

10 DEF USR=&H26
20 T=USR(0):T=PEEK(&H600D):IF T=88 THEN GOSUB 1000
30 GOTO 20
1000 PRINT"Zo makkelijk komt u niet van me af !"
1010 RETURN

```

Oppassende lezertjes zullen opmerken dat dit een ideale manier is om nooit meer het programma te kunnen verlaten, behalve dan met een paardemiddel: door op RESET te drukken of de computer uit te schakelen. Maar dat is niet zo. Drukt u tweemaal snel achtereen op STOP, dan wacht de computer op de derde STOP. Komt die, dan krijgt u "STOP in 20" op het scherm en staat de computer in de directe stand. U kunt het programma dus altijd onderbreken door driemaal achtereen en met zo kort mogelijke tussenpozen STOP in te drukken.

De derde aardigheid is min of meer de omgekeerde van de eerste. Sommige toetsen geven altijd dezelfde ASCII-waarde, onverschillig of u al dan niet SHIFT hebt ingedrukt. Dat geldt onder meer voor de spatiebalk, de terugtoets ("RETURN"), CODE, de punt en de komma. Maar die toetsen geven met en zonder SHIFT wel een verschillende toetscode. U kunt uw programma nu dus zo inrichten dat CODE iets anders doet dan SHIFT + CODE, en RETURN iets anders dan SHIFT + RETURN.

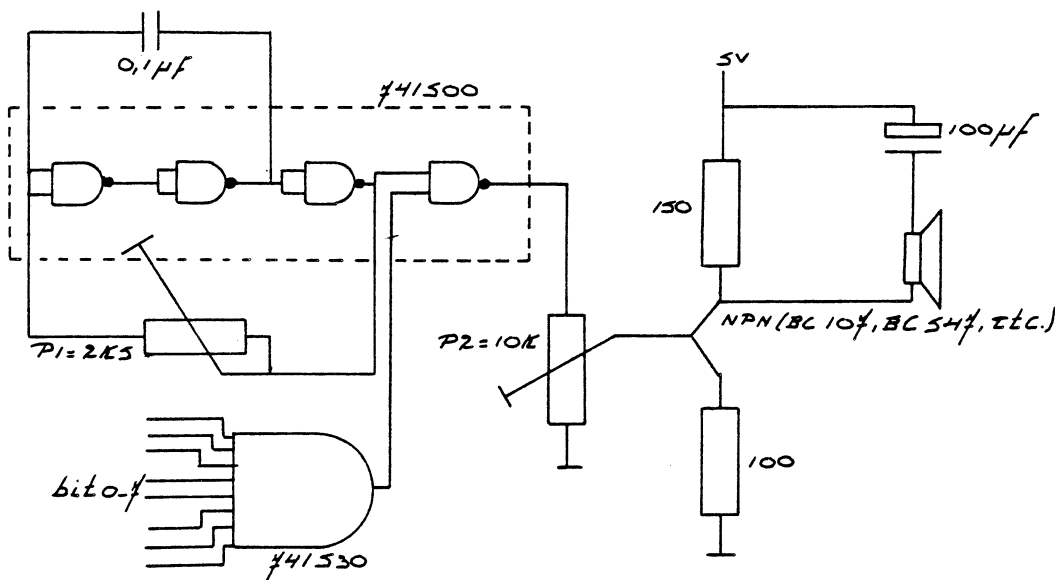
De vierde en laatste aardigheid die ik kan bedenken is dat u aan functieloze toetsen, zoals INL en OPN (die ASCII-waarde 7 opleveren), een functie kunt geven zonder met toetscodetabellen te gaan rommelen. Met "IF T=139 THEN" kunt u bij voorbeeld het programma naar een bepaalde regel sturen als u "INL" indrukt.

Rob Geutskens



Piepend toetsenbord

Met de onderstaande schakeling klinkt bij iedere toetsindruk een piepje uit de luidspreker.



Bit 0 t/m 7 zijn afkomstig van het toetsenbord. Het zijn de derde tot en met tiende draad van rechts, wanneer u normaal voor het toetsenbord staat. Deze draden moeten worden AFGETAKT, dus niet ONDERBROKEN OF DOORGEKNIPT worden. U kunt zelf controleren of u de juiste draden aftakt, door van tevoren de spanning op een draad te meten. Deze moet ca. 3,5 volt zijn en wanneer u een toets indrukt die op deze draad is aangesloten (even zoeken) ca. 0,7 volt. De impedantie van de luidspreker moet 100-600 Ohm zijn. Erg geschikt zijn de kleine luidsprekertjes die in veel kleine draagbare radio's zitten. Deze passen mooi in de P2000, vlakbij het toetsenbord, net voor de cassetterecordereenheid. Ook de schakeling zelf kan hier ondergebracht worden. Met P1 is de toonhoogte regelbaar, met P2 de geluidsterkte. Eventueel kan de piep uitgeschakeld worden met behulp van een schakelaar in de voedingsleiding.

Bron: Bas Wolfs.
Nat.Lab.Thuis Computer Club.



Soms is het nuttig twee programma's te kunnen koppelen, bij voorbeeld als u een programma waarin nogal wat is gerommeld, opnieuw van ordentelijke regelnummers wilt voorzien met het programma "Hernummer". Dit programma, dat begint met regelnummer 64000, moet achter het programma worden geplakt dat u wilt hernummeren.

Een voorwaarde voor het koppelen is dat de regelnummers van de twee programma's elkaar niet overlappen. Verwijder dus zonodig de programmastaart met regelnummers hoger dan 64000.

Het koppelen van het hernummerprogramma met het programma dat u wilt hernummeren gebeurt als volgt. Laad het te hernummeren programma op de normale manier met CLOAD"Naam". Als dit programma in het geheugen staat toetst u in de directe stand in:

```
POKE &H625C,PEEK(&H6405)-2:POKE &H625D,PEEK(&H6406):NEW
```

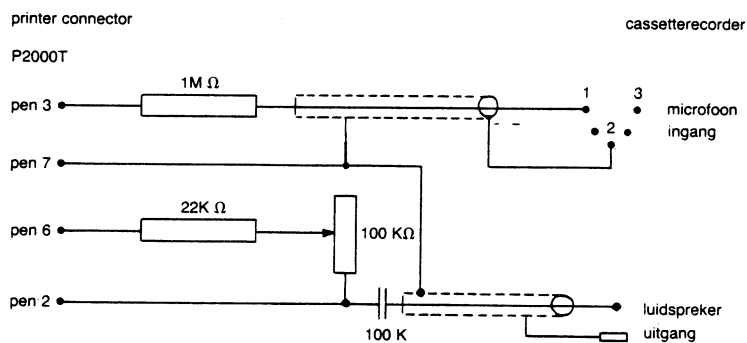
Wat u hiermee doet is: kijken waar de eindwijzer van het BASIC-programma staat (die staat twee posities na het laatste byte van het programma), om vervolgens de beginwijzer (die normaal op &H6547 staat) aan het einde van het programma te zetten. Geeft u LIST, dan is het programma schijnbaar verdwenen. De P2000 denkt dat er geen programma in het geheugen aanwezig is.

Laad nu het hernummerprogramma met CLOAD"Hernummer". De P2000 zet dit programma achter het eerste programma omdat daar de beginwijzer staat. Hierna moet u de beginwijzer weer terugzetten naar de plaats waar hij hoort, namelijk op &H6547. Dit doet u (in de directe stand) met:

```
POKE &H625C,&H47:POKE &H625D,&H65
```

Geeft u nu LIST, dan zult u zien dat de twee programma's keurig achter elkaar staan. U kunt nu het hernummerprogramma laten lopen met RUN 64000 (niet met START, en ook niet met RUN zonder regelnummer, want dan start u het programma dat u wilt hernummeren).

Aansluiten van de P2000T op een tweede cassetterecorder:



Er zijn verschillende methoden om de uitvoer (de "output") van een programma op een bepaalde plaats op het scherm af te drukken. De meest bekende methode is:

```
PRINT CHR$(4)CHR$(vert)CHR$(hor) "af te drukken gegevens"
```

De af te drukken gegevens worden dan op regel vert en kolom hor op het scherm afgedrukt. Deze methode zullen we hier aanduiden met "CHR\$(4)".

In plaats van iedere keer weer CHR\$(4) CHR\$(vert) CHR\$(hor) in te tikken, kan men deze instructie als een gedefinieerde functie in het programma opnemen, bij voorbeeld als:

```
10 DEF FNCC$(V,H) = CHR$(4) + CHR$(V) + CHR$(H)
...
100 PRINT FNCC$(10,5)A$:REM Druk A$ af op regel 10 kolom 5
```

Deze methode heet in de tabel "DEF FN".

In plaats van de gedefinieerde functie kunt u ook een "afdrukstring" definiëren en deze aanroepen bij het afdrukken. Dat gaat bij voorbeeld zo:

```
10 S$ = CHR$(4) + CHR$(10) +CHR$(5)
...
100 PRINT S$ A$: REM Druk A$ af op regel 10 kolom 5
```

Deze methode zullen we aanduiden met "S\$".

Tenslotte kunt u nog de cursor naar een bepaalde plaats op het scherm POKEn alvorens af te drukken. Bij voorbeeld:

```
100 POKE &H60B3,4:POKE &H60B4,9:PRINT A$:REM POKE cursor
    horizontaal op kolom 5, verticaal op regel 10 en druk af
```

De laatste mogelijkheid zullen we aanduiden met "POKE".

Het effect van deze vier methoden is hetzelfde, maar de snelheid verschilt nogal. Die snelheid hebben we op de volgende manier gemeten, waarbij we gebruik hebben gemaakt van de "interrupt clock" van de P2000:

```
5 PRINT CHR$(2): REM zet cursor uit
10 POKE &H6010,0: POKE &H6011,0: REM klok op nul
20 FOR I%=1 TO 1000: printopdracht: NEXT
30 PRINT PEEK(&H6010) + 256*PEEK(&H6011)
```

Na correctie voor de tijd die de FOR-NEXT-lus zelf nodig heeft, vinden we de volgende tabel:

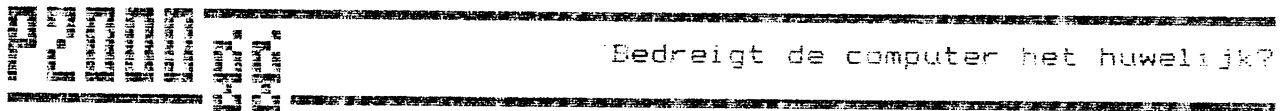
	S\$	CHR\$(4)	POKE	DEFFN	
A\$ = 1 karakter lang	3,7	8,8	6,3	15,7	ms
A\$ = 200 karakters lang	89,6	94,7	92,0	102,7	ms
A% = integer getal van 2 cijfers	6,9	12,0	9,3	18,9	ms
A! = enkeleprecisie-var. (4 cijfers)	12,7	17,9	15,2	24,3	ms
A# = dubbeleprecisie-variabele	47,7	52,8	48,9	54,4	ms
PRINT USING "##.##";A!	20,3	25,4	22,7	32,3	ms

Uit deze tabel blijkt dat de methode met de vaste string ("S\$") in alle gevallen de snelste is. De methode met "CHR\$(4)" kost ongeveer 5 milliseconden meer, met "POKE" ongeveer 2,5 ms meer en met "DEF FN" zelfs 12 ms meer.

Het gebruik van "PRINT USING" met plaats voor vier cijfers kost ongeveer 8 ms meer dan A! met vier cijfers. Dat is de prijs die moet worden betaald voor een "nette" afdruk.

Het afdrukken van strings gaat blijkbaar snel. Als er niet al te veel cijfers moeten worden afgedrukt, helpt het om het getal eerst in een string om te zetten. "PRINT LEFT\$(STR\$(A!),6)" kost minimaal 16,1 ms, en "PRINT USING "##.##";A!" neemt ten minste 20,3 ms in beslag.

Dirk J. Kroon



Gelezen in het Utrechts Nieuwsblad van 10 september 1984.

LOUGHBOROUGH (AP) - vrouwen moeten het tegen een nieuwe rivaal opnemen in de strijd om de aandacht van hun man: de computer. Dit meldt een onderzoekster van de universiteit van Loughborough in Groot-Brittannie.

Margaret Shotton zegt dat zij honderd 'computer-weduwen' heeft ontdekt, wier mannen zich na hun werk naar huis spoeden om eindeloos met hun computer te spelen.

De vrouwen komen gewoon niet aan bod, aldus Shotton in de Daily Express. Hun mannen zijn de hele dag op hun werk. Als ze thuis komen zijn ze tot drie, vier vier uur 's nachts met hun computer in de weer.

De computer is 24 uur per dag beschikbaar en dat maakt hem veel erger dan andere hobbies. De computer staat altijd voor ze klaar, aldus Shotton.

Sebastiaan Hijmans Utrecht



In de BASIC-NL is een mogelijkheid aanwezig voor het afhandelen van functietoetsen.

Voorwaarden voor het gebruik van functietoetsen zijn:
1 alleen in directe stand, dus niet in een programma
2 alleen op een lege regel

Voor BASIC-NL zijn voor FORMAT, LIST, ZOEK, START en DEF functietoetsen gedefinieerd. Met een klein machinetaalprogramma is het mogelijk zelf functietoetsen te definiëren. Functietoetsen hebben een ASCII-code van &H80 en hoger, BASIC-NL zelf gebruikt de code's &H80 t/m &HB7.

De vertaaltabel van toetscode naar ASCII-code staat vanaf adres &H1814. Dit adres zelf staat op adres &H6096. Door dit adres te veranderen kan een geheel nieuwe vertaaltabel worden gemaakt. Omdat dit wat omslachtig is, kunnen ook enkele toetsen in een tabel worden gewijzigd. Het adres van deze tabel staat op &H6094 en wijst na het opstarten naar een lege tabel. Door dit adres te vervangen door een adres dat het begin van een eigen tabel aangeeft, kunnen we de toetsen veranderen.

Voorbeeld (alle getallen in HEX)

adres	inhoud		commentaar
6094-6095	50	9F	adres nieuwe tabel
9F50	02		aantal gewijzigde toetsen
	toetscode	ASCIIcode	
9F51-9F52	22	32	"a" is nu "2"
9F53-9F54	1D	45	"b" is nu "E"

Om te kijken of er extra functietoetsen gedefinieerd zijn, springt de BASIC naar &H60E9. Normaal staat hier een RETURN. Door hier een JUMP naar de eigen tabel te zetten, zijn extra functietoetsen te definiëren.

Een tabel voor functie-toetsen wordt steeds voor 8 toetsen gemaakt.

Voorbeeld (alle getallen in HEX)

adres	inhoud		commentaar
6094-6095	27	65	adres vertaaltabel
60E9	C3		JUMP naar functietestprogr.
60EA-60EB	30	65	adres functietestprogramma

TOETSTABEL

adres	toetscode	ASCII-code	
6527	04		aantal wijzigingen
6528-6529	7A	88	toets SHIFT 8 kleine toetsenb.
652A-652B	8B	89	toets INL
652C-652D	88	8A	toets OPN
652E-652F	82	8B	toets SHIFT 2 kleine toetsenb.

FUNCTIETEST

adres	code	memonic	commentaar
6530	FE 88	CP 88	nieuwe functietoets ?
6532	D8	RET C	nee; terug naar BASIC
6533	21 38 65	LD HL,6538	laadadres functietabel
6534	C9	RET	return naar basic

FUNCTIE TABEL

adres	code	adres functie
6538-6539	00 E0	88
653A-653B	10 9F	89
653C-653D	F0 DF	8A
653E-653F	00 88	8B
6540-6541	00 00	8C
6542-6543	00 00	8D
6544-6545	00 00	8E
6546-6547	00 00	8F

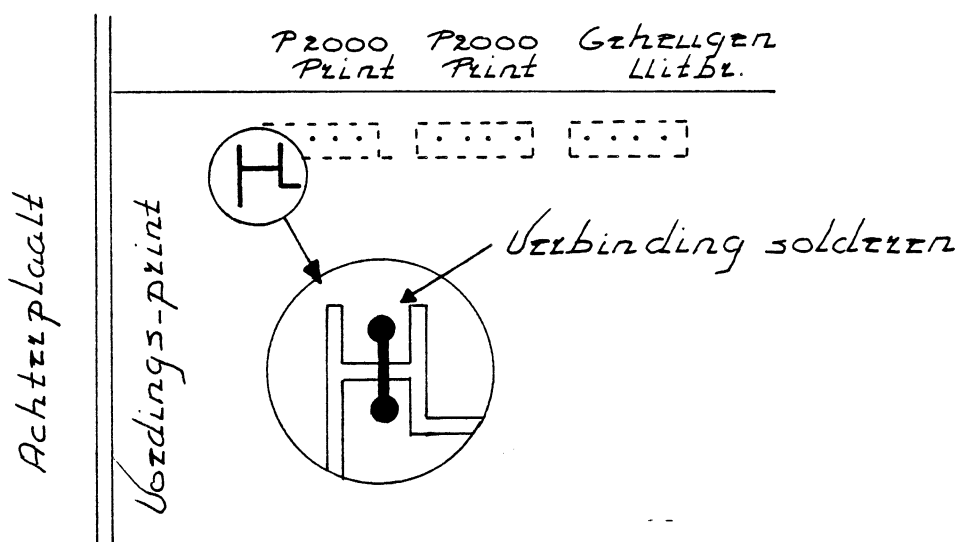
Vervolg op pag. 80

De hier gebruikte toetsen worden normaal niet door BASIC gebruikt, maar elke toets kan als functietoets worden gebruikt. Niet gebruikte codes in de functietabel worden met nullen gevuld.



Voeding van de P2000 verzwaren

Op de voedingsprint van de P2000 zitten connectortjes voor voeding van 3 prints. In het M-model worden die ook gebruikt. Elke connectortje heeft zijn eigen stabilisator voor 5 volt. De twee achterste connectors kunnen op de voedingsprint worden doorverbonden (zie onderstaande tekening), waardoor in principe meer stroom geleverd kan worden. Dit is speciaal interessant bij het aansluiten van interfaces, die veel stroom opnemen (EPROMprogrammer).



Op de bovenstaande tekening staat aangegeven waar een draadje moet worden gesoldeerd om de verbinding tot stand te brengen. Sluit nu het voedingskabeltje naar de processor print aan op het middelste of achterste connectortje en de eventuele geheugen uitbreiding op het overblijvende.

Praktijkvoorbeeld

Onderstaand programma kunt u zo in het monitorprogramma zetten, waarna het programma met SHIFT 2 kleine toetsenbord kan oproepen.

adres	code	memonic	kommentaar
670B	21 E9 60	LD HL,60E9	hier stond C9 RET
670E	36 C3	LD (HL),C3	
6710	23	INC HL	
6711	36 30	LD (HL),30	
6713	23	INC HL	
6714	36 65	LD (HL),65	
6716	11 27 65	LD DE,6527	
6719	ED 53 94 60	LD (6094),DE	
671D	21 27 67	LD HL,6727	
6720	01 1A 00	LD BC,001A	
6723	ED B0	LDIR	
6725	C9	RET	
6726	00	NOP	
6727	04		toetstabel
6728	7A 88		
672A	8B 89		
672C	88 8A		
672E	82 8B		
6730	FE 88	CP 88	functie-zoekprogramma
6732	D8	RET C	
6733	21 38 65	LD HL,6538	
6736	C9	RET	
6737	00	NOP	
6738	C4 00		functietabel
673A	C4 00		
673C	C4 00		
673E	00 XX		XX hangt af van loc. monitor.
6740	00 00		MON XX
6742	00 00		6800 78
6744	00 00		8800 98
6746	00 00		A800 B8
			C800 D8
			E800 F8

Succes

Ton Hilgersom



Goed nieuws voor P2000-gebruikers: er is een nieuwe, onafhankelijke vereniging opgericht, de Vereniging van Philips Thuis-Computergebruikers, kortweg PTC geheten. Over het hoe en waarom leest u uitgebreid in het onderstaande artikel.

Waarom een nieuwe vereniging ?

De P2000gg is de snelst groeiende gebruikersgroep van Nederland. In november 1983, toen het nieuwe bestuur aantrad, hadden we circa 800 leden. Nu, vijftien maanden later, zijn dat er ongeveer 4500; een gemiddelde groei van 250 leden per maand. Dat is natuurlijk verheugend, maar het brengt ook de nodige problemen met zich mee. Het maakt het voor het bestuur bij voorbeeld bijna ondoenlijk om alle nieuwe leden bericht te sturen. Al het werk moet namelijk in hun schaarse vrije tijd door vrijwilligers worden gedaan. En behalve het verwelkomen van nieuwe leden is er nog zoveel meer te doen: programmacassettes verkopen, gebruikersdagen organiseren, nieuwsbrieven publiceren en uiteraard het eigenlijke bestuurlijke werk. U hoort ons niet klagen, maar het is gebleken dat dit werk niet meer alleen door vrijwilligers kan gebeuren. Daardoor ook zijn we niet toegekomen aan nieuwe taken waar grote behoefte aan bestaat, zoals het steunen van regionale en plaatselijke groepen, het steunen van de bezitters van andere kleine Philips computers en het (laten) maken en verkopen van "hardware"-uitbreidingen voor de P2000. Het is ons al maanden geleden duidelijk geworden dat we moesten zoeken naar wegen om een deel van het werk uit te besteden aan professionals.

De eerste reden voor een nieuwe vereniging is dat het vele werk niet meer alleen door amateurs kan worden gedaan

Versnippering kan niet meer

Zoals u waarschijnlijk wel weet bestaat er ook nog een Philips P2000 Computer-Club, kortweg P²C² geheten, waarvan alleen Philips-medewerkers lid kunnen worden. Deze club is voortgekomen uit de fameuze NatLab Computerclub, een club van mensen die zeer veel hebben gedaan om de P2000 te maken tot wat hij nu is. De P²C² heeft ongeveer 1500 leden. Hoewel deze club wat kleiner is dan de P2000gg, werd ook het bestuur van de P²C² geconfronteerd met overbelastingsproblemen. Evenals bij de P2000gg zijn deze het gevolg van het feit dat veel nieuwe leden nieuwkomers zijn in computerland, die zeker de eerste tijd nauwelijks een bijdrage kunnen leveren aan de verdere ontwikkeling van hardware en software voor de P2000.

Eind 1984 zijn de besturen van P²C² en P2000gg gezamenlijk tot de slotsom gekomen dat het niet langer verantwoord was er twee of meer organisaties van P2000-gebruikers op na te houden, met even zovele besturen, redacties, software-commissies, hardware-commissies enzovoort. Wij hebben toen in principe besloten samen een nieuwe vereniging op te richten voor alle bezitters van een Philips computer. Enerzijds zouden wij dan een veel doelmatiger gebruik kunnen maken van de beschikbare actievelingen, anderzijds zou zo'n nieuwe vereniging sterk genoeg kunnen zijn om een belangrijk deel van het werk door betaalde krachten te laten uitvoeren.

De tweede reden voor een nieuwe vereniging is dus het voorkomen van versnippering en het bundelen van de krachten

De P2000 is een ouwe taaie, maar

De P2000T is waarschijnlijk de oudste nog leverbare thuiscomputer van dit moment. Alle andere thuiscomputers van zo'n vijf jaar geleden zijn inmiddels vervangen door nieuwere typen, die vaak niet de minste gelijkenis vertonen met hun voorgangers. Dat de P2000T nog steeds wordt geleverd, bewijst alleen maar dat het een machine is die nog volop in de race is. Hoewel we daar geen informatie over hebben, mogen we aannemen dat de P2000 nog geruime tijd leverbaar zal blijven. Toch zal er een moment komen dat Philips stopt met de P2000 zoals die nu wordt geleverd. Dat is het onverbidelijke lot van elk industrieel produkt, of het nu een auto of een thuiscomputer is. Dit betekent natuurlijk niet dat u uw P2000 in de vuilnisbak moet gooien op het moment dat hij niet meer te koop is. Ook van een auto die niet meer wordt gemaakt kun je nog jarenlang plezier hebben. Het betekent wel dat er nog vele jaren behoefte zal zijn aan steun voor P2000-bezitters.

Dit neemt niet weg dat er meer Philips thuiscomputers zijn en komen die de aandacht zullen vragen. Toen de P2000gg werd opgericht, kwamen alleen de P2000T en de P2000M in aanmerking. Vandaar de naam van de gebruikersgroep. Maar inmiddels is de reeks kleine Philips computers uitgebreid met de P2500 (die niet meer wordt geleverd), de P2000C en de P3100. Daarbij komen nog de VG 8010 en de VG 8020, twee MSX-thuiscomputers die Philips in maart zal introduceren.

Ook de bezitters van deze computers zullen behoefte hebben aan ondersteuning en aan informatie over hun computer. Na het voorgaande zal het duidelijk zijn dat de P2000gg die taken er niet nog eventjes bij kan nemen. De nieuwe vereniging zal dat wel kunnen omdat de krachten worden gebundeld en de vereniging sterk genoeg zal zijn om een groot deel van het "uitvoerende" werk uit te besteden aan betaalde medewerkers. En daarmee hebben we de derde reden voor het oprichten van een nieuwe vereniging:

De derde reden voor een nieuwe vereniging is dat de PTC steun zal kunnen geven aan de bezitters van ALLE kleine Philips computers

De HCC heeft het een beetje laten afweten

We willen ons hier niet afzetten tegen de Hobby Computer Club HCC, maar we kunnen er niet omheen er toch enkele opmerkingen over te maken. Eén van de voornaamste doelstellingen van de HCC is steun te verlenen aan de bezitters van home-, hobby- en thuiscomputers, ongeacht merk en type. In dit verband wordt vaak de vergelijking gemaakt met de ANWB, oorspronkelijk opgericht door wielrijders om hun belangen te behartigen, zoals het aanleggen van fietspaden en het plaatsen van wegwijzers. Daarbij speelde het merk van de fiets nauwelijks een rol. Of men nu een Batavus of een Gazelle berijdt, het nut van een fietspad is precies even groot.

Bij computers blijkt dat toch even anders te liggen. De bezitter van een P2000, een Apple, een Commodore en noem maar op, blijkt vrijwel uitsluitend belangstelling te hebben voor informatie over de eigen computer, om de eenvoudige reden dat hij weinig of niets kan beginnen met informatie over andere computers. Al kort na de pioniersfase begon de HCC dan ook al ontbindingsverschijnselen te vertonen. Vooral de grotere gebruikersgroepen vonden dat zij veel te weinig aan de bak kwamen, met name op het gebied van publicaties in de HCC-Nieuwsbrief.

Als u weet dat bijna een kwart (23%) van alle HCC-leden lid is van de P2000gg, en als u vervolgens kijkt hoeveel procent van de HCC-Nieuwsbrief aan de P2000 is gewijd, dan is het duidelijk dat wij gedwongen waren een eigen P2000gg-Nieuwsbrief te publiceren. Maar omdat de f 45,- contributie

vrijwel helemaal opgaat aan de HCC-Nieuwsbrief, het organisatorische apparaat van de HCC, de regionale afdelingen en dergelijke, waren wij genoodzaakt voor onze eigen Nieuwsbrief een abonnementsprijs te vragen van f 15,- per jaar. Veel leden vroegen zich dan ook af waarom zij nog lid waren van de HCC, als zij extra moesten betalen om wat informatie over hun eigen computer te krijgen.

Voor deze onbevredigende situatie zijn verschillende oorzaken aan te wijzen:

- * Een computer is nu eenmaal geen fiets.
- * Het HCC-bestuur, dat geheel bestaat uit vrijwilligers, is al evenmin tegen het vele werk opgewassen als het bestuur van de P2000gg. Al zou het willen, dan zou het ons nog niet de steun kunnen geven die wij als gebruikersgroep nodig hebben om goed te kunnen functioneren.
- * De HCC-bestuursleden, die verantwoordelijk zijn voor de Nieuwsbrief, willen een blad maken dat in de kiosk kan concurreren met de commerciële computerbladen, in plaats van een blad van en voor de leden.

Laat er geen misverstand over bestaan dat de P2000gg niet alleen verreweg de grootste computergebonden gebruikersgroep van de HCC is, maar ook de trouwste. Andere grote gebruikersgroepen zijn ons al veel eerder voorgedaan door met stichtingen en andere constructies te trachten meer voor hun leden te doen dan binnen HCC-verband mogelijk blijkt te zijn. Het is het P2000gg-bestuur in 15 maanden met moeite gelukt enkele afspraken met het HCC-bestuur te maken, en het is niet gelukt om die afspraken geëffectueerd te krijgen. Dat geldt bij voorbeeld voor de afspraak dat wij per nummer 2 pagina's in de HCC-Nieuwsbrief ongcensureerd over de P2000 zouden mogen publiceren.

De vierde reden voor een nieuwe vereniging is dat de HCC ons niet voldoende steun kan geven

Wat willen de leden ?

De problemen die wij hebben geschetst dateren niet van vandaag of gisteren, en bij het zoeken van een oplossing is het P2000gg-bestuur niet geheel op eigen houtje te werk gegaan. Tijdens de ledenvergadering van 3 november vorig jaar heeft het bestuur de problemen uit de doeken gedaan en de leden gevraagd in welke richting een oplossing moest worden gezocht. Een schriftelijke stemming had als resultaat dat de ledenvergadering het bestuur opdracht gaf te zoeken naar wegen om naast de P2000gg een organisatie (gedacht werd aan een stichting) in het leven te roepen voor het financieel beheer. In die organisatie zou onder meer de verkoop van programma's en apparatuur ondergebracht moeten worden, zodat de gebruikersgroep zelf het beheer zou kunnen voeren over de financiële middelen die de leden bijeen brengen door middel van de verkoop van programma's en dergelijke.

Het bestuur is van mening dat het naar beste kunnen aan de opdracht van de ledenvergadering heeft voldaan door, samen met het bestuur van de P²C², een nieuwe vereniging in het leven te roepen die zelf geheel en al verantwoordelijk zal zijn voor het beheer.

De vijfde reden voor een nieuwe vereniging is de opdracht van de ledenvergadering van 3 november 1984

Wat is de PTC ?

De Vereniging Philips Thuiscomputergebruikers PTC is een onafhankelijke vereniging voor bezitters van een kleine Philips computer. Het Algemeen Bestuur van de vereniging zal uit vijftien personen bestaan en zal, zoals bij

verenigingen gebruikelijk is, door en uit de leden worden gekozen. Omdat de bestuursverkiezingen pas kunnen plaats vinden als de vereniging goed draait, zal een voorlopig bestuur worden gevormd. Hiervoor hebben zich de volgende personen beschikbaar gesteld:

Dick van den Berge	Gerben Mooiweer
Frits Feldbrugge	Gerrit Pompert
Rob Geutskens	Jan van Rekum
Herman Heijting	Klaas Robers
Sebastiaan Hijmans	Elly Stolze - Hougee
Wil Hildering	Wim de Vrijer
Ton Hilgersom	Ad Welten
Ted Jonker	Hans Wildeman
Dirk Kroon	

Uit het Algemeen Bestuur zal een Dagelijks Bestuur worden gevormd, dat waarschijnlijk uit vijf personen zal bestaan.

Bureau PTC

In het voorgaande hebben we geconstateerd dat het niet meer mogelijk is een grote vereniging van computergebruikers helemaal door amateurs te laten runnen, als een masochistische vorm van vrijetijdsbesteding. Er moet een scheiding worden gemaakt tussen beleidszaken en de uitvoering van het uitgestippelde beleid. Om die reden zal naast de PTC een uitvoerend orgaan worden opgezet, dat kortweg Bureau PTC zal heten.

Het bureau zal worden bemand en bevrouwd (bemenst dus) door betaalde, deskundige medewerkers (m/v). Het is lastig in kort bestek een precieze afbakening te geven van de taken van Vereniging en Bureau, en bovendien ligt die nog niet helemaal vast. In grote lijnen zal het er echter op neer komen dat het Bureau PTC onder meer de volgende uitvoerende taken zal krijgen:

- * het voeren van de ledenadministratie van de Vereniging PTC;
- * het verzorgen van de financiële administratie;
- * het verzorgen van de programmadistributie (het administratief verwerken van bestellingen van cassettes, het kopiëren en verzenden daarvan enz.);
- * het laten maken van uitbreidingen en het verzorgen van de "hardware"-distributie;
- * het uitgeven (d.w.z. laten zetten, drukken en verzenden) van het verenigingsorgaan, dat "PTC PRINT" zal gaan heten;
- * het beantwoorden van schriftelijke en telefonische vragen.

De Vereniging zal de taken houden die P2000gg en P²C² nu ook hebben, ontdaan van het uitvoerende deel. In grote lijnen:

- * het beoordelen en vrijgeven van door de leden gemaakte programma's;
- * het ontwikkelen van "hardware"-uitbreidingen die in produktie genomen zullen worden;
- * het voeren van de redactie van het verenigingsorgaan PTC PRINT;
- * het organiseren van demonstraties en manifestaties;
- * het stimuleren en steunen van plaatselijke en regionale gebruikersgroepen;
- * het organiseren van ledenvergaderingen en andere bijeenkomsten.

De Vereniging en het Bureau zullen "self supporting" moeten zijn. De kosten zullen dus worden bestreden uit de opbrengsten, dat wil zeggen

de contributies en een bescheiden winst uit de verkoop van programma-tuur en apparatuur. Anderzijds zal de PTC geen winst hoeven te maken. Een eventueel batig saldo zal dus op de een of andere manier ten goede komen aan de leden.

Samenvattend: de leden van de PTC zullen bepalen hoe de vereniging zal functioneren en wie het bestuur zullen vormen. Democratischer kan het niet. Het gekozen bestuur zal zich laten bijstaan door een professioneel bureau, dat een belangrijk deel van het uitvoerende werk voor zijn rekening zal nemen.

Een jaar gratis lidmaatschap voor P2000gg-leden

Iedere bezitter van een P2000 of een andere kleine Philips computer kan lid worden van de PTC. De contributie is voorlopig vastgesteld op f 30,- per jaar. Voor dat bedrag zal men gebruik kunnen maken van alle faciliteiten van de vereniging, en het verenigingsorgaan ontvangen, dat in principe zesmaal per jaar zal verschijnen (dat zal dit jaar misschien niet helemaal lukken omdat het jaar al een eindje op gang is).

Als de PTC een feit is, zal de P2000gg stoppen met een aantal activiteiten (daarover straks meer). Wij willen degenen die hun HCC-contributie voor 1985 hebben voldaan echter niet in de kou laten staan. Daarom is besloten dat u als P2000gg-lid dit jaar gratis gebruik kunt maken van de faciliteiten van de PTC, op voorwaarde dat u uw HCC-contributie voor 1985 hebt betaald en u zich aanmeldt door middel van het bijgesloten aanmeldingsformulier.

Wenst u een abonnement op het nieuwe blad van de vereniging, dan kunt u zich daar op abonneren voor f 15,-. Geef dit aan op het aanmeldingsformulier. U ontvangt dan te zijner tijd een acceptgirokaart voor het abonnementsgeld. Hierbij merken wij op dat u als P2000gg-lid ook voor een abonnement op de P2000gg-nieuwsbrief tot nu toe f 15,- moest betalen. Het lidmaatschap van de PTC kost u voor 1985 dus niets extra's, maar levert u wel een aantal interessante voordelen op.

Wat gebeurt er met de P2000gg ?

Het antwoord op deze vraag kan kort zijn: de P2000gg blijft als afdeling van de HCC gewoon bestaan. Daar zijn verscheidene redenen voor aan te voeren. In de eerste plaats heeft de P2000gg geen eigen rechtspersoonlijkheid. Alleen de HCC-ledenraad zou de P2000gg kunnen opheffen, en waarom zou zij dat doen ? In de tweede plaats zijn sommige P2000gg-leden ook nog lid van andere gebruikersgroepen, bij voorbeeld de CP/M-groep. En in de derde plaats onderschrijft het bestuur van de P2000gg nog steeds de doelstellingen van de HCC, ook al komt daar in de praktijk minder van terecht dan we zouden wensen.

Een aantal taken van de P2000gg zal echter overgaan naar de PTC. Wij zullen stoppen met:

- * het publiceren van een eigen nieuwsbrief (dit is het laatste nummer, dat niet alleen aan de abonnees maar aan alle leden is toegezonden);
- * het verkopen van cassettes met programma's;
- * het verkopen van "hardware"-uitbreidingen (een activiteit die er de laatste tijd een beetje bij is ingeschoten);
- * het beantwoorden van telefonische vragen op woensdagavond.

De P2000gg zal doorgaan met het organiseren van gebruikersdagen in de Technische School "De Bron" in Utrecht. De data in 1985 zijn:

- * 4 mei
- * 31 augustus
- * 23 november

De gebruikersdag van 4 mei staat nog niet vast. Deze datum volgt kort op het weekeinde van 13 en 14 april, waarop een grote landelijke manifestatie zal worden gehouden. Elders in deze nieuwsbrief leest u er meer over.

Verder zal de P2000gg doorgaan met het verkopen van programmadiskettes voor de P2000M, de P2000C en de P2000T met disk drives.

Wat mag u als PTC-lid verwachten ?

Als lid van de PTC, dat wil zeggen als u zich hebt aangemeld via bijgaand formulier, zult u gebruik kunnen maken van alle faciliteiten die de nieuwe vereniging te bieden heeft, met uitzondering van het tijdschrift PTC PRINT (voor 1985 kunt u zich daar afzonderlijk op abonneren voor f 15,-). Die faciliteiten zijn:

- * Het kopen van programmacassettes, voorlopig tegen dezelfde lage prijzen als de P2000gg tot dusver hanteerde. De bestelprocedure blijft voornamelijk dezelfde en omdat het gironummer van de PTC nog niet bekend is, kunt u tot nader aankondiging het oude gironummer blijven gebruiken. U stort dus f 20,- per cassette + f 2,50 verzendkosten per zending op:

giro 1 93 17 69
t.n.v. HCC-P2000gg
Schiedam

Vermeld op de girokaart uw HCC- of PTC-lidmaatschapsnummer en de nummers van de cassettes die u wilt ontvangen. (Niet-leden betalen f 25,- per cassette + f 2,50 verzendkosten.)

- * Het kopen van "hardware"-uitbreidingen, voor zover die niet via de detailhandel worden verkocht. Alle uitbreidingen die tot dusver alleen voor leden van de P²C² te koop waren, zullen binnenkort voor alle PTC-leden verkrijgbaar zijn via het Bureau PTC. Welke artikelen dat zijn, wat ze gaan kosten en hoe u ze moet bestellen is nog niet bekend. Zodra dat bekend is, zal het worden gepubliceerd in PTC PRINT.

- * Het bezoeken van regionale bijeenkomsten. Het is bekend dat veel leden behoefte hebben aan bijeenkomsten vaker dan drie- of viermaal per jaar en dichter bij huis dan "De Bron" in Utrecht. De PTC zal dan ook het opzetten van plaatselijke en regionale gebruikersgroepen zo veel mogelijk bevorderen en deze groepen steunen. Bedenk hierbij dat het opzetten van een netwerk van plaatselijke en regionale gebruikersgroepen, dat het hele land bestrijkt, veel voorbereiding vraagt en niet van de ene op de andere dag te organiseren is. Het kan dus nog even duren voordat in uw omgeving een regionale of plaatselijke gebruikersgroep actief is. Bovendien zullen in alle delen van het land mensen moeten worden gevonden die bereid zijn zo'n gebruikersgroep op te richten en te besturen. Als u bereid bent daaraan mee te werken, meld dit dan op het aanmeldingsformulier.

* Het stellen van telefonische vragen als u met problemen zit. Het telefoonnummer van het Bureau PTC wordt: (040) 78 45 37. De mogelijkheid telefonisch vragen te stellen komt in de plaats van het telefonische spreekuur op woensdagavond. Zolang het Bureau PTC nog niet telefonisch bereikbaar is, kunt u op woensdagavond tussen 18.30 en 22.30 uur bellen met Rob Geutskens, (04990) 74290.

* Het stellen van schriftelijke vragen. U kunt schriftelijke vragen als volgt adresseren:

Bureau PTC
Postbus 67
5600 AB EINDHOVEN

* Het tegen gereduceerde prijs kopen van publicaties over uw computer, zoals boekjes en documentatie. Een voorbeeld daarvan is de "Monitor-listing", een volledige lijst van alle machinetaalroutines van de monitor (adressen &H0000...&H0FFF) in de P2000, uitgespit en van Nederlands commentaar voorzien door Ron Eijthoven. Ook de beschrijving van de geheugenadressen &H6000...&H6547, die wij hebben gepubliceerd in Nieuwsbrief nr. 8, zal te zijner tijd in een uitgebreide en verbeterde versie bij de PTC verschijnen. Prijs en wijze van bestellen zullen worden gepubliceerd in PTC PRINT.

* Het ter beoordeling insturen van programma's. Hebt u een aardig programma gemaakt dat interessant kan zijn voor andere gebruikers, stuur het dan in naar de PTC. De (nog te vormen) programmacommissie zal het dan beoordelen op punten als originaliteit, uitvoering en bedieningsgemak. Wordt het programma goedgekeurd, eventueel nadat u het hebt verbeterd, dan zal het in principe in het programmabestand in Viditel en op cassette worden opgenomen. De maker zal dan een passende beloning krijgen.

De liefde kan niet van één kant komen

Ondanks de bundeling van krachten en het professionaliseren van een deel van het werk, die de voornaamste redenen zijn voor het oprichten van een nieuwe vereniging, zal het succes van de PTC afhangen van de activiteiten van de leden. De leden zullen de plaatselijke en regionale gebruikersgroepen gestalte moeten geven, de programma's maken en de artikelen schrijven voor het verenigingsblad PTC PRINT. Het Bureau PTC kan niet meer doen dan de administratie daarvan voeren, de door u gemaakte programma's distribueren en de door u geschreven artikelen publiceren. Hebt u een aardig programma of een interessant verhaal, aarzel dan niet het in te sturen naar:

Bureau PTC
Postbus 67
5600 AB EINDHOVEN

Het Bestuur van de P2000gg

In de vorige nieuwsbrief heeft een uitprintroutine gestaan voor (super)snelzoekboek, waarmee het gehele beeld wordt uitgeprint. Zonder deze mogelijkheid te verliezen, is het mogelijk een extra toevoeging te maken, waarmee ook het uitprinten van een adressticker mogelijk wordt. Omdat de onderste regel anders niet meer op het scherm past, moet ook de "scherm-uitprint" ietsje worden veranderd: Voortaan moet je op "P" (van Print) drukken voor het uitprinten van het hele scherm en op "S" (van Sticker) voor het maken van de adres-sticker.

Een paar opmerkingen vooraf: Er wordt uitgegaan van de veronderstelling, dat de naam, het adres en de postcode + woonplaats de eerste drie gegevens vormen. Is dat niet zo, maar heb je bijvoorbeeld eerst een lidmaatschapsnummer dan zet je in regel 10010 FORI=2T06 i.p.v. FORI=0T04. De Postcode en de Woonplaats moeten op dezelfde regel staan, dus niet als in het voorbeeld van SZB op aparte regels.

Drie regels komen op de sticker. Wil je er vier, zet dan in regel 10010 FORI=0T06 en in regel 10030 FORI=1T05. Moeten er 5 gegevens op de sticker, dan dus in regel 10010 FORI=0T08 en in regel 10030 FORI=1T04 enzovoorts. Er is uitgegaan van stickers die, inclusief tussenruimte, 9 regels hoog zijn. Ook dit is dus aan te passen in regel 10030.

Dan volgen nu de aanpassingen c.q. toevoegingen. Zowel in Super SZB als gewoon SZB moet je dezelfde subroutine toevoegen:

```
10000 REM STICKERS UITPRINTEN
10010 POKE&H60AD,13:POKE&H60AF,0:FORI=0T04STEP2:POKE&H60AE,I:PRINTCHR$(5);:NEXT I
10020 POKE&H60AD,0:POKE&H60AE,0:POKE&H60AF,23
10030 FORI=1T06:LPRINT:NEXT I
10040 RETURN
```

Vervolgens moet je de volgende regels veranderen c.q. toevoegen. In Super SZB:

```
8625 PRINTFNP$(24,1)M$CHR$(219)"opnieuw "CHR$(208)"rint "CHR$(197)"dit "CHR$(21
1)"ticker verder"CHR$(221);
8630 I=INP("");IFI=69DRI=101THEN9800ELSEIFI=16THEN8450ELSEIFI=19THEN8520ELSEIFI=
112DRI=80THENPRINTCHR$(15)CHR$(5);:GOTO8625ELSEIFI=115DRI=83THENGOSUB10000:GOTO8
630ELSE8630
```

en in gewoon SZB:

```
1190 PRINTFNP$(24,1)M$CHR$(219)"opnieuw "CHR$(208)"rint "CHR$(197)"dit "CHR$(21
1)"ticker verder"CHR$(221);
1205 IFTC=53THENPRINTCHR$(15);:IF INP(32)AND2THENPRINTFNP$(24,1)M$"zet printer aa
n";:GOTO1205ELSEPRINTCHR$(5);:GOTO1190
1206 IFTC=11THENIF INP(32)AND2THENPRINTFNP$(24,1)CHR$(21)M$"zet printer aan";:GOT
O1206ELSEGOSUB10000:GOTO1190
```

Johan Elzenga
Burg. Verheullaan 59
2396 EP Koudekerk a/d Rijn
Vidibusnr: 400010961

In het tijdschrift 80 Micro wordt in 6 afleveringen een machine-taal listing gegeven van een spel "BUGS from outer Space" (no's maart tot en met augustus 1984); een ruimtevaartactiespel met duikende en schietende UFO's. Bij mijn poging om dit spel in de P2000 te zetten deed zich het volgende probleem voor:

Er is een test nodig om vast te stellen of de UFO's de rand van het scherm bereikt hebben; in de oorspronkelijke versie die geschreven is voor een TRS-80 computer gaat dit op eenvoudige (en korte) wijze AND 40

RET Z.

Nl. de TRS-80 heeft regels met lengte = 64 = &H40 en de nummering gaat op de volgende regel aansluitend door. De P2000 daartegen heeft regellengte = 40 en elke volgende regel heeft een beginadres dat 80 hoger is. De elegante test bij de TRS-80 is mogelijk omdat $64 = 2^6$ een macht van twee is, zodat elk beginadres van een regel hier op 6 nullen eindigt ($64 = 01000000$

$128 = 10000000$ enz.)

Nu is weliswaar $40 = 32 + 8 = 2^5 + 2^3$ maar dit leidt niet tot een overeenkomstige eenvoudige test, omdat de veelvouden van 40 binair gezien, een geheel andere structuur hebben. Gewoonlijk wordt de plaats op het scherm het beste aangegeven, door een pointer in een tabel op te nemen die aangeeft hoever de UFO op de regel gevorderd is. Omdat het "BUGS"-programma tabellen gebruikt die, door in bouw, niet eenvoudig zijn aan te passen volgt hieronder een overzicht van een aantal manieren om toch, en liefst zo snel mogelijk, vast te stellen of "het eind in zicht is".

Voor we ons werk beginnen, kunnen we echter eerst nog wat vereenvoudigingen aanbrengen:

- 1) de te onderzoeken schermadressen liggen tussen 20480 en 22320; de kenmerken van deelbaarheid door 80 veranderen niet als we overal een 80 vanaf trekken d.w.z. we onderzoeken alleen de (gereduceerde) adressen tussen 0 en 1840.
- 2) omdat $80 = 5 \cdot 16$ moet een 80-voud zowel door 5 als door 16 deelbaar zijn; de computer kan heel gemakkelijk vaststellen of een getal een 16-voud is: in binaire vorm moet dit dan eindigen op 4 nullen (bv. $32 = 0010000$), de resterende bits behoeven dan alleen nog getest te worden op een 5-voud.

20480	=	0101	00000000	0000	→	0000	00000000	0000
20560	=	0101	00000101	0000	→	0000	00000101	0000
20640	=	0101	00001010	0000	→	0000	00001010	0000
randadres	const	signif.stuk	16voud				signif.byte	

Het blijkt dus dat van alle randadressen alleen het middenstuk (bit 4 t/m 11) significant is en dit past precies in 1 byte (=8 bit).

Routine voor bepalen van tijdsverloop: --

EI	FB	wacht op interrupt
HALT	76	teller voor aantal herhalingen
LD B,n	06 n	
LD HL, 0000	21 00 00	
LD (6010), HL	22 10 60	zet klok op nul
PUSH BC	C5	
.....	(bv. CALL 0 AF2=CD F2 0A)
POP BC	C1	
DJNZ	10 F9	herhaal
LD HL (6010)	2A 10 60	laad klok
CALL 3E9A	CD 9A 3E	zet tyd op scherm
RET	C9	

Deze routine gebruikt de P2000 klok op 6010 en 6011, die elke 20-mseceen verhoogd wordt. Vul de te onderzoeken routine in op de stippellijn (omdat deze routine gemakkelijk gewijzigd moet kunnen worden, kan dit het beste met een CALL nn opdracht). Omdat machinetaalroutines veel sneller zijn dan 20 ms, kan de routine tot 255 x herhaald worden via een teller in register B. Na afloop wordt de tijd in HL gezet en door CALL 3E9A op het scherm gezet. De opdracht HALT zorgt ervoor, dat het op nul zetten van de klok gebeurt tegelijk met het optreden van het 20 msec interrupt en niet ergens halverwege in een 20 msec (yclus); nl. door de opdracht HALT gaat de computer NOP opdrachten uitvoeren, totdat het tijd is voor de eerstvolgende 20 msec interruptpuls. (HALT is dus geen definitieve programma beëindiging zolang er maar interrupt mogelijk is, vandaar EI). Overigens nemen ook de PUSH, POP en DJNZ tijd in beslag, maar voor vergelijkende metingen is dit geen bezwaar. Als controle is een delay routine uit het cassettegedeelte ingevuld nl. CALL 0 AFZ= vertraag 520 msec; het resultaat op het scherm zegt 26 klokcycli, dat wil zeggen $26 * 20 = 520$ msec.

I

De eerste methode voor het delen berust op het herhaald aftrekken d.w.z. pak het deeltal, trek er 5 vanaf, test of het restant nul of negatief is, zo nee herhaal dan. Een mogelijke routine is:

CD BD, 00005	01 05 00	laad deler =5
LD HL, 0078	21 78 00	deeltal < 5 * 24 regels = 78H
XOR A	AF	Carry = 0
SBC HL, BC	ED 42	trek af
RETZ	C8	set Z als deelbaar
JRNC	30 FB	herhaal, als niet negatief
RET	C9	exit met Carry (niet deelbaar)

In het ongunstigste geval moet er 24 keer afgetrokken worden (er zijn 24 regels); deze routine, aangeroepen en gemeten door de tijdmeetroutine kan dan circa 54 keer herhaald worden binnen een enkele klokcyclus, d.w.z. 1 enkele "deling" duurt circa 20-m. sec.: $54 = 370$ microsec.

Deze methode duurt vrij lang omdat de aftrekcyclus zo vaak (max. 24 keer) doorlopen moet worden.

II

Een betere deelroutine is daarom de pen en papier methode nl. zoals een gewone staartdeling op papier wordt uitgevoerd: de deler wordt onder het deeltal gezet, afgetrokken, daarna wordt het resultaat 1 plaats opgeschoven (= 1 cijfer bijhalen). Als het deeltal 8 bits groot is, behoeft dus maar hoogstens 8 keer de aftrekcyclus doorlopen te worden. Een snelle (de snelste?) routine hiervoor staat in L.A. Leventhal "Assembly language programming" p8-15. Het deeltal komt aanvankelijk in HL, na aftrekken komt door doorschuiven (ADDHL,HL) uiteindelijk het quotient in L en de rest van de deling in H, d.w.z. het adres is deelbaar door 5 als na afloop H=0 (omdat het quotient in L voor ons niet van belang is kan de opdracht INCL ook nog vervallen).

LD HL,0078	21 78 00	deeltal in HL
LD BC,0805	01 05 08	B = teller = 8 keer / C = deler = 05
ADD HL,HL	29	shift deeltal, quotient links
LD A,H	7C	kan de deler afgetrokken?
SUB C	91	
JRC	38 02	nee, doe volgende stap
LD H,A	67	ja, dan deeltal 5 kleiner
INC L	2C	noteer 1 in quotient
DJHZ	10 F7	
RET	C9	

Deze routine duurt per deling circa 179 microsec.

III

In het decimale stelsel is direct te zien of een getal deelbaar is door 5, dan is het eindcijfer nl. 0 of 5. Nu is er een speciale code (de BCD code) waarin 2 decimale getallen in binaire vorm worden opgeslagen: in 8 bits en wel 4 bits voor elk cijfer bijvoorbeeld: $\underline{0\ 0\ 0\ 1} \mid \underline{0\ 1\ 0\ 1}$ stelt 15 voor.

decim=1 =5

Als we dus kunnen beschikken over de BCD vorm van het te testen adres is het voldoende om te kijken of de laatste 4 bits 0 of 5 zijn. Het probleem is nu de omzetting van binair BCD. De Z80 beschikt over speciale opdrachten voor BCD verwerking bijvoorbeeld: de DAA-opdracht, die na een optelling de accu inhoud zodanig aanpast dat daar direct de genoemde BCD codes verschijnen. De volgende routine schuift het (bi-naire) deeltal vanuit H, via de carry naar A; dan volgt de BCD correctie en het (voorlopige) resultaat wordt in C opgeborgen; het schuiven moet dus 8 keer plaatsvinden; tenslotte volgt de test of er 0 of 5 achteraan staat:

HL HL,nn	21 00 nn	deeltal in H
LD BC, 0800	01 00 08	B=teller; C=00
ADC HL,HL	ED 6A	schuif bit in carry
LDA,C	79	pak voorlopige resultaat
ADC A,A	8F	tel bit erbij
DAA	27	BCD correctie
LDC,A	4F	store resultaat
DJNZ	10 F8	herhaal

AND 7	EG 07	pak rechtse 4 bit van resultaat
CP 0	FE 00	test op 0
RETZ	C8	set Zvlag als deelbaar
CP 05		
CP 05	FE 05	test op 5
RET	C9	set Zvlag als deelbaar

Deze routine duurt per keer circa 185 microsec Ook hier blijkt dat de meeste tijd nodig is in het 8 keer herhalen van dezelfde lus.

IV

De nu volgende routine trekt steeds 2 bit tegelijk af, zodat de cyclus hoogstens 4x herhaald behoeft te worden. De werking berust op de volgende eigenschap van 5-vouden: een 5-voud verandert niet, als een ander 5-voud wordt afgetrokken. Als voorbeeld zullen we bekijken hoeveel elk binair bit bijdraagt om een 5-voud te vormen:

bit	7	6	5	4	3	2	1	0
binare waarde	128	64	32	16	8	4	2	1
rest bij deling door 5	3	4	2	1	3	4	2	1
'-' notatie	-2	-1	2	1	-2	-1	2	1

U ziet dat de resten een periodiek patroon vertonen, zodanig dat de rest voor bit 0 opgeheven wordt door bit 2, en evenzo heffen bit 1 en bit 3 elkaar op; een binair getal waarin dus (alleen) bit 0 en bit 2 voorkomen is dus een 5-voud (controle 00000101=5, evenzo 00001010= 10 dec voor bit 1 en 3)

	b7	b6	b5	b4	b3	b2	b1	b0
Stel nu gegeven het binare getal	0	0	1	0	0	0	1	1
Trek hiervan af het getal A =	a7	a6	a5	a4	a3	a2	1	1 = 35
	0	0 = C

A eindigt op dezelfde 2 bits als 35 zodat het verschil C eindigt op 2 nullen. Nu zijn 35 en C beide vijfvoud (of beide niet) als A ook een vijfvoud was. Voor welke keuze van a7 tot en met a2 = a een vijfvoud? Kies daarvoor a2=a0 en a3=a1 en a4=a5=a6=a7=0 dat wil zeggen:

copieer het blok b1, b0 in a, a0, en in a3, a2.

B = 00100011

A = 00001111

----- min.

C = 00010100

U ziet hoe het te onderzoeken getal B gereduceerd wordt tot een getal C met dezelfde eigenschap (deelbaar door 5) door aftrekken van een geschikt getal A. A ontstaat uit B door de laatste 2 bit-ten van B tweemaal achter elkaar te schrijven. C eindigt dan op 2 nullen. Het proces kan nu herhaald worden op bit 2 en 3 van C, waardoor deze op hun beurt tot nul herleid worden enz. Zo schuiven we in totaal 3 stappen door het getal B heen; als B een 5-voud was, dan moet na afloop C=0 zijn, zo niet, dan is ook C on-

gelijk aan 0.

Voorbeeld:

```

                                Algoritme:
      B=00100011 = 5vd ← a) pak 2 bits
      A=000011   ← b) schuif 2 plaatsen
----- min ← c) trek af
B1=  C=000101
A1=   0001
----- min          herhaal nog 2 keer
      0000
      00
----- min
      00          resultaat C=0
```

Omdat de 2 laatste bits toch steeds identiek nul worden, behoeven we ze niet steeds mee te nemen, daarom zijn ze in het voorbeeld niet meer opgeschreven; dit komt ook in de te gebruiken routine goed uit: i.p.v. de 2 bits naar links te schuiven, schuiven we dan het getal B 2 plaatsen naar rechts (SRL), dit komt neer op afhakken van C tussen het 1e en 2e bit in.

Een routine die het genoemde algoritme uitvoert is de volgende:

```

LDA, n          3E 23 zet deeltal in A
LDB, 03         06 03 teller voor 3 cycli
LD H A         67   zet deeltal klaar in H
AND 03         E6 03 pak 2 laatste bits
SRL H         CB 3C schuif deeltal 2x rechts
SRL H         CB 3c
SUB A,H        94   trek nu af: A - H
NEG           ED 44 corrigeer : H - A
LDH,A         67   bewaar resultaat voorlopig in H
DJNZ         10 F4 herhaal nog 2 keer
RET          C9
```

Deze routine duurt circa 94 microsec. een verdubbeling in snelheid vergeleken met de vorige. Nu moet deze routine nog uitgebreid worden met een gedeelte dat het constante adresbrok 20480 (=5000 Hex) aftrekt en een deel dat test of het adres een 16 voud is (test op 4 eind nullen). Het bijzondere is, dat blijkt dat de volledige (=uitgebreide) routine nog iets sneller is dan de bovengevonden 94 microsec.; dit is mogelijk door niet de "herhaal" opdracht DJNZ te gebruiken maar de routine 3 keer volledig achter elkaar uit te schrijven: dit bespaart circa 3 x 6 microsec en dit levert dus tijdwinst op in ruil voor wat extra geheugenruimte. De volledige routine, die test op 80-voud is:

```

LD HL, nn      21 00 50 laad het te onderzoeken adres
LD A OF       3E 0F   test op 16-voud
AND A,L       A5
RETNZ        CO      NZ= geen 16-voud
LDA, OF       3E 0F   trek 5000 H af via
```

AND A,H	A4	maskeren in A
ADD A,L	85	verzamel significant byte in A
LDH, A	67	store in A
AND 03	E6 03	pak laatste 2 bits
SRL H	CB 3C	schuif deeltal 2 x rechts
SRL H	CB 3C	
SUB A,H	94	trek af: A - H
NEG	ED 44	corrigeer H - A
LDH,A	67	bewaar in H
AND 03	E6 03	
SRL H	CB 3C	
SRL H	CB 3C	
SUB A,H	94	
NEG	ED 44	
LDH, A	67	
AND 03	E6 03	
SRL H	CB 3C	
SRL H	CB 3C	
SUB A,H	94	
RET	C9	Exit met Z vlag = 1 80-voud Exit met Z vlag = 0 geen 80-voud

De gehele routine duurt slechts circa 86 microsec. Als u de routine doorneemt, zult u bemerken dat in het significante byte dat in A verzameld wordt, de hoogste en laagste 4 bits in verwisselde volgorde staan: voor de deelbaarheidstest maakt dit geen verschil:

als $16 * A_{\text{hoog}} + A_{\text{laag}}$ een 5 voud is, dan is ook $A_{\text{hoog}} + 16 * A_{\text{laag}}$ een 5 voud (beide termen zijn veranderd met een 15 voud) Om de werking van de machinetaal routine toe te lichten volgt hier de BASIC "vertaling":

```

10 FORJ=20480 TO 22359:A=J           : REM test alle
                                     : adressen
20 A=A-20480                         : REM constante eraf
30 if A mod 16=0 then A=A/16 else nextJ : REM test 16-voud
40 FORI=1 TO 3                       : REM test 5-voud
50 A=A:4-(A AND 3)                  : REM hier is het in-
                                     : teger deelteken
70 if A=0 then printJ               : REM print de
                                     : gevonden 80-vouden
80 NEXTJ

```

Regel 40 - 60 is het "deel door 5" algoritme nl. ":" is het integerdeelteken (kleine toetsenbord links boven) d.w.z. A:4 deelt A door 4, maar laat de rest weg en dit is hetzelfde als het binaire patroon van A 2 plaatsen naar rechts te schuiven. - (A AND 3) pakt de 2 laatste bits van A en trekt ze af. (N.B. vergeet de haakjes niet, want - heeft een hogere prioriteit dan AND!)

U kunt de gehele operatie ook volgen door op kritieke plaatsen de waarde van A binair uit te laten printen door tussenvoegen

```

van:
GOSUB 100
100 FOR N=7 TO 0 STEP-1
110 PRINT (A AND 2↑N)/2↑N;:NEXTN           : REM ↑ is het teken
                                           voor exponent

```

Tenslotte nog 1 opmerking over het testen van andere eindkolommen: tot nu toe hebben we getest of de 1e kolom (20480 + n.80) bereikt is (bv. de eindkolom 20519 + N.80) bereikt is; als we willen weten of een bepaalde andere kolom bereikt is (bv de eindkolom 20519 + n.80) dan kunnen we deze kolom reduceren tot de beginkolom door aftrekken van 39 (20519-39=20480) daarna kan dan de gewone routine voor de beginkolom gebruikt worden.

Bijlage I

Onderzoek van de verwerkingsnelheden van BASIC opdrachten. Dit gebeurt met het volgende programma:

```

40 POKE&H6010,0:POKE&H6011,0:REM zet klok op nul
50 REM Vul hier routine in
60 PRINT PEEK(&H6010)+256*PEEK(&H6011):REM Lees klok

```

Door in 50 achtereenvolgens verschillende opdrachten in te vullen kunnen we hun benodigde tijd te weten komen. De tijd wordt gemeten in klokeenheden van elk 20 microsec groot. We proberen eerst een FOR-NEXT lus, met enkele varianten.

OPDRACHT	KLOKEENHEDEN
a) 50 FORI=1TO1000:NEXTI	75
b) 50 FORI=1TO1000:NEXT	56
c) 50 FORI=1TO1000 52 NEXTI	78
d) 30 DEFINTI 50 FORI=1TO1000:NEXT	34
e) 30 A=333:B=444 50 FORI=1TO1000:NEXTI	79
f) 30 I=0:A=333:B=444 50 FORI=1TO1000:NEXTI	75

U ziet hier hoe u uw BASIC programma moet opbouwen om het zo snel mogelijk te laten lopen: meerdere opdrachten per regel, NEXT zonder I, en vooral het gebruik van integer variabelen kan de snelheid meer dan verdubbelen; f) laat zien dat u de meest gebruikte variabele helemaal vooraan het programma moet declareren; de variabelen worden nl. in volgorde van declaratie in een lijst gezet en de interpreter doorloopt steeds deze lijst van bovenaf, zodat de eerst genoteerde variabele het vlugst gevonden wordt. U ziet de vertraging ten gevolge van de (verder nutteloze) variabelen A en B.

N.B. Hetzelfde geldt voor subroutines: plaats deze niet achteraan, maar juist vooraan in het programma; de opzoektijd is dan minder.

Nu de tijd voor de FOR-NEXT lus bekend is, kunt u andere

opdrachten in de FOR-NEXT lus opnemen om ook hun tijdsduur te vinden. Stel bv. u wilt een deling uitvoeren om te zien of adres 20491 aan de rand van het scherm ligt:

KLOKEENHEDEN

1) 30 DEFINTI 50 FORI=1TO1000:A=20491/40:NEXT	373
2) 30 DEFINTI:B=20491:C=40 50 FORI=1TO1000:A=B/C:NEXT	310
3) 30 DEFINTI,A,B,C:B=20491:C=40 50 FORJ=1TO1000:A=B/C:NEXT	374
4) 30 DEFINTI:B=20491:C=40 50 FORI=1TO1000:A=B C:NEXT	294

REM INTEGER DEELTEKEN

5) 30 DEFINTI,A,B,C:B=20491:C=40 50 FORI=1TO1000:A=B C:NEXT	198
6) 30 DEFINTI:B=20491:C=40 50 FORI=1TO1000:A=BMODC:NEXT	298
7) 30 DEFINTI,A,B,C:B=20491:C=40 50 FORI=1TO1000:A=BMODC:NEXT	202

U ziet in (1) dat het delen van getallen zelf vrij langzaam gaat; sneller is meestal variabelen in te voeren (B = 20491 en C = 40) en dan deze variabelen te delen (zie 2) A = B/C); het opzoeken van variabelen gaat nl. sneller dan het lezen en interpreteren van 5 losse ASCII cijfers. Er zijn meerdere "deeltokens" op uw toetsenbord aanwezig: het integerdelen (met) in 4) en 5), en de MOD opdracht in 6) en 7). Merkwaardig is, dat het gebruik van integervariabelen (vgl. 4-5 en 6-7) sneller gaat voor "gedeeld door" dan voor MOD, maar juist langzamer bij het gewone deelteken (vgl. 2-3). Eveneens bijzonder is de variatie die optreedt in 1) als andere getallen als adres gekozen worden bijvoorbeeld: A = 20480/40 duurt korter (369 eenheden) en kortere getallen duren (soms) langer bv. A = 2048/40 duurt 374 eenheden!

Dick van den Berge

P2000gg-bestand in Viditel

Oplettende Viditelkijkertjes zullen ongetwijfeld hebben gemerkt dat het bestand van de P2000gg in Viditel sinds medio december niet meer toegankelijk is. Het bestand is op ons verzoek onbereikbaar gemaakt omdat het verouderde informatie bevatte; het was sinds juli 1984 niet meer bijgewerkt door degeen die daar verantwoordelijk voor was. Het vernieuwen van zo'n bestand is een heel karwei, en tot onze spijt is het door alle drukte niet gelukt om bijtijds nieuwe pagina's aan te leveren. Waarschijnlijk zal op de vrijgekomen pagina's binnenkort actuele informatie over de Vereniging van Philips Thuiscomputergebruikers PTC worden gezet. Even geduld nog.

Uit enkele reacties heb ik de indruk gekregen dat sommige lezers van de Nieuwsbrief teleurgesteld waren dat in nr. 10 niet de beloofde derde aflevering over tekstverwerking stond. Helaas; de tijd heeft me ontbroken. Bovendien was die Nieuwsbrief toch al dik genoeg.

Om het goed te maken hierbij een extra lang verhaal over tekstverwerking, met een beschouwing over hoe u op eenvoudige wijze teksten kunt produceren met de kale P2000T, dus zonder tekstverwerkingmodule in de voorste brievenbus. Verder het programma "Simpeltekst" en een programma om bestanden, die u met het Familiegeheugen hebt gemaakt, onder BASIC af te drukken.

Eenvoudig beginnen

De meest eenvoudige manier om (onder BASIC) een stuk tekst te schrijven en te laten afdrukken is gebruik maken van de instructie LPRINT. Bij voorbeeld:

```
10 LPRINT"Dit is een poging om een stuk tekst af te
drukken."
20 LPRINT"Het is de simpelste manier om dat te doen."
30 enz.
```

Deze methode heeft een paar nadelen. Voor elke tekstregel moet u meer dan tien toetsen aanslaan die niets met de eigenlijke tekst te maken hebben, namelijk een regelnummer met spatie, het woord LPRINT en aanhalingstekens openen en sluiten. Een tweede nadeel is dat u de tekst niet eerst op het scherm kunt bekijken voordat u hem afdrukt. Het laatste nadeel kunnen we ondervangen door de tekst op te nemen in DATA-regels:

```
10 READ A$:PRINT A$:GOTO 10
100 DATA"Dit is een poging om een stuk tekst af te drukken."
110 DATA"Het is een simpele manier om dat te doen."
120 enz.
```

Met dit programma kunt u de tekst eerst bekijken. Is hij naar uw zin, dan verandert u in regel 10 PRINT in LPRINT, zodat de tekst niet naar het scherm maar naar de printer wordt gestuurd.

Een voordeel van deze eenvoudige methoden is dat u volop gebruik kunt maken van de geweldig praktische "editor" van BASIC-NL. U kunt naar hartelust typfouten verbeteren, woorden tussenvoegen of verwijderen en zelfs hele stukken tekst invoegen.

Minder toetsen aanslaan

Is er geen eenvoudige methode om het aantal extra toetsen, dat u bij elke tekstregel moet indrukken, te verminderen?

Wel degelijk. Twee zelfs. De eerste is het gebruik van de opdracht AUTO. Toetst u die in, dan zorgt de P2000 zelf voor het genereren van regelnummers. Daar hebt u dus geen omkijken meer naar en het scheelt 2 tot 5 toetsaanslagen. Bovendien kunt u dan niet per ongeluk twee keer hetzelfde regelnummer gebruiken, waardoor de eerste regel met dat nummer zou worden overschreven.

U kunt met AUTO bij een bepaald regelnummer beginnen en ook het increment zelf bepalen. Bij "AUTO 10000,20" begint de P2000 te nummeren bij regel 10000, gaat daarna naar regel 10020 enzovoort.

De tweede truc is elke regel te beginnen met PRINT in plaats van DATA omdat u voor PRINT het vraagteken kunt gebruiken. Dat gaat als volgt:

```
AUTO 10000
10000 ?"Hoe verminderen we het aantal aanslagen tot het
minimum ?"
10010 ?"enz.
```

In plaats van 10000 DATA"....." (12 extra aanslagen) kunt u nu volstaan met ?"....." (3 extra aanslagen); een vermindering met 75%. Als u dit probeert zult u zien dat u meestal ook de laatste aanhalingstekens nog kunt weglaten, zodat er maar twee extra aanslagen overblijven: ?".

Sjoemelen met tokens

Aan PRINT-regels hebt u niet zo veel. U zou elke PRINT met de editor moeten veranderen in LRPINT om een afdruk van de tekst te kunnen maken. Daarom gebruiken we een list.

De BASIC-"tokens" zoals PRINT, DATA en LPRINT staan in het geheugen van de P2000 in de vorm van een code. Deze codes staan in tabel 6 van uw gebruiksaanwijzing. DATA staat bij voorbeeld als code 131 (= &H83) in het geheugen, PRINT als 165 (&HA5) en LPRINT als 162 (&HA2) (in oudere drukken van de gebruiksaanwijzing staat per abuis &HA0). Het moet mogelijk zijn de P2000 de codes &HA5 te laten opzoeken en deze dan te veranderen in &HA2. PRINT wordt dan LPRINT. Maar dat is makkelijker gezegd dan gedaan. Als we namelijk een programma bedenken dat deze omzetting automatisch uitvoert, kunnen we de PRINT-opdracht niet meer in ons programma gebruiken omdat die overal zou worden vervangen door LPRINT. Daarom gaan we het anders doen, en wel met het volgende eenvoudige programma:

```
10 X=&H6547
20 Y=PEEK(X)+256*PEEK(X+1)
30 IF PEEK(X+4)=&HA5 THEN POKE X+4,&HA2
40 X=Y:IF Y=0 THEN PRINT "Klaar":END ELSE 20
```

Toets nu in de directe stand in: AUTO 10000 en daarna een aantal keren:

```
10000 ?
10010 ?
10020 ?
enz.
```

Geeft u hierna LIST, dan staat in de regels vanaf 10000 PRINT. Laat u het programma lopen en geeft u dan nogmaals LIST, dan blijkt vanaf regel 10000 PRINT te zijn veranderd in LPRINT. Maar in regel 40 is PRINT blijven staan ! PRINT wordt alleen veranderd in LPRINT als de opdracht aan het begin van een BASIC-regel staat. Hoe komt dat ?

```

10 CLEAR1000
20 FORI=&H61F0T0&H61F5:READA#:POKEI,VAL("&H"+A#):NEXT
30 DATA 7E,CD,26,00,77,C9
90 ONERRORGOTO9000
100 DEFUSR=&H61F0:DEFUSR1=&H23:DEFUSR2=&H20:B#=CHR$(132)+CHR$(1
57):C#=B#+CHR$(141):D#=C#+CHR$(135):E#=D#+CHR$(130):F#=B#+CHR$(1
35):G#=CHR$(130):S#=CHR$(12):DEFFNP$(Y,X)=CHR$(4)+CHR$(Y)+CHR$(X
):UIT#=CHR$(2)
110 PRINTS#E#SPACE$(9)"Simpeltekst":PRINT:PRINTB#:PRINTD#" 1 U
itleg":PRINT:PRINTB#:PRINTD#" 2 Tekst invoeren":PRINT:PRINTB#:P
RINTD#" 3 Tekst weergeven":PRINT:PRINTB#:PRINTD#" 4 Tekst afdr
ukken"
120 PRINT:PRINTB#:PRINTD#" 5 Tekst bewaren":PRINT:PRINTB#:PRIN
TC#CHR$(129)" S Stoppen":PRINT:PRINTB#:PRINTE#SPACE$(9)"Maak uw
keus"
200 T=USR(0):IFT=46DRT=59THEN6000ELSEIFT=63DRT=58THEN1000ELSEIF
T=40DRT=56THEN2000ELSEIFT=70DRT=67THEN4000ELSEIFT=50DRT=66THEN5000E
LSEIFTMOD72=11DRT=88THENPRINTS#:ENDELSE200
1000 PRINTS#:AUTO10000
2000 IR=40:IL=IR:GOSUB3000:RESTORE10000:PRINTS#
2010 IL=IR:PRINTUSR2(""):GOSUB2020:GOTO100
2020 FORN=1TO4:IB=1:READA#:GOSUB2030:NEXT:PRINTFNP$(24,3)F#" me
er "CHR$(156)CHR$(17):T=USR(0):PRINTS#:GOTO2020
2030 II=INSTR(IB,A#,"#"):IFI=0THENY#=MID$(A#,IB):IFLEN(Y#)=0TH
ENRETURNELSEGOSUB2040:RETURNELSEY#=MID$(A#,IB,II-IB):GOSUB2040:I
B=II+1:GOTO2030
2040 IFY#=""THEN2060ELSEIFLEN(Y#)<=ILTHENPRINTY#;:IL=IL-LEN(Y#)
:GOTO2060
2050 IFMID$(Y#,IL+1,1)=" "THENPRINTLEFT$(Y#,IL):Y#=MID$(Y#,IL+2
):IL=IR:GOTO2040ELSEIL=IL-1:IFIL=-1THENPRINT:IL=IR:GOTO2040ELSE2
050
2060 IFII=0THENRETURNELSEPRINT:IL=IR:RETURN
2500 FORI=20640TO22240STEP80:POKEI,20:POKEI+1,29:POKEI+2,7:NEXT
:RETURN
3000 PRINTUSR1(0):X=&H6547:PRINTS#FNP$(8,10)E#"Ogenblikje "CH
R$(156)
3010 Y=PEEK(X)+256*PEEK(X+1):IFPEEK(X+4)=&HA5THENPOKEX+4,&H83
3020 X=Y:IFY=0THENRETURNELSE3010
4000 REM Printeroutines
4010 IFINP(32)AND2THENPRINTS#FNP$(1,1)E#SPACE$(6)"Printer niet
klaar":PRINTFNP$(3,1)E#" Druk op een toets of op STOP":T=USR(0):
GOTO4020ELSE4025
4020 IFT=88THEN100ELSE4010
4025 LT=8:LK=10:IR=60:RP=66:BR=6
4030 PRINTS#UIT#E#" Printer instellen":PRINT:GOSUB2500
4035 PRINTFNP$(5,3)G#" (0...63; zie printerhandleiding)":PRINTFN
P$(6,3)G#" (Voorinstelling 8)":PRINTFNP$(4,4)"Lettertype"SPACE$(2
0);:INPUTLT:IFLT<0ORLT>63THEN4035
4040 PRINTFNP$(9,3)G#" (0...40; voorinstelling = 10)":PRINTFNP$(
8,4)"Linkerkantlijn"SPACE$(16);:INPUTLK:IFLK<0ORLK>40THEN4040
4045 PRINTFNP$(12,3)G#" (20...132; voorinstelling 60)":PRINTFNP$(
11,4)"Regellengte"SPACE$(19);:INPUTIR:IFIR-LK<10RIR-LK>132THEN4
045
4050 PRINTFNP$(15,3)G#" (1...255; voorinstelling 66)":PRINTFNP$(
14,4)"Aantal regels per pagina ";:INPUTRP:IFRP<20RRP>255THE
N4050
4055 PRINTFNP$(18,3)G#" (0...255; voorinstelling 6)":PRINTFNP$(1
7,4)"Aantal blanco regels"SPACE$(10);:INPUTBR:IFBR<10RBR>255THEN
4055

```


Om dat in te zien moeten we nog even herhalen hoe een BASIC-programma in elkaar zit. De eerste regel van het programma begint bij de P2000 doorgaans op geheugenplaats &H6547. In regel 10 van het programma kennen we deze waarde toe aan de variabele X. X geeft dus aan waar de eerste BASIC-regel begint.

Op de eerste twee posities van elke BASIC-regel (dus op X en X+1) staat het adres waar de volgende regel begint. In regel 20 bepalen we dit adres, dat we "onthouden" in de variabele Y. Op de volgende twee posities (X+2 en X+3) staat het regelnummer; daar doen we verder niets mee.

Het eerste token van de BASIC-regel staat op X+4. In regel 30 peeken we of daar misschien &HA5 staat, en als dat zo is poken we er &HA2 voor in de plaats.

In regel 40 wordt X gelijk gemaakt aan Y. X bevat nu dus het adres van de tweede BASIC-regel, en daar gaan we weer kijken of op de vijfde positie PRINT staat. Het programma gaat net zo vaak terug naar regel 20 totdat Y nul wordt, en dat is aan het einde van het BASIC-programma.

Met dit programma hebben we dus bereikt dat alle PRINT-opdrachten worden omgezet in LPRINT, maar alleen als ze aan het begin van een BASIC-regel staan. PRINT-opdrachten verderop in de regel worden met rust gelaten.

Verandert u in regel 30 "&HA2" in "&H83", dan worden alle PRINT-opdrachten aan het begin van een BASIC-regel omgetoverd in DATA.

"Simpeltekst"

Het voorgaande is de kern van het programma "Simpeltekst", dat u hierbij vindt afgedrukt. Dat het toch nog een tamelijk lang programma is geworden en er niet zo erg simpel uitziet, is een gevolg van het feit dat er een flink stuk tekst en uitleg in zit en dat ik wat extra snuisterijen heb ingebouwd, die het werken vergemakkelijken.

Voordat u het programma gaat overtypen eerst even een belangrijke opmerking. Als u een BASIC-regel intypt die met PRINT (of ?) begint, moet u na het regelnummer twee spaties intypen, anders wordt die PRINT veranderd in DATA en loopt het programma niet meer.

Voor de netheid en om vergissingen te voorkomen ben ik bij het intypen elke BASIC-regel begonnen met twee spaties. Alleen de DATA-regels van de uitleg (regel 7000...7799) hebben maar één spatie na het regelnummer. Sterker nog: Die regels heb ik ingetypt alsof het een gewone tekst betrof, dus na AUTO 7000,5 met ?".....". Het programma zelf heeft er netjes DATA-regels van gemaakt. Het gemak dient de mens.

Hoe zit "Simpeltekst" in elkaar ?

"Simpeltekst" is geen moeilijk programma. Er zitten een paar kunstgrepen in, waarvan de meeste in het voorgaande al ter sprake zijn gekomen, maar een hoogstandje op programmeergebied is het zeker niet. Zo is het ook niet bedoeld.

Toch is gebleken dat er veel P2000-gebruikers zijn die graag zo af en toe een programma eigenhandig willen intypen. Dan kun je beter zien hoe de programmeur bepaalde problemen heeft opgelost; of niet heeft opgelost. Ook kun je dan beter naar eigen smaak

```

4060 PRINTFNP$(20,7)E$"Goed zo ? (J/N)";T=USR(0):IFTMOD72=14THE
N4070ELSEIFT=88THEN100ELSE4010
4070 POKE&H6253,0:POKE&H60A1,0:POKE&H60A9,8R:POKE&H60AA,RP:POKE
&H60AB,IR:LPRINTCHR$(27);"!";CHR$(LT);:LPRINTCHR$(27);"1";CHR$(L
K);
4100 GOSUB3000:RESTORE10000
4110 IL=IR:PRINTUSR2(""):GOSUB4120:GOTO100
4120 IB=1:READA$:GOSUB4130:GOTO4120
4130 II=INSTR(IB,A$,"#"):IFI=0THENY$=MID$(A$,IB):IFLEN(Y$)=0TH
ENRETURNELSEGOSUB4140:RETURNELSEY$=MID$(A$,IB,II-IB):GOSUB 4140:
IB=II+1:GOTO4130
4140 IFY$=""THEN4160ELSEIFLEN(Y$)<=ILTHENLPRINTY$;:IL=IL-LEN(Y$
):GOTO4160
4150 IFMID$(Y$,IL+1,1)=" "THENLPRINTLEFT$(Y$,IL):Y$=MID$(Y$,IL+
2):IL=IR:GOTO4140ELSEIL=IL-1:IFIL=-1THENLPRINT:IL=IR:GOTO4140ELS
E4150
4160 IFII=0THENRETURNELSELPRINT:IL=IR:RETURN
5000 PRINTS$FNP$(1,1)E$"Wegschrijven programma + tekst"
5010 PRINTFNP$(3,1)E$"Toets naam in .....":PRINTFNP$(3,
21);:LINEINPUTNAAM$
5020 PRINTFNP$(5,1)E$" Is dit goed zo ? (J/N) ";:T=USR(0):IF
TMD72=14THENPRINT"Ja"ELSEPRINT"Nee":GOTO100
5030 CSAVENAAM$:GOTO100
5900 DELETE6010-7799:END
6000 REM Uitleg
6010 PRINTS$UIT$:RESTORE7000:J=1
6020 PRINTS$FNP$(1,1)E$"Uitleg 'Simpeltekst' pag. ";J:RE=3:G
OSUB2500
6040 RE=RE+1
6050 READA$:IFA$="**"THEN6070
6060 PRINTFNP$(RE,4)A$:GOTO6040
6070 PRINTFNP$(23,1)F$CHR$(130)" [terug"CHR$(146)"■■"CHR$(130)
"afdruk START verder]";
6075 T=USR(0):IFTMOD72=0THEN6080ELSEIFTMOD72=23THEN6090ELSEIFTM
OD72=56THEN100ELSEIFT=90THEN6100ELSEIFT=112THEN5900ELSE6075
6080 J=J-1:IFJ<1THENJ=1:GOTO6075ELSE6200
6090 J=J+1:GOTO6200
6100 IFINP(32)AND2THENPRINTFNP$(1,1)E$CHR$(136)SPACE$(8)"Geen p
rinter !";SPACE$(10):FORQ=1TO5000:NEXT:J=1:GOTO6020ELSEPRINTCHR$(
5):GOTO6075
6200 ONJGOTO6210,6220,6230,6240,6250,6260,6270,6280,6290
6210 RESTORE 7000:GOTO6020
6220 RESTORE 7100:GOTO6020
6230 RESTORE 7200:GOTO6020
6240 RESTORE 7300:GOTO6020
6250 RESTORE 7400:GOTO6020
6260 RESTORE 7500:GOTO6020
6270 RESTORE 7600:GOTO6020
6280 RESTORE 7700:GOTO6020
6290 GOTO100
7000 DATA"Met het programma 'Simpeltekst'
7005 DATA"kunt u op eenvoudige manier een
7010 DATA"tekst voor een brief of een rap-
7015 DATA"port samenstellen.
7020 DATA
7025 DATA"Als u in het keuzemenu kiest voor
7030 DATA"'2 Tekst invoeren' dan wordt het
7035 DATA"scherf gewist en verschijnt het
7040 DATA"regelnummer 10000 op het scherm.
7045 DATA

```

veranderingen aanbrengen. Om die reden zullen we het programma in grote lijnen beschrijven.

In regel 10 wordt een stringruimte van 1000 bytes gereserveerd. Een beetje minder zal waarschijnlijk ook wel werken.

In regel 20 worden 6 bytes machinecode (ze staan in regel 30) op een vrij plaatsje in het geheugen gezet. Ze vormen een eenvoudig stukje machinetaal dat wordt gebruikt om toetsen af te vangen. Het is een variant op de USSR-functie die ik in Nieuwsbrief 10 heb beschreven (in "Toetsen afvangen").

In regel 100 is deze USSR-functie gedefinieerd (DEFUSR=&H61F0), zodat de P2000 weet waar hij naar toe moet springen als de functie wordt aangeropen.

Er worden hier nog twee gebruikersfuncties gedefinieerd: USR1 schakelt het toetsenbord uit, USR2 schakelt het weer in. Deze functies worden gebruikt als de P2000 bezig is met PRINT omzetten in DATA.

Verder zijn in deze regel een aantal strings gedefinieerd, die worden gebruikt om de zaken netjes op het scherm te krijgen zonder elke keer hele reeksen CHR\$'s te moeten intoetsen.

De regels 110 en 120 zetten het menu op het scherm. Regel 200 roept de toets-afvangroutine aan. Het programma reageert alleen op de cijfers 1 tot en met 5, s, S en STOP. Er wordt naar de toetscodes gekeken, dus niet naar de ASCII-waarden. Het indrukken van een verkeerde toets heeft tot gevolg dat het programma dezelfde regel nogmaals uitvoert.

Regel 1000 is alles wat nodig is om het scherm te wissen en de P2000 regelnummers vanaf 10000 te laten genereren.

Tekst schouwen

Op regel 2000 begint de routine die nodig is om de tekst te kunnen schouwen op het scherm, voordat hij wordt afgedrukt. U vindt een beschrijving van deze routines in het boek "BASIC-notities voor de P2000" van Dirk Kroon. Ik heb ze gegapt omdat ze een slimme oplossing bieden voor het probleem van de afgebroken woorden. Een woord dat niet meer op een regel kan, gaat in zijn geheel naar de volgende regel. Het einde van een alinea, als dus op een nieuwe regel moet worden begonnen, wordt aangegeven met een hekje; een regel wit met twee hekjes.

In regel 2000 wordt de printbreedte IR op 40 gezet omdat het scherm meestal niet meer dan 40 tekens op een regel toelaat. Hebt u een "80-karakterprint", dan kunt u IR=80 nemen.

N in regel 2020 is een gok. Bij N=4 worden steeds vier DATA-regels ingelezen en op het scherm gezet. Bij gemiddeld 200 karakters per DATA-regel zijn dat 800 karakters, en dat kan er net op. Schrijft uw P2000 80 karakters per regel, dan kunt u N veranderen in 7 of 8.

Regel 2500 is hier een beetje verdwaald. Het is een routine die het scherm blauw schildert. Op de eerste, tweede en derde positie van elke schermregel zet deze routine achtereenvolgens de codes voor blauw (20), balk (29) en wit (2).

7050 DATA"U hoeft nu alleen maar een vraag-
7055 DATA"teken en aanhalingstekens te typen,"
7060 DATA"gevolgd door tekst."
7065 DATA
7070 DATA"BASIC maakt van het vraagteken au-
7075 DATA"tomatisch PRINT en het programma zet
7080 DATA"PRINT automatisch om in DATA.
7099 DATA"**
7100 DATA"U kunt in uw tekst alle letters,"
7105 DATA"cijfers en leestekens gebruiken,"
7110 DATA"behalve de aanhalingstekens."
7115 DATA"Gebruik daarvoor het teken ' boven
7120 DATA"de 7.
7125 DATA"
7130 DATA"U hoeft meestal de tekstregel niet
7135 DATA"af te sluiten met aanhalingstekens,"
7140 DATA"behalve als deze een komma bevat
7145 DATA"of met een spatie eindigt."
7150 DATA"
7155 DATA"Elke regel biedt plaats aan circa
7160 DATA"240 tekens, maar het verdient aanbe-"
7165 DATA"veling niet verder te gaan dan 200
7170 DATA"tekens (5 schermregels). Dan houdt
7175 DATA"u nog wat ruimte over om zo nodig
7180 DATA"later letters toe te voegen."
7199 DATA"**
7200 DATA"U kunt alle tekst achter elkaar
7205 DATA"typen. Als de printer op een nieuwe
7210 DATA"regel moet beginnen, toetst u #.
7215 DATA"Een regel overslaan geeft u dus aan
7220 DATA"met ##.
7225 DATA"
7230 DATA"Als de brief klaar is, drukt u op"
7235 DATA"STOP. Druk daarna op START om het
7240 DATA"programma opnieuw te starten."
7245 DATA"
7250 DATA"Met '3 Tekst weergeven' kunt u de
7255 DATA"tekst op het scherm bekijken voor-
7260 DATA"dat u hem laat afdrukken door de
7265 DATA"printer."
7270 DATA"
7275 DATA"Op de volgende pagina vindt u een
7280 DATA"voorbeeld."
7299 DATA"**
7300 DATA" VOORBEELD"
7305 DATA"
7310 DATA"10000 ?'Beste F2000-gebruiker.##Met
7315 DATA"deze tekst willen wij u laten zien
7320 DATA"hoe u de tekstregels kunt invoeren.#
7325 DATA"Een hekje betekent dat de printer
7330 DATA"10010 ?'op een nieuwe regel moet be-
7335 DATA"ginnen. Twee hekjes betekent een re-
7340 DATA"gel overslaan.##Wij hopen dat u ple-
7345 DATA"zierig met deze eenvoudige tekstver-
7350 DATA"werker zult werken. - -
7355 DATA"10020?'Hoogachtend,#de auteur.#Rob
7360 DATA"Geutskens##'"
7365 DATA"
7370 DATA"OPMERKING: in plaats van ' de echte
7375 DATA"aanhalingstekens (SHIFT 2) gebruiken

Omzetten van PRINT in DATA

In de regels 3000...3020 zult u gemakkelijk het programma herkennen dat we eerder hebben gebruikt om PRINT om te zetten in LPRINT. Hier wordt echter PRINT vervangen door DATA. Het programma is aangevuld met enkele instructies om tijdelijk het toetsenbord uit te schakelen en een bericht op het scherm te zetten.

Afdrukken van de tekst

Op 4000 beginnen de afdrukrouines. Deze zijn in grote lijnen gelijk aan de printrouines voor het scherm (vanaf regel 2000), maar ze zijn aangevuld met wat handige extra's. In regel 4010 kijkt de P2000 eerst even of er wel een printer is waaraan hij zijn tekst kan slijten.

In regel 4025 staan de voorinstellingen voor de printer: lettertype 8, linkerkantlijn op 10, regelbreedte 60, aantal regels per pagina 66 en aantal blanco regels tussen de pagina's 6. In de regels 4030...4055 kunt u deze parameters op het scherm veranderen. In regel 4060 krijgt u een herkansing voor het geval u een fout hebt gemaakt.

In 4070 worden de parameters weggepookt en wordt de printer ingesteld.

Vanaf 4100 zijn de routines vrijwel gelijk aan die vanaf 2000. Er is ook geen principieel verschil tussen afdrukken en op het scherm zetten. Het enige verschil is eigenlijk dat hier LPRINT staat in plaats van PRINT, zoals bij de schermrouines.

Bewaren op cassette

In regel 5000 beginnen de routines die u kunt gebruiken om het programma "Simpeltekst" plus de ingevoerde tekst weg te schrijven op een cassette.

De P2000 vraagt niet eerst toestemming om een programma met dezelfde beginletter te overschrijven. Kies dus een naam die niet met de "S" van "Simpeltekst" begint als u uw programma ongeschonden wilt bewaren, vooral niet als u de uitleg hebt verwijderd.

De P2000 wacht wel als het programma door het toevoegen van tekst zoveel langer is geworden dat het niet meer op de oorspronkelijke cassetteruimte past, en er na het programma nog een programma op de band staat.

Verkorten van "Simpeltekst"

Regel 5900 maakt het mogelijk "Simpeltekst" aanzienlijk te verkorten door de hele uitleg te wissen. U kunt natuurlijk het tikwerk aanzienlijk beperken door de regels 6010 tot en met 7799 überhaupt niet in te toetsen.

Ik ben uiterst spaarzaam geweest met REM-regels. Regel 6000 heeft echter een functie, namelijk dat hij automatisch wordt overgeslagen. Zijn de regels 6010...7799 aanwezig, dan springt het programma van regel 6000 naar 6010. Ontbreken ze, dan gaat het programma linea recta naar regel 7900, waar staat dat de uitleg is weggehaald.

In regel 6010 begint het programmadeel dat de uitleg op het scherm zet. Er worden vanaf regel 7000 DATA-regels ingelezen totdat het programma tweemaal een sterretje tegenkomt (het teken

7399 DATA"***
7400 DATA" CORRECTIES"
7405 DATA
7410 DATA"Om correcties aan te brengen in de
7415 DATA"tekst kunt u gebruik maken van de
7420 DATA"BASIC-editor, die uitgebreide moge-"
7425 DATA"lijkheden biedt om tekens te veran-
7430 DATA"deren, te verwijderen en tussen te"
7435 DATA"voegen.
7440 DATA
7445 DATA"Om een verandering aan te brengen in
7450 DATA"regel 10010 toetst u in 'EDIT 10010'.
7455 DATA"De regel verschijnt dan op het
7460 DATA"scherf en u kunt er met de cursor-
7465 DATA"toetsen doorheen wandelen. Zie ver-
7470 DATA"der de gebruiksaanwijzing van de
7475 DATA"F2000T voor de werking van de editor.
7499 DATA"***
7500 DATA"U kunt de EDIT-stand verlaten door
7505 DATA"op de CODE-toets te drukken. Drukt
7510 DATA"u daarna 'N' in, dan gaat de F2000
7515 DATA"naar de volgende regel, die u dan
7520 DATA"kunt bewerken."
7525 DATA
7530 DATA"U kunt ook tussen elke twee tekstre-
7535 DATA"gels maximaal 9 regels tussenvoegen.
7540 DATA"U dient dan zelf het regelnummer in
7545 DATA"te toetsen.
7550 DATA
7555 DATA"Een bestaande tekst wordt regel voor
7560 DATA"regel gewist als u '2 Tekst invoer-
7565 DATA"ren' kiest.
7570 DATA"Aan een bestaande tekst, die tot bv"
7575 DATA"regel 10120 loopt, kunt u tekst toe-"
7580 DATA"voegen door 'AUTO 10130' te typen.
7599 DATA"***
7600 DATA" BEWAREN"
7605 DATA
7610 DATA"U kunt een tekst, compleet met het"
7615 DATA"programma 'Simpeltekst' op cassette
7620 DATA"bewaren. Daarvoor kunt u elke naam
7625 DATA"gebruiken die niet met de hoofdlet-
7630 DATA"ter 'S' begint (anders wordt het
7635 DATA"oorspronkelijke programma over-
7640 DATA"schreven).
7645 DATA
7650 DATA"LET OP: De F2000 vraagt niet of
7655 DATA"het programma met dezelfde naam
7660 DATA"mag worden overschreven !
7665 DATA"Uiteraard kunt u een programma met
7670 DATA"de tekst ook weer van cassette in-
7675 DATA"lezen, bij voorbeeld om iets te ver-"
7680 DATA"anderen en de tekst opnieuw af te
7685 DATA"drukken.
7695 DATA"***
7700 DATA"Om het programma 'Simpeltekst' te
7705 DATA"verkorten, en zodoende meer ruimte"
7710 DATA"over te houden voor tekst, kunt u"
7715 DATA"deze uitleg verwijderen.
7720 DATA"Druk daartoe op de 'wis het scherm-

* heet officieel asterisk, maar dat kan ik net zo moeilijk uit mijn mond krijgen als obelisk). De regels 6210...6280 maken dat de P2000 bij een bepaald regelnummer DATA begint te lezen, zodat u vooruit en achteruit door de pagina's kunt bladeren.

Tenslotte regel 9000. Het programma springt hier naar toe als het een fout constateert. Er wordt maar één type fout afgevangen: fout nr. 4 ofwel "Out of DATA". Als het programma de laatste DATA-regel heeft gelezen, verschijnt de mededeling "niet meer" onder op het scherm. Drukt u een toets in, dan keert het programma terug naar het menu.

Uitbreiden van een bestaande tekst

Hebt u een tekst ingetikt op regel 10000 tot en met bij voorbeeld 10230, en wilt u daar regels aan toevoegen, dan zou dat moeten kunnen door via het menu "2 Tekst invoeren" te kiezen. De P2000 begint dan weer te nummeren bij regel 10000, maar als een regelnummer al bestaat wordt dit nummer afgedrukt met een sterretje. 10000* betekent dus dat dit regelnummer al is gebruikt. Drukt u de regeltoets met de grote haak in, dan zou de inhoud van regel 10000 ongeschonden moeten blijven; hij wordt echter gewist. De enige mogelijkheid om de tekst nog te redden is op STOP drukken.

Dit is waarschijnlijk een foutje in de BASIC, want de 24K-BASIC-interpreter doet het wel goed. Die laat de inhoud van de regel ongemoeid. Je kunt dan net zo lang de regeltoets indrukken totdat een regelnummer zonder sterretje verschijnt.

Om te voorkomen dat u bestaande tekst wist, kunt u bij BASIC-NL als volgt handelen. Kijk met LIST 10000- wat het hoogste regelnummer met tekst is. Is dat 10230, toets dan in de directe stand in: AUTO 10240. U kunt dan vrolijk regels toevoegen zonder iets te wissen.

Uitbreidingen en verbeteringen

Het programma is tamelijk haastig in elkaar gesleuteld omdat PBNA erop zat te wachten in verband met de cursus "Doe meer met uw P2000T". Doorknede BASIC-programmeurs zullen er ongetwijfeld verbeteringen in kunnen aanbrengen.

Natuurlijk zijn ook uitbreidingen welkom. Ik kan er zelf wel een paar bedenken:

- * het bewaren van de printerinstellingen in een array, dat met het programma en de tekst wordt weggeschreven;
- * de mogelijkheid lange woorden te voorzien van conditionele afbreekstreepjes die worden weggelaten als het woord midden in een regel komt maar worden gebruikt als het woord moet worden afgebroken;
- * de mogelijkheid alleen de regels vanaf 10000 weg te schrijven (dat is simpel) en weer in te lezen (dat is minder simpel want dan moeten ze gekoppeld worden met "Simpeltekst");
- * de mogelijkheid weinig gebruikte tekens zoals &, £ en \$ om te zetten in é, è, ë en dergelijke;
- * de mogelijkheid woorden te onderstrepen.

```

7725 DATA"toets', rechtsboven op het kleine"
7727 DATA"toetsenbord. Druk daarna op START.
7730 DATA
7735 DATA"Wilt u de uitleg niet verwijderen,"
7740 DATA"druk dan nu op START of op J om
7745 DATA"terug te keren naar het menu.
7750 DATA
7755 DATA"Veel plezier met 'Simpeltekst' !
7799 DATA"**
7900 PRINTS#FNP$(1,1)E#SPACE$(5)"Uitleg is weggehaald.":PRINTFN
P$(3,1)E#SPACE$(7)"Druk een toets in."
7910 T=USR(0):GOTO100
9000 IFERR=4THENPRINT:PRINTFNP$(24,3)F#"Niet meer "CHR$(156)CH
R$(17):T=USR(0):RESUME100:ENDELSEEND
10000 DATA

```

Ik heb met opzet de regelnummering ruim opgezet, zodat er plaats is voor fikse uitbreidingen, zonder dat het programma moet worden hernummerd.

Rob Geutskens



De fiscus en de P2000

Als belastingconsulent heb ik te maken met mensen die een Personal Computer hebben aangeschaft en deze voor de inkomstenbelasting als kostenpost ter verwerving van inkomen willen aftrekken.

Dit is in de meeste gevallen niet mogelijk.

Nu blijkt echter dat de P2000 met tekstverwerker en printer hierop een uitzondering is. In de meeste gevallen accepteert de belasting een aftrek van 50% van de aanschafwaarde van computer, printer, monitor, programma en dergelijke. Dit bedrag mag dan ineens of gespreid over 3 of 5 jaar worden afgeschreven. De belastinginspecties hanteren hiervoor verschillende richtlijnen. Overleg met uw belastinginspecteur verdient dus aanbeveling.

Voor mensen die hem, zoals ik, gebruiken in hun bedrijf, is hij vanzelfsprekend helemaal een kostenpost.

Kleine zelfstandigen kunnen hem ook alleen voor boekingswerk gebruiken, in combinatie met bij voorbeeld een boekhoudbureau, maar dan is de investering alleen als kostenpost te verwerken. Hij is dan te klein voor opvoeren op de balans.

J. C. Reijngoud
R&R Administratie Service
Arcenstraat 19
Rotterdam

Het Familiegeheugen is een uitstekend programma om allerlei bestanden bij te houden: adressen, boeken, platen, tijdschrift-artikelen enzovoort. Een nadeel van het Familiegeheugen is dat de mogelijkheden om die bestanden netjes af te drukken, beperkt zijn. Het is bij voorbeeld nauwelijks mogelijk de printer in te stellen; de bestanden worden afgedrukt zoals ze op het scherm verschijnen. Vandaar het nu volgende programma, waarmee onder BASIC Familiegeheugen-bestanden kunnen worden afgedrukt.

Eerst nog even een trucje, dat soms uitkomst kan bieden en dat ook leuk is bij gebruik van de module Tekstbewerking. Zet de BASIC-interpretator in de P2000, zet de computer en de printer aan en kies het lettertype, de linkerkantlijn en dergelijke (met LPRINT CHR\$(27)...). Schakel dan de P2000 uit maar laat de printer ingeschakeld. Steek het Familiegeheugen of de tekstbewerker in de eerste sleuf en zet de computer weer aan. De instellingen van de printer blijven zo behouden.

Boekenbestand

Het hierbij afgedrukte programma is door Joep Willemsen geschreven om de geheugenadressen af te drukken, die zijn verzameld met het Familiegeheugen. Met het oorspronkelijke programma is het overzicht afgedrukt dat in Nieuwsbrief 8 stond. Ik heb me gepermitteerd er een paar dingen aan te veranderen; niet uit eigenwijsheid, want ik vind het een buitengewoon knap programma, maar uit praktische overwegingen.

In de eerste plaats wil niet iedereen een overzicht van geheugenadressen afdrukken, en in de tweede plaats zat achter het programma een stukje machinetaal geplakt. Bij LISTen is dat niet te zien. Daarom heb ik het programma geschikt gemaakt voor een boekenbestand en de machinetaal opgenomen in een DATA-regel. U kunt het programma nu dus zo overtypen.

Het programma werkt uitstekend op mijn printer, een FX-80. Misschien zijn voor andere printers kleine aanpassingen nodig. Let vooral op "_". Als ik het programma op het scherm LIST, staat er "£". Begin dus maar met "£" in te toetsen waar "_" in de afdruk staat.

Hoe werkt het ?

Het programma is bijzonder compact. Als u de programmastaart en de REM-regels 11, 12, 90 en 91 weglaat (in de laatste twee staat ' in plaats van REM, wat hetzelfde effect heeft), houdt u maar een handjevol BASIC-regels over. De auteur vertelde me dat hij zo'n programma op de normale manier opbouwt, met nieuwe regelnummers voor de meeste statements. Daarna stopt hij zo veel mogelijk instructies in één regel en laat alles weg wat maar enigszins gemist kan worden. Deze compacte bouw maakt het lastig het programma helemaal te verklaren, maar ik zal toch een poging doen de grote lijnen aan te geven.

In regel 10 wordt de vrije geheugenruimte bepaald en wordt ruimte gereserveerd voor het in te lezen bestand. Hier worden ook de parameters gedefinieerd, zoals regellengte LR, scherm-breedte SB en het aantal regels per pagina PL, en verder de

```

10 CLEAR2048,PEEK(&H6405)+PEEK(&H6406)*2
56+3072:DEFSTR:AR=35:LR=80:SB=70:FL=66:
ER=6:BA=PEEK(&H6405)+PEEK(&H6406)*256:DE
FUSR=&H61F0:DIMR(AR):R1=CHR$(27)+"F":R2=
CHR$(27)+"E":R3=SPACE$( (LR-SB)/2 ):R4=CHR
$(130):R5=CHR$(131):X=1
11 REM  AR = Aantal Regels per
      "fam.-scherm"
      LR = Lengte van print-Regel
      SB = SchermBreedte waarbinnen
          geprint wordt
      FL = PaginaLengte
      ER = Extra Regels per
          "fam.-scherm"
12 REM  BA = pointer naar einde BASic
      R() = fam-regels 1-AR
      R1 = dun gedrukt
      R2 = dik gedrukt
      R3 = spatie tot begin scherm
      X = aantal afdrukken
13 FORI=&H61F0TO&H61F5:READA$:POKEI,VAL(
"&H"+A$):NEXT
14 DATA 7E,CD,18,00,77,C9
15 PRINTCHR$(2)CHR$(28)R4CHR$(141)"
      Print-out FAM":PRINT:PRINT:PRINTCH
R$(6)R5"  Plaats cassette en druk op EN
TER";:H=(2*PEEK(&H605C)+(PEEK(&H605C)=3)
)*&H2000+&H5FFF
20 BA=BA+3073:I=INP(""):PRINTCHR$(15)R4"
Laad file";:DV=USR(0):DV=USR(1):IFDVTHEN
BOELSEPOKE&H6030,0:POKE&H6031,0:POKE&H60
32,1:POKE&H6033,0:POKE&H604F,1:DV=USR(6)
:IFDVTHENBOELSEIFPEEK(&H6041)<>65THENPRI
NT:PRINTCHR$(133)"Verkeerd file-type":CL
EAR50,H:END
25 IFBA+PEEK(&H6034)+PEEK(&H6035)*256>HT
HENPRINT:PRINTCHR$(133)"Te grote file":C
LEAR50,H:END
30 PRINTCHR$(15)R4"File-naam:"R5;:FORI=&
H6036TO&H603D:PRINTCHR$(PEEK(I));:NEXT:P
RINT:DV=USR(1):IFDVTHENBOELSEPOKE&H6030,
BAMQD256:POKE&H6031,BA/256:DV=USR(6):IFD
VTHENBO
35 PRINT:PRINTCHR$(6)R5"Zet printer goed
en druk op ENTER";:I=INP(""):PRINTCHR$(
15)R4"Print file";
40 FORTE=1TOX:POKE&H60AB,LR:POKE&H60A1,0
:POKE&H60AA,FL:N=1:L=1:LPRINTSPC(LR-20)C
HR$(14)"Pagina"R2;NCHR$(20):LPRINTSTRING
$(LR,"_"):LPRINT:LPRINT:FORI=BA+2TOBA+PE
EK(&H6034)+PEEK(&H6035)*256:P=PEEK(I):IF
P=255THENGOSUB100:GOTO70ELSEIFP=1THENGOS
UB100:R="":NEXT
50 IFP=13THENIFLEN(R(L))+LEN(R)>SBTHENL=
L+1:R(L)=R:L=L+1:R="":NEXTELSER(L)=R(L)+
R:R="":L=L+1:NXTELSER=R+CHR$(P):IFP=32T
HENIFLEN(R(L))+LEN(R)>SBTHENL=L+1:R(L)=R
:R="":NXTELSER(L)=R(L)+R:R="":NXTELSEN
EXT
70 LPRINTCHR$(12);:NEXTTE:PRINTCHR$(15)C
HR$(133)"Klaar":CLEAR50,H:END

```

```
80 PRINT:PRINTCHR#(133)"Cassette fout "C
HR#(DV):CLEAR50,H:END
```

```
90 ' Vanaf hier is het mogelijk om
    een eigen print routine te
    zetten voor je eigen
    FAMILIE-GEHEUGEN-FILE
91 ' -Start moet op regel 100 zijn.
    -In R() staan de regels van een
    "fam-scherm".
    -Regel 60000 moet bestaan
    blijven.
100 IFPEEK(&H60A1)+L+ER>=PLTHEN:N=N+1:LP
RINTCHR#(12)SPC(LR-20)CHR#(14)"Pagina"R2
;NCHR#(20):LPRINTSTRING$(LR,"_")
110 LPRINTR3;R1"Boektitel : "R2;R(1):LPR
INTR3;R1"Auteur : "R2;R(2):LPRINT:FOR
J=3TOL:LPRINTR3;R1;R(J):NEXT:LPRINTR3;ST
RING$(SB,"_")R1
60000 ERASER:DIMR(35):L=1:RETURN
65520 'P2000
65521 'Programmanaam: Print-out FAM
65522 'Versienr.: 1.0
65523 'Datum goedkeuring: Geen
65524 'Naam auteur: Joep Willemsen
65525 'Bewerking : Rob Geutskens
```

manier waarop de tekstdelen moeten worden afgedrukt. De verklaring van de parameters staat in de regels 11 en 12. U kunt ze zelf aanpassen aan bij voorbeeld de lengte van het papier dat u gebruikt, het maximum aantal regels per pagina en de regellengte.

In regel 10 wordt ook uitgerekend hoeveel spaties de printer moet geven voordat hij het eerste teken van een regel afdrukt, namelijk de helft van het verschil tussen LR en SB.

De regels 13 en 14 zetten een kort stukje machinetaal in het geheugen (op dezelfde plaats als de toetsafvangroutine in "Simpeltekst"). Dit wordt gebruikt om de cassetteroutine in de monitor aan te roepen, die begint op adres &H0018.

In de regels 20, 25 en 30 wordt getracht het bestand in te lezen. Er wordt gecontroleerd of het wel een Familiebestand is, en of het bestand niet te groot is. Is het laatste het geval, dan moet u met behulp van het Familiegeheugen het bestand opsplitsen, voordat u opnieuw een poging waagt het bestand af te drukken.

Regel 35 maakt een aanvang met het afdrukken. De tekst in deze regel komt op het scherm te staan en de P2000 wacht op een toetsindruk. Komt die, dan worden in regel 40 eerst de printerparameters weggePOKEd, net als in "Simpeltekst". Daarmee wordt ervoor gezorgd dat de regelteller &H60A1 op nul wordt gezet en de aantallen karakters per regel en regels per pagina op de juiste waarde worden gezet. Daarna print deze programmaregel met

grote letters "Pagina 1" op het papier, zoals in het voorbeeld, gevolgd door een streep over de hele breedte van het papier.

In de regels 40 tot en met 110 wordt telkens een stuk tekst ingelezen uit het Familiebestand. Die tekst wordt eerst keurig bemeten en gerangschikt, voordat hij wordt afgedrukt. Daarmee wordt bereikt dat er geen woorden in mootjes worden gehakt en dat er steeds een geheel aantal "schermen" op een pagina gaat. Kan een scherm er niet meer op, dan gaat het in zijn geheel naar de volgende bladzijde. Maar eerst wordt "Pagina .." afgedrukt.

Regel 110 zorgt ervoor dat op elke eerste regel "Boektitel:" en op elke tweede regel "Auteur:" wordt afgedrukt. U kunt dit naar wens vervangen door wat anders, bij voorbeeld "Muziekstuk:" en "Componist:".

Invoeren van een bestand

Het programma leest regel voor regel van het Familiebestand. Dat vraagt een zekere discipline bij het intoetsen van het bestand. Op de eerste regel moet altijd de naam van het boek staan en op de tweede regel altijd de naam van de auteur.

De witregel tussen de tweede en de derde regel zit in het af-drukprogramma. Dit betekent dat u de omschrijving moet intoetsen vanaf de derde regel op het scherm; dus geen regel overslaan. De eerste selectie, het boek van Zaks, voert u dus als volgt in met het Familiegeheugen:

Programming the Z80
Zaks, Rodney
Een overzicht van de instructies
..... en programmeervoorbeelden.

Let op: laat bij het beëindigen van de selectie (met STOP) de cursor staan na "programmeervoorbeelden". Drukt u de regelterug-toets in ("ENTER"), dan ziet het programma een extra regel en wordt tussen "programmeervoorbeelden" en de streep een extra witregel afgedrukt.

En zo ziet het miniboekenbestand afgedrukt eruit:

Pagina 1

Boektitel : Programming the Z80
Auteur : Zaks, Rodney

Een overzicht van de instructies voor de microprocessor Zilog Z80, die onder meer in de P2000 wordt gebruikt. Voorzien van illustraties en programmeervoorbeelden.

Boektitel : BASIC-Notities voor de P2000
Auteur : Kroon, Dirk

Een aantal handige programmeertips voor BASIC-programmeurs die meer uit hun P2000 willen halen

Boektitel : Het grote BASIC-probeerboek
Auteur : Boer, Chris de

Een gedegen boek voor beginnende P2000 programmeurs, met veel handige tips en nuttige informatie.

Andere bestanden

In regel 110 worden de eerste twee regels van het Familiebestand afzonderlijk ingelezen en afgedrukt. Daarna worden met "FOR J=3 TO L" de regels 3 tot en met de laatste (ten hoogste regel 35) gezamenlijk afgehandeld. Dat wil zeggen: er worden zo veel mogelijk woorden op een regel afgedrukt. Bij sommige bestanden wilt u echter meer dan twee kopregels hebben, of alleen maar kopregels zoals bij een platenbestand. Verander dan regel 110 bij voorbeeld als volgt:

```
110 LPRINTR3;R1"Muziekstuk      : "R2;R(1):LPRINTR3;R1"Compo
nist      : "R2;R(2):LPRINTR3;R1"Orkest      : "R2;R(3):LPR
INTR3;R1"Solist      : "R2;R(4):LPRINT:FORJ=5TOL:LPRINTR
3;R1;R(J):NEXT:LPRINTR3;STRING$(SB,"E")R1
```

Deze wijziging maakt dat er vier kopregels worden afgedrukt. Vanaf regel 5 ("FORJ=5TOL") worden de regels op een hoop geveegd en gezamenlijk afgedrukt.

Etiketten

Als u even wat experimenteert met het programma, zult u ontdekken dat het ongekende mogelijkheden biedt. U hebt bij voorbeeld een adressenbestand met informatie over uw vrienden en verwanten. Het bestand bestaat achtereenvolgens uit naam, straat + huisnummer, woonplaats, telefoonnummer, huisgenoten en verjaardagen. U kunt dit bestand afdrukken door in regel 110 "Muziekstuk" te veranderen in "Naam", "Componist" in "Straat" enzovoort. Als niet alles in één regel kan worden ondergebracht, splitst u regel 110 in tweeën. Regel 110 en 111 zien er dan bij voorbeeld zo uit:

```
110 LPRINTR3;R1"Naam          : "R2;R(1):L
PRINTR3;R1"Straat           : "R2;R(2):LPRINT
R3;R1"Woonplaats          : "R2;R(3):LPRINTR3;R1
"Telefoon                : "R2;R(4):LPRINTR3;R1"Huis
genoten : "R2;R(5):LPRINT:LPRINTR3;R1"Ve
rjaardagen:" :FORJ=6TOL
111 LPRINTR3;R1;R(J):NEXT:LPRINTR3;STRIN
G$(SB,"_")R1
```

Voor de aardigheid heb ik een minibestandje gemaakt met maar twee selecties. Hoe dat eruit ziet als u het afdrukt, ziet u op de volgende pagina.

Nu wilt u kerstkaarten versturen aan al uw naasten, en dus wilt u adresetiketten afdrukken. Dit betekent dat de woorden "Naam", "Straat" en dergelijke niet moeten worden afgedrukt. Ook de telefoonnummers en de verjaardagen moeten niet op de adresetiketten.

Dat is niet alles. U wilt voor de duidelijkheid tussen naam en straat, en tussen straat en woonplaats een regel wit. En u bent gebonden aan een vast aantal regels per etiket, anders krijgt u verloop. Bij de etiketten aan de rol die ik gebruik zijn dat 10 regels. Kortom: u wilt van elke selectie maar drie regels afdrukken en die moeten precies op de etiketten terecht komen.

Naam : Fam. R. Geutskens
 Straat : Amstellaan 14
 Woonplaats : Son
 Telefoon : (04990) 74290
 Huisgenoten : Rob, Roefie, Mascha, Yoeri

Verjaardagen:
 Rob: 11 juni
 Roefie: 27 april
 Mascha: 15 april
 Yoeri: 9 juni

Naam : Fam. J. Mattijssen
 Straat : Rijksstraatweg 125a
 Woonplaats : Duivendrecht
 Telefoon : (020) 990480
 Huisgenoten : Johan, Geeke, Eric, Roland

Verjaardagen:
 Johan: 15 juli
 Geeke: 16 januari
 Eric: 2 oktober
 Roland: 5 maart

U moet nu wat meer gaan veranderen. In regel 10 moeten LR, SB, PL en ER worden aangepast. In regel 40 moet wat worden gesloopt (anders begint het programma met het afdrukken van "Pagina 1") en ook regel 110 moet worden aangepast. De veranderde regels staan hieronder:

```

10 CLEAR2048,PEEK(&H6405)+PEEK(&H6406)*2
56+3072:DEFSTR:AR=35:LR=44:SB=30:PL=72:
ER=0:BA=PEEK(&H6405)+PEEK(&H6406)*256:DE
FUSR=&H61F0:R1=CHR$(27)+"F":R2=CHR$(27)+
"E":R3=SPACE$( (LR-SB)/2 ):R4=CHR$(130):R5
=CHR$(131):X=1

```

```

40 FORTE=1TOX:POKE&H60AB,LR:POKE&H60A1,0
:POKE&H60AA,PL:POKE&H60A9,0:N=1:L=1:FORI
=BA+2TOBA+PEEK(&H6034)+PEEK(&H6035)*256:
P=PEEK(I):IFP=255THENGOSUB100:GOTO70ELSE
IFP=1THENGOSUB100:R="":NEXT

```

```

110 LPRINTR3;R2;R(1):LPRINT:LPRINTR3;R2;
R(2):LPRINT:LPRINTR3;R2;R(3):FORJ=1TO4:L
PRINT:NEXT

```

Met het aldus gemodificeerde programma kunt u etiketten afdrucken. Die zien er zo uit:

	Fam. R. Geutskens	
	Amstellaan 14	
	Son	
	Fam. J. Mattijsen	
	Rijksstraatweg 125a	
	Duivendrecht	

U ziet dat alleen naam, straat en woonplaats worden afgedrukt. Als u ook vindt dat er meer PTT-beambten werkloos moeten worden en u bereid bent mee te betalen aan hun uitkering, kunt u de postcode erbij zetten.

Nog wat opmerkingen

De cassetteroutines in het programma zijn tamelijk eenvoudig. Dat heeft tot gevolg dat alleen het eerste Familiegeheugenbestand wordt ingelezen dat op de cassetteband staat. Wilt u een bestand afdrucken dat verderop op de band staat, dan moet u het eerst (met het Familiegeheugen) aan het begin van een schone cassette zetten.

De REM-regels in het programma kunt u korthedshalve weglaten. Wilt u ze erin hebben, bedenk dan dat de nette opmaak van die regels is verkregen met behulp van de tabulator. Druk dus niet per ongeluk op RETURN, want dan sluit u de regel af. Om deze reden zijn de programmaregels in dit verhaal afgedrukt met 40 karakters per regel, anders blijft er van de nette opmaak op papier niets over. Een bijkomend voordeel is dat u gemakkelijker kunt zien of u nergens iets bent vergeten.

Rob Geutskens

U kunt aan een basic programma gegevens overhandigen, (bijvoorbeeld: getallen of strings) door de opdrachten INPUT A of LINE-INPUT B\$ te gebruiken. Op deze manier kunt u echter alleen gegevens intypen die letterlijk door de computer geslikt kunnen worden. Als u gegevens kwijt wilt die eerst nog voorberekt moeten worden, moet u wat anders doen: een voorbeeld is het doorgeven van een functie. Stel u hebt een programma dat voor u een cirkel kan tekenen op het scherm; er zal dan ergens een BASIC regel zijn waarin de formule voorkomt.

```
100 DEFFN(X)=SQR(R*R-X*X)
```

Als u nu een andere functie wilt laten tekenen bijvoorbeeld:

```
100 DEFFN(X)=SIN(X)
```

dan moet u zelf regel 100 wijzigen, want hier kunt u niet de INPUT opdrachten gebruiken. De reden hiervoor is, dat het woord SIN als BASIC opdracht niet uit 3 losse letters achter elkaar bestaat, maar dat SIN afgekort wordt met een enkel codegetal (het TOKEN). (De afkorting voor SIN is &HDB; u vindt een lijst met alle TOKENS in de "blauwe map", en in de BASIC NL gebruiksaanwijzing pagina 152, en een programma in Nieuwsbrief P2000 nummer 3, pagina 5. 12). Deze wijziging voert u uit via de EDIT mode, en de EDIT routine verzorgt dan het omzetten in de gewenste TOKENS, en de lengte aanpassing als de nieuwe regel langer (of korter) is dan zijn voorganger. De P2000 springt zelfs automatisch in de EDIT stand als u de regel 90 EDIT 100 opneemt. Tot zover hebt u, ook met nieuwe functies, weinig problemen. Een kleinigheidje is dat de computer na het uitvoeren van de EDIT opdracht stopt met werken, "OK" op het scherm zet en dan rustig in de BASIC wachtstand gaat zitten wachten. Hij zit daar noodgedwongen, want hij is al zijn variabelen kwijt! Zet u maar Q=3 voor in het programma, en als u dan na het EDITEN vraagt wat Q is, krijgt u als antwoord Q=0. Deze weggooimanie kunnen we hem niet kwalijk nemen, want de variabelen staan vlak onder het BASIC programma; als u nu een extra (lange) nieuwe BASIC regel toevoegt, schuift het programma op, over de variabelen heen. Dit weggooien van de variabelen is de aanleiding tot het schrijven van dit artikel met een oplossing hiervoor.

Het eerste deel van dit artikel geeft de oplossing voor het toevoegen van één BASIC regel. In het tweede deel wordt het toevoegen van meerdere regels behandeld; dit lijkt een (Belgisch) grapje: hoe voer ik 10 nieuwe regels in? Antwoord: een voor een; maar voor een computer geldt niet, dat als hij het één keer kan, dat hij het dan ook tien keer achter elkaar kan! Het toevoegen van meerdere regels heeft (bv.) de volgende toepassing: op cassette 2 15a staat het programma "Pascal"; dit programma vertaalt uw ingevoerde Pascal programma in BASIC regels; deze regels worden in stringvorm opgeleverd. Om het zo verkregen BASIC programma te kunnen RUNNEN, moet u eerst zelf alle strings, voorzien van re-

gelnummers, intypen. U begrijpt dat een automatische regel invoer hier onmisbaar is. De methode om meerdere regels in te voeren is een uitbreiding van de methode voor een regel.

EERSTE DEEL

Typ het volgende programma in:

```
1500 LINEINPUT"Typ nu het gewenste regelnummer en de BASIC
      opdrachten";BA$(1)
1510 BA$(1)=BA$(1)+CHR$(0)
1520 Restore 1520:DATA 21,0,0,C3,79,25
1530 DATA 13,49,37,49,46,5,41,45,52
1540 FORI=0TO5:READA$:A$="&H"+A$:POKE&H6150+I,VAL(A$):NEXT:
      DEFUSR9=&H6150
1550 FORI=0TO8:READB:POKE&H6000+I,B:NEXT:POKE&H600C,9
1560 P=&H6150:Q=VARPTR(BA$(1))
1570 POKEP+1,PEEK(Q+1)-1:POKEP+2,PEEK(Q+2)
1580 A=USR9(0)
1590 PRINTCHR$(17)CHR$(15):PRINT"KLAAR"
```

Het principe is in het kort het volgende:

- A) zet de tekst voor de nieuwe regel ergens in het geheugen.
- B) geef dit opslagadres door aan de EDIT routine.
- C) de EDIT routine zorgt voor de rest.
 - n1. C1) vertalen in TOKENS.
 - C2) invoegen van de nieuwe regel tussen de andere.

Toelichting op het programma (regel voor regel)

1500 Via LINE INPUT vraagt de computer u om de nieuwe regel. U typt bijvoorbeeld in "220 REM Dit is een nieuwe regel" hierin is 220 het nieuwe, gewenste regelnummer; hierachter kunt u alle normale BASIC opdrachten intypen.

1510 Elke BASIC regel moet eindigen met een (echte) nul; deze wordt hier toegevoegd via CHR\$(0). U bereikt dit niet door bij LINEINPUT met een nul te eindigen, want deze wordt dan als een ASCII nul (=code 48) behandeld.

1520 Het machine taalprogramma: 21 00 00 LDHL,nn
 C3 79 25 JP 25 79

Het register HL moet het adres bevatten waar de nieuwe tekst staat. HL is nu nog leeg (data 0,0), maar het wordt gevuld in 1570. JP2579 is een sprong naar de EDIT routine.

1530 Na het uitvoeren van een EDIT opdracht eindigt de computer steeds in de BASIC READY stand en zet "ok" op het scherm en stopt ermee. Als u hierna echter (automatisch) door zou willen gaan, kunt u dat bereiken door van te voren de KEYBOARD buffer met een opdracht te vullen: deze opdracht luidt hier GOTO 1590. De gegeven DATA zijn de toetscodes van de aparte letters van deze opdracht; (bijvoorbeeld: toetscode voor G is 13; NB de toetscodes zijn niet dezelfde als de ASCII codes, zie BASIC NL Handleiding pagina 151).

1540 Zet de machinetaalroutine in het geheugen (6150 is een ongebruikt huishoudgedeelte).

1550 Vul de KB buffer met GOTO 1590 (afsluiten met Return=52). Zet de bufferaantalteller op 9.

1560 VARPTR(BA\$(1))= het adres waar de tekst "22OREM nieuwe regel" is opgeslagen.

1570 Vul nu de pointer naar de tekst in in de machinetaalroutine op de plaats van de 0,0 uitregel 1520. Detail: HL moet 1 plaats voor het begin van de tekst wijzen.

1580 Roep nu de EDIT routine aan: de nieuwe regel wordt nu ingevoegd. Hierna wordt (automatisch) de KB buffer bekeken en de opdracht GOTO 1590 wordt nu uitgevoerd.

1590 De opdracht GOTO 1590 wordt ook op het scherm gezet; omdat dit verder geen betekenis heeft wordt deze opdracht hier weer van het scherm verwijderd. Vanaf hier kan het programma verder vervolgd worden.

TWEEDE DEEL

Als u nu meerdere regels wilt toevoegen, en u doet dit via een herhalingslusje:

```
FORI=1TO10
VOEG 1 regel TOE met behulp van programma 1
NEXTI
```

dan merkt u, dat na de 1e EDIT ronde de I=0 is geworden en ook de 10 genummerde strings BA\$(I) (I=1,...10) bent u kwijtgeraakt. We moeten hier de computer helpen door zelf de strings BA\$(I) en de teller I ergens veilig op te bergen waar ze niet uitgepoetst kunnen worden, om ze dan weer ongeschonden tevoorschijn te halen op het moment dat de computer ze nodig heeft. Programma 2 doet dit voor u; de regelnummers en de namen van de variabelen zijn zodanig gekozen, dat u dit programma zonder meer via EDIT in uw "PASCAL" programma kunt invoegen; daarna beschikt u dan over een volautomatische regeltoevoeger. In programma 2 herkent u een groot aantal regels, die, met dezelfde regelnummers ook in programma 1 voorkwamen.

PROGRAMMA 2

```
05 CLEAR 600
1260 PRINT:PRINT"even geduld a.u.b.; vertaling van regel:"
1520 Restore 1520:DATA21,0,0,c3,79,25
1530 DATA13,49,37,49,46,5,41,45,52
1540 FORI=0TO5:ReadA$:A$="&H"+A$:POKE&H6150+I,VAL(A$):
NEXT:DEFUSR9=&H6150
1542 LB=LB-1:PRINT:FORI=1TOLB:BA$(I)=STR$(I*10)+BA$(I)+CHR$(0):PRINTBA$(I):NEXT
1545 FORI=1TOLB*3:POKE&H615F+I,PEEK(VARPTR(BA$(1))+I-1):NEXT
1548 POKE&H6160+3*LB,0:POKE&H615F,0
1550 Restore 1530:FORI=0TO8:READB:POKE&H6000+I,B: NEXT:POKE&H600C,9
1560 P=&H6150:T=&H615F
```

```

1565 Q=T+2+PEEK(T):IFPEEK(Q-1)=0THEN POKE&H600C,0: PRINT
      "KLAAR:Type RUN":END
1570 POKEP+1,PEEK(Q)-1:POKEP+2,PEEK(Q+1)
1580 PRINT USR9(0)
1590 PRINT CHR$(17)CHR$(15)CHR$(17);:POKE&H615F,PEEK(&H615F)
      +3:GOTO 1550

```

Tot slot moet u in het "pascal" programma nog de opdracht: PRINT LB*10 toevoegen in elk van de volgende regelnummers: 5470, 5660, 6270, 7460, 7730, 7980, 8040, 8240, 8330, 8360, 8620, 8710, 8900, 9020, 9080 en 9450.

TOELICHTING:

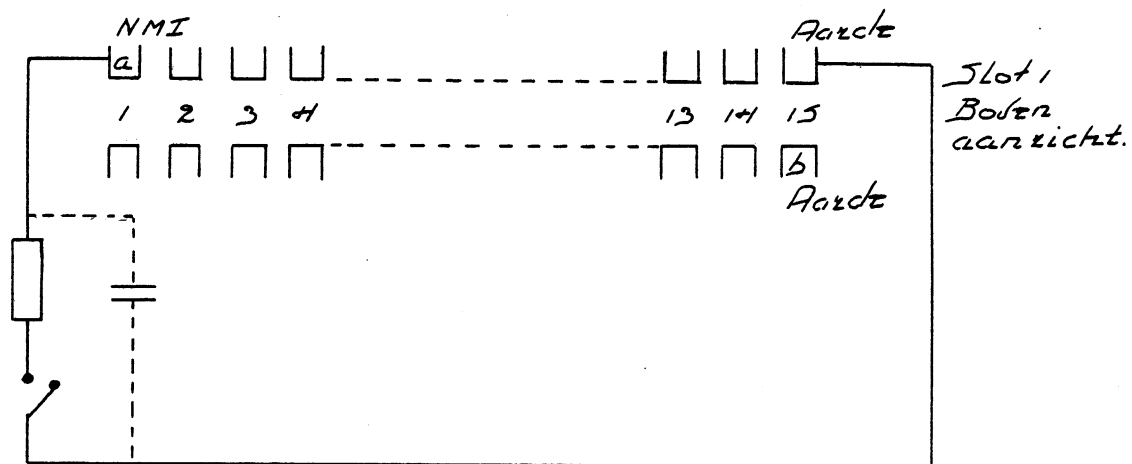
- 5 Omdat het gehele in BASIC vertaalde programma in de stringruimte moet passen, moet u hier een groot (genoeg) deel voor reserveren.
- 1260 De vertaling verloopt vrij traag; deze regel meldt u tussentijds welke BASIC regel nu aan de beurt is: de toevoeging PRINTLB*10 in de diverse "PASCAL" regels zorgt hiervoor.
- 1542 De (vertaalde) BASIC regels worden opgeleverd in stringvorm onder de naam BA\$(I); deze regels worden hier voorzien van een regelnummer en een afsluitnul.
- 1545 Nu gaan we deze strings veilig opbergen: omdat de stringteksten zelf al veilig in de stringruimte staan, behoeven we alleen nog maar de stringdescriptors te redden (deze descriptors zijn blokjes van 3 byte waar resp. de lengte en het opbergadres van de string instaan); deze descriptors zelf, vindt u op een adres aangewezen door de VARPTR(BA\$(1)). Ze worden (veilig) opgeborgen in het huishoudgedeelte achter &H6160; dit is een leeg stuk ter grootte van 160 byte zodat u 160:3=53 regels hier kunt opslaan. Als uw Pascal programma meer regels vergt, kunt u zelf een ander veilig opslagadres kiezen.
- 1548 Zet achteraan een nul neer als teken dat het einde bereikt is. Het byte in de locatie 615F gaat als teller dienst doen om het aantal BASIC regels bij te houden: zet de teller op 0.
- 1550 Vul de KB buffer (weer), met 'GOTO 1590'; het aantal letters is 9.
- 1560 T=telleradres
- 1565 Q:=de (achtereenvolgende) pointeradressen van de stringdescriptors; als de bijbehorende lengte (=1e byte) nul is (zie 1548) dan "klaar"!
- 1570 Hier worden de adressen waar de stringtekst staat "doorgegeven" aan het machinetaalprogramma (ze worden ingevuld op de plaats van de 2 nullen in 1520). N.B. het LSB adres wordt met 1 verminderd, omdat de EDIT routine verwacht dat de tekstpointer 1 plaats voor de tekst aanwijst.
- 1580 Doe nu "EDIT" d.w.z. de BASIC regel wordt nu toegevoegd. Na afloop verzorgt de KB buffer de opdracht "GOTO 1590".
- 1590 Veeg de (zinloze) tekst GOTO 1590 uit; verhoog de teller met 3 (1 stringdescriptor is 3 byte) en spring terug naar 1550 om de volgende BASIC regel in te voegen. De regelnummers

klimmen met 10 omhoog, te beginnen bij 10. In het pascal programma is invoegruijnte gereserveerd vanaf regel 10 tot 500. Hebt u meer dan 50 regels nodig dan kunt u het regelnummer increment aanpassen bijvoorbeeld: 1542
.....=STR\$ (I* 5) i.p.v. 10. Als u het pascalp rogramma (na wijziging van de DATA) opnieuw wilt RUNNEN, vergeet u dan niet de zojuist ingevoegde regels weer te verwijderen, met DELETE 10-...

PROGRAMMA IN DE KNOOP? UIT DE KNOOP!

Deze titel slaat op het volgende: uw BASIC programma kan soms zodanig op hol slaan dat u het niet meer kunt stoppen, en als de normale STOP toets niet meer werkt, blijft er maar een redding: de RESET knop (als u met machinetaal werkt zult u dit verschijnsel omgetwijfeld van zeer nabij kennen). Echter, na RESET is uw programma weg, en het gehele geheugen is, keurig gevuld met nullen, afgeleverd. Hoor ik daar iemand kreunen, dat hij juist nu "eventjes vergeten" had de laatste versie te CSAVEN?! Het "uit-poets" effect wordt veroorzaakt doordat de P2000 na een RESET automatisch op adres 0000, zichzelf herstart en direct daarna een RAM test gaat uitvoeren, waarbij het geheugen leeg wordt achtergelaten. Deze test staat in het ROM gedeelte zodat dit proces niet onderbroken of gewijzigd kan worden (alleen met een andere ROM module, b.v. een onderhoudsmodule inslot 1 is dit te omzeilen). Deze start op 0000 heet een koude start, er is echter ook nog een warm startpunt, te bereiken via adres 0066. Bij een warme start wordt de RAM test overgeslagen, er wordt alleen een eenvoudiger reinitialisatie van het BASIC uitgevoerd, met als gevolg dat uw toetsenbord weer kan reageren, en als mooiste, uw programma is ongeschonden blijven staan: u kunt het direct weer LISTEN en RUNNEN. Hoe bereikt u nu dit magische adres 0066? Nu, speciaal hiervoor is uw processor (de Z80) voorzien van een interruptlijn, de zogeheten NMI (= niet maskeerbare interrupt). Dit NMI is gewoon een van de pootjes van de Z80: als u dit pootje even kort aan de aarde legt springt de Z80 automatisch naar het adres 0066! De stand van zaken is dus de volgende:

- a) er is een NMI pootje
- b) het bijbehorende afhandelingsprogramma voor een warme start is eveneens aanwezig (vanaf 0066)
- c) er is echter geen voorziening in de P2000 aanwezig die u in staat stelt van a) en b) te profiteren. Een dergelijke voorziening is echter wel op eenvoudige manier in uw P2000 aan te brengen; benodigdheden: 2 draadjes, 1 drukschakelaar en 1 weerstand. Als u nu kijkt in Nieuwsbrief van de P2000 nummer 9, pagina 36 dan ziet u daar de connector getekend van SLOT 1 (waar de Interpreter inzit).



Op contact no. 1a is het gezochte NMI uitgevoerd. De gehele voorziening bestaat nu, uit het verbinden, via weerstand en schakelaar van contact 1a (=NMI) met 15a (=Aarde). In plaats van aan de connector te solderen kunt u beter de contactbaan nemen van de Interpreter die in dit slot 1 hoort; als deze op de normale wijze in de connector is gestoken, is nog een deel van de contactbaan zichtbaar; om meer ruimte te hebben kunt u de Interpreter openschroeven (4 kruiskoppen), dan de 2 draadjes aan de contactbaan 1 en 15 vast solderen; (als de soldeerdruppel wat groot is geworden, maakt u een klein keepje in het plastic huis) (merk op dat de contactbaan in de Interpreter verder nergens aan verbonden zit!) De draadjes voert u binnen door de interpreter naar boven toe eruit (via inkepinkje); buiten bevestigt u het drukschakelaartje eraan (het schakelaartje moet normaal open staan en slechts kort gesloten worden doordat u er even op drukt; Tandy cat nummer 275-1571 voldoet prima); een weerstand van ca. 1K Ohm beperkt de stroom; als de schakelaar denderneigingen vertoont kunt u nog een condensator (ca.1micro) over contact en weerstand heen hangen. Liefhebbers kunnen misschien zelfs een aparte schakeling maken met een "one shot" om een mooie, korte naar nul gaande puls te maken. De eenvoudige oplossing (met 1 weerstand en schakelaar) voldoet bij mij echter prima en het loont echt de moeite als je met 1 druk op de knop je verloren gewaande programma weer " uit de knoop" haalt.

Dick van den Berge
 H. Boschstrat 10
 Hazerswoude dorp
 tel 01728-8239

PHILIPS heeft nu een opgevoerde versie van de P2000T op de markt gebracht. De P2000T-102K (overigens heel eenvoudig verder op te voeren tot 104K).

Da's natuurlijk aardig, maar..... hoe ?? werkt dat nou, 6 geheugen banken ??

Wanneer je zo'n nieuw model of een opvoersset tot 102K(104K) hebt aangeschaft dan blijkt, dat je als beginnend (of gevorderd) computer niet zo heel veel kunt doen met al dat geheugen. PHILIPS levert er helaas (nog?) geen systeemsoftware bij. Jammer. Een beetje een gemiste kans. Want PHILIPS is toch maar de eerste en de enige, die de mogelijkheid van een RAM-DISK op een thuiscomputer heeft

Wat is eigenlijk een RAM-DISK.

Wel een RAM-DISK is een stuk geheugen, waarmee je kunt werken alsof het een DISK-DRIVE is, alleen het werkt vele malen sneller en..... het is nog goedkoper ook

Daarom is PHILIPS met z'n opvoersset de beste en is het jammer, dat er niet meteen een nieuwe versie van BASIC NL werd bijgeleverd. (in EPROM 4 zijn nog zo'n 300 bytes vrij)

Daarom voorlopig een stukje systeemsoftware, waarmede een begin gemaakt kan worden voor het gebruik van de RAM-DISK.

Op een eenvoudige manier kan de programmeur de RAM-DISK gebruiken voor bestandsverwerking. De software voor programma-verwerking is in voorbereiding en komt binnenkort. (Volgende nieuwsbrief)

Als eerste programma op de cassette komt dan een programma, waarmede de routine's worden ingelezen en de programma's die verder op cassette staan worden dan in de RAM-DISK ingelezen.

Daarna wordt bij CLOAD'PROGRAMMA' of RUN'PROGRAMMA' het programma niet meer op de cassette opgezocht maar direct uit de RAM-DISK gehaald. Dit maakt een ongelofelijk snelle programma wisseling mogelijk

Voorlopig echter het belangrijkste de bestandsverwerking.

Er behoeft alleen een string te worden gedefinieerd en met een tweetal POKE's en het gebruik van USER5(schrijven) en USER6(lezen) werkt de RAM-DISK.

Het gemakkelijkst is het om van een cassette 40K naar het geheugen van de RAM-DISK te schrijven en dan met het bestand in RAM-DISK te werken en na ver- of bewerking de 40K weer naar cassette te schrijven.

Bij gebruik van R.C.A. (Random Cassette Acces (r)) is dit al wel heel eenvoudig.

Definieer eerst de string (in een GOSUB routine)

```
510 C$=SPACE$(128)
520 POKE &H6070,PEEK(VARPTR(C$)+1)
530 POKE &H6071,PEEK(VARPTR(C$)+2)
540 RETURN
```

Om de gegevens van cassette naar RAM-DISK over te zetten

```
100 FOR I%=1 TO 319:GOSUB 500:X=USR3(I%):X=USR5(I%):NEXT
```

En weer terug van RAM-DISK naar cassette

200 FOR I%=1 TO 319:GOSUB 500:X=USR6(I%):X=USR2(I%):NEXT
Wil je verder inlichtingen of is iets niet geheel duidelijk dan kun
je ons bellen (s.v.p alleen 's-zaterdag's) telefoon 01804-11221

Job van Broekhuijze

PS Random Cassette Acces is de beste manier om heel eenvoudig
strings naar cassette te schrijven of van cassette te lezen.
Met Random Cassette Access denkt de P2000, dat hij een kleine
DISK-DRIVE heeft i.p.v. een cassette recorder.

```
10 *****
11 * JOB VAN BROEKHUIJZE COMPUTERSYSTEMEN Rijsingel 13, Ridderkerk *
12 *                               telefoon 01804-14354 *
13 *****
14 * routine voor het gebruik van de 6 geheugen banks (&HE000 tot &HFFFF *
15 * van de PHILIPS P2000 T&M microcomputer als RAM-DISK *
16 * copyright(c)5/1/85 J.v.Broekhuijze. Het gebruik van (delen van) deze*
17 * routine is uitsluitend toegestaan voor niet commerciële doeleinden. *
18 *****
19 * beschikbaar 6 memory-banks ad 8K-bytes start op &HE000 *
20 * onderverdeeld 6 X 64 sectors omschakelbaar &H94,n% *
21 * aantal sectors 384 sector lengte 128 bytes *
22 *****
23 * definieer een file 10 RD#=space$(128) *
24 * file opgeven aan systeem 20 POKE &H6070,PEEK(VARPTR(RD#)+1) *
25 * sectoren nummer 0 t/m 383 30 POKE &H6071,PEEK(VARPTR(RD#)+2) *
26 * LEZEN USR5(sectorna%) SCHRIJVEN USR6(sectorna%) *
27 *****
28 CLEAR ,&HD000-1,512:FOR I=&HD000 TO &HD056:READ I#:POKE I,VAL("&H"+I#):NEXT
29 DEF USR5=&HD047:DEF USR6=&HD03C:' schrijven USR5(n%) lezen USR6(n%)
30 R#=SPACE$(255):POKE &H6070,PEEK(VARPTR(R#)+1):POKE &H6071,PEEK(VARPTR(R#)+2)
31 RETURN:
32 DATA 23: 'D000 INC HL ADRES1:
33 DATA 7E: 'D001 LD A.(HL) :
34 DATA FE,02: 'D002 CP 02 max 384 sectors :
35 DATA F0: 'D004 RET P te veel dan terug :
36 DATA 1E,06: 'D005 LD E.06 max banks + 1 :
37 DATA 32,85,60: 'D007 LD (6085).A :
38 DATA FE,00: 'D00A CP 00 :
39 DATA 28,02: 'D00C JR Z ADRES2 :
40 DATA 1E,02: 'D00E LD E.02 rest banks + 1 :
41 DATA 2B: 'D010 DEC HL ADRES2:
42 DATA 7E: 'D011 LD A.(HL) :
43 DATA 32,84,60: 'D012 LD (6084).A :
44 DATA 16,40: 'D015 LD D.40 nbr sect/bank in E :
45 DATA 1D: 'D017 DEC E ADRES3:
46 DATA FE,40: 'D018 CP 40 :
47 DATA 38,03: 'D01A JR C ADRES4 :
48 DATA 92: 'D01C SUB D :
49 DATA 18,F8: 'D01D JR ADRES3 :
50 DATA 32,84,60: 'D01F LD (6084).A sectornr 6084 ADRES4:
51 DATA 7B: 'D022 LD A.E number of sectors/bank:
52 DATA 32,85,60: 'D023 LD (6085).A banknumber 6085 :
53 DATA D3,94: 'D026 OUT 94 switch bank :
54 DATA 3A,84,60: 'D028 LD A.(6084) sectornbr in A ADRES5:
55 DATA 3C: 'D02B INC A :
56 DATA 21,00,E0: 'D02C LD HL.E000 start memorybank :
```

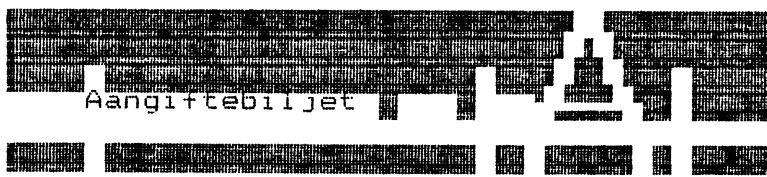
```

57 DATA 01,80,00:      'D02F LD BC.0080          record 128 bytes      :
58 DATA 3D:           'D032 DEC A              :
59 DATA FE,00:        'D033 CP 00              :
60 DATA 28,03:        'D035 JR Z END   find sector :
61 DATA 09:           'D037 ADD HL.BC sector not found  :
62 DATA 18,F8:        'D038 JR ADRES6 try again      :
63 DATA C9:           'D03A RET                          ADRES7:
64 DATA 00:           'D03B NOP                          :
65 DATA CD,00,D0:     'D03C CALL ZOEK                          LEZEN :
66 DATA ED,5B,70,60: 'D03F LD DE.(6070)                          :
67 DATA ED,B0:        'D043 LDIR   BC 128 bytes DE string HL sector :
68 DATA 18,0A:        'D045 JR D051                          :
69 DATA CD,00,D0:     'D047 CALL ZOEK                          SCHRIJVEN :
70 DATA ED,5B,70,60: 'D04A LD DE.(6070)                          :
71 DATA EB:           'D04E EX DE.HL                          :
72 DATA ED,B0:        'D04F LDIR   BC 128 bytes DE sector HL string :
73 DATA AF:           'D051 XOR A                          :
74 DATA D3,94:        'D052 OUT 94                          :
75 DATA C9,00,00:     'D054 RET                          :
76 :::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
77 FOR IX=0 TO 383:GOSUB 30:MID$(R$,1)="RAMDISK sector nummer "+STR$(IX)
78 X=USR5(IX):PRINT USING"____";IX;:NEXT:END:'          voorbeeld schrijven
79 FOR IX=0 TO 383: GOSUB 30:X=USR6(IX):PRINT R$:NEXT:END:'voorbeeld lezen

```

Philips 51581102a Oc

B e l a s t i n g



Aangiftebiljet

1 9 8 4

Toets #

Philips 51581102b Oc
 B e l a s t i n g 1 9 8 4

NEUW in Viditel

Dit programma kan U helpen bij het invullen van het aangiftebiljet model A. Reeds ingevoerde gegevens kunnen samen met het programma worden opgenomen op een minicassette. Indien een printer is aangesloten kunt U ze ook uitprinten.

Wel dient U er zelf voor te zorgen dat U de correcte gegevens invoert. Raadpleeg hiervoor de toelichting die U bij het aangiftebiljet krijgt.

Dit programma bestaat uit 3 delen #

Versie N2.4 Totaal 28 blokken 12 feb 85
 minimaal 32K geheugen nodig
 Totaal 114 pagina's Totaal f 11.40

8 vorige 0 overzicht 9 vorige

Het aantal computerboeken is niet meer te tellen, maar het aantal boeken dat zich speciaal tot de P2000-gebruiker richt is op de vingers van één hand te tellen. Blijde verrassing dus toen Nijgh & Van Ditmar Educatief ons het boekje Informatica 2000 toestuurde, geschreven door J. Acda en L. Koning en met als ondertitel "Programmeren en bedienen voor het lbo/mavo (en (k)mbo) op de P2000".

Deze kromme zin is, helaas, kenmerkend voor het boekje, dat vol slordigheden staat. Er worden instructies gebruikt die pas pagina's verder worden behandeld. Op pag. 26 wordt bij voorbeeld de puntkomma gebruikt om te voorkomen dat de cursor naar de volgende regel gaat en pas op pag. 36 is dit uitgelegd. Op dezelfde bladzijde: "Met het GOTO-statement gaat de computer terug (sprongopdracht)." Kan de P2000 niet vooruit springen? Overigens heb ik nergens een verklaring van het woord "statement" gevonden, maar dat kan aan mij liggen.

Op pag. 43: 100 PRINT "WIL JE NOG LANGER DOORGAAN?" (j/n). Als ik dat probeer krijg ik "Division by zero in 100" op het scherm, maar ik heb dan ook een heel oude P2000. Waarom worden trouwens alle strings met hoofdletters ingetikt, behalve dan dat (j/n), terwijl BASIC-NL met kleine letters begint? De arme beginner moet dan de hoofdletter-vastzettoets indrukken en dan niet vergeten terug te schakelen als er cijfers in de tekst voorkomen, anders komt er "!£\$%" te staan en dat ziet eruit als een vloek. Of met SHIFT + TAB overschakelen, maar dat staat nergens.

Door het hele boekje heen kreeg ik trouwens de indruk dat het voor een andere computer is geschreven, en alleen maar hier en daar is aangepast aan de hebbeligheden van de P2000. Op pag. 45 bij voorbeeld: "Alleen als het 'geSAVED' hebben op een bandje, een ponskaart of een schijf, kunnen we met een LOAD-commando". Over het vergeten van "we" in deze zin zullen we maar niet vallen. Wel over het feit dat je met de P2000, zoals die in het onderwijs wordt gebruikt, geen programma's op ponskaarten kunt bewaren. Dat kan wel op een "bandje", maar dan moet je CSAVE gebruiken.

Pag. 46: "Het geheugen herinnert zich dus niets, omdat het tot geen enkele activiteit in staat is." Pag. 49, over de BASIC-interpretor: "Dit programma leest het aangeboden programma, maar voert het niet uit! (Het wordt daarom Read Only Memory genoemd, afgekort ROM.)" Op pag. 70 een afbeelding van een transistor met het bijschrift "weerstand". Dat zijn geen slordigheden meer, dat is onwetendheid.

Op pag. 73 verschijnt het hekje ten tonele. Althans: dat was kennelijk de bedoeling, want het hele boekje door wordt consequent (dat wel) PRINT USING "****.*" gebruikt. Sterretjes in plaats van hekjes dus.

Vanaf bladzijde 76, halverwege het boekje, gaat het plotseling niet meer over de P2000 maar over treinkaartjes, persoonlijke leningen, hypotheeken, valuta's, ongehuwde moeders (met 1 kind en recht op kinderbijslag), wegenbelasting, de school, acceptgirokaarten en Philips Telecommunicatie en Informatica B.V.

Kortom: een slordig en onbekwaam in elkaar geknutseld boekje. En warrig bovendien. Tekstjes over het tweetalig stelsel gaan zonder waarschuwing over in FOR...NEXT-lussen. Een verhaaltje over "de geschiedenis van de computer", waar iedereen die iets van computers weet moeiteloos meer gaten in schiet dan er in Emmenthaler zitten, gaat uit als een nachtkaars. Een onexact epistel over de opbouw van computersystemen, waarover de auteurs zeer persoonlijke opvattingen blijken te hebben, voert de leerling tot de sprongopdrachten.

Zouden de auteurs geen onderscheid weten te maken tussen het programmeren van een computer in BASIC, het gebruik van een computer dat een uitgeleerde (k)mbo-student in de praktijk ontmoet, de opbouw en de werking van een computer en de geschiedenis van dat apparaat. Dat lijken me vier verschillende zaken, die je niet te veel door elkaar moet haspelen.

Niet kopen dus dat boekje, als u het vermijden kunt.

Rob Geutskens