

VIC-PRIMEURS

JAARGANG 2

No. 2



- * **De functie-toetsen**
- * **Hoge resolutie (deel 2)**
- * **Programma-listings**
- * **VIC-FORTH**
- * **BASIC-assembler**

INHOUD

1. AAN DE LEZER
2. HET SGRAANKBORD
3. VRAAGT U MAAR
4. GREGORIAANSE KALANDER
5. PROGRAMMABEURS
6. NIEUWE ROMCASSETTE
7. MUZIEK
8. VICKIES
9. GEBRUIKERSCLUB
10. PENNY LANE
11. VIC-REVEALED
12. FORTH
13. VICTIP
14. HOGE-RESOLUTIE
15. 3K , 8K , 16K RAM gebruiksaanwijzingen
16. VICKIES
17. VICTIP
18. PRIJSLIJST
19. DEALERLIJST

DE VOLKSWAGEN WERD OOK EERST UITGELACHEN



En de VolksComputer? Dat zal de VIC-20 Volks-Computer niet overkomen! De VIC-20 is dé eerste volwaardige micro waarmee je het 'computeren'

AAN DE LEZER

Met de derde editie van VIC-primeurs introduceren we opnieuw een aantal nieuwe rubrieken. Belangrijke ontwikkelingen binnen COMMODORE worden met dit nummer aangekondigd. Opvallend zijn alle reakties , die we zowel van VIC-eigenaren als van dealers voor deze editie hebben mogen ontvangen.

Voor alle nieuwkomers nog even dit.

VIC-primeurs is een tweemaandelijks tijdschrift en wordt speciaal samengesteld voor de eigenaren van de VIC-20 computer. Het heeft ook tot doel de VIC gebruikers clubs , die in het land worden opgezet , te ondersteunen. In VIC-primeurs worden allerlei zaken over de VIC-20 gepubliceerd , in de vorm van artikelen door deskundigen , informatie van en over dealers en natuurlijk niet in de laatste plaats de bijdragen van de lezers zelf.

adres voor inzendingen is :
Aan de VIC-gebruikersclub
Postbus 12972
1100 AZ AMSTERDAM

VIC-primeurs geeft ook informatie uit het buitenland. Deze informatie wordt vergaard uit alle inzendingen , die wij van alle COMMODORE bedrijven (ruim twee honderd over de hele wereld) , ontvangen. Tevens zullen de lezers exclusieve aanbiedingen van nieuws , literatuur en software voor de VIC-20 uit andere landen krijgen.

In nederland zijn al veel dealers en VIC gebruikers bezig om hard- en software voor de VIC te ontwikkelen. In een aantal rubrieken zal hier uitgebreid aandacht aan worden besteed. De stroom van informatie uit de afgelopen weken is in deze editie verwerkt en daarvoor onze hartelijke dank. Misschien denk je dat jouw ontwikkelingen of programma niet de moeite waard zijn voor de PRIMEURS. Stuur ze toch in . Want veel lezers zijn beginners en van elkaar kunnen we veel leren , maar vooral goede ideeën opdoen. Dus graag , alle inzendingen zijn welkom.

Wederom zouden we VIC eigenaren willen aansporen om lid te worden van de VIC-gebruikersclub. VIC-primeurs , het tijdschrift van de club , is echt onmisbaar als u alles uit de VIC-20 wilt halen , wat eruit te halen valt.Bij HANDIC ontvangt men veel vragen , die reeds in de PRIMEURS zijn beantwoord. Hierdoor ontdekken ook veel nieuwe beginners VIC-primeurs. Wilt u dus meepraten met de andere 2000 leden wordt dan lid en ontvang elke twee maanden VIC-primeurs door overmaking van Fl. 50,- met behulp van de acceptgirokaart , die u in deze editie aantreft. Vergeet niet daarop uw naam , adres , woonplaats en postcode te vermelden.

U kunt ook lid worden door Fl. 50,- over te maken op Postgiro 2930100 , t.n.v. HANDIC BENELUX BV.
Postbus 213
1850 AE HEILOO
onder vermelding van VICCLUB lidmaatschap en uw naam , adres , postcode en woonplaats.

Veel plezier met deze primeurs en tot volgende maand. HOI.

Het SGRAANKBORD

Onder deze titel (de programmanaam van een de inzenders) de uitslag van de programma prijsvraag van de vorige 'VIC PRIMEURS'. De belangstelling was enorm. Er bleven nieuwe oplossingen van het schaakbord-probleem binnen komen. Ook bijzonder leuk was het enthousiasme waarmee de meeste inzenders aan de slag zijn gegaan en de opgave hebben opgelost. Dat programmeurs meer in hun mars hebben dan alleen maar het schrijven van droge computerprogramma's, maar ook een bijzonder gevoel voor humor hebben, mag blijken uit een drietal citaten uit de ingezonden programma's. Enkele oplossingen waren zelfs van geluidseffecten voorzien, 1 oplossing speelde zelfs complete melodietjes.

Dit alles maakte het kiezen van de winnende oplossing, uit de ruim 25 goede oplossingen, natuurlijk erg moeilijk.

De enigste inzender die alleen een oplossing voor het 'makkelijke gedeelte' inzond was Robert Landaal uit Beverwijk, hij ontvangt van ons als TR00STPRIJS het "informatieboek voor VIC-programmeurs".

De HOOFPRIJS is gewonnen door dhr. Ernst uit Waalwijk. Hij zal de cassette 'PROGRAMMERS AID' ontvangen. Zijn programma is zo uitgebreid dat het helaas niet mogelijk is het in 'VIC PRIMEURS' af te drukken, vandaar een foto van een van de afbeeldingen in zijn programma.

Van alle andere inzendingen heeft loting moeten beslissen wie in aanmerking zou komen voor de TWEEDE PRIJS, een ROM-spelletje. De gelukkige is geworden dhr. Kuypers uit Delft. Zijn programma staat aan het einde van deze uitslag vermeld.

Wegens tijdgebrek komt de prijsvraag in dit nummer van 'VIC PRIMEURS' helaas te vervallen.

CitatEN :

"De keizer, vwoed schaker als hij was, raakte direct enthousiast en bood prompt

1.84467441E+19

graankorrels aan.

Onze bescheiden koopman speelde hierop in met de historische woorden:

Laten we het eerlijk houden, en schenkt u mij er
18446744073709551615 "

(H. Bierman, Heeskerk)

"Een speld in een hooiberg zoeken is kinderspel vergeleken bij een zoek aktie om het schaakbord onder de graanberg terug te vinden."

(M. Hielkema, Stadskanaal)

"De keizer was verrukt over de vindingrijkheid van zijn onderdaan en bood hem 2 ministersposten aan, landbouw (vanwege het graan) en financien (vanwege het rekenen).

De grootste beloning voor de koopman kwam echter pas veel later Zijn idee werd in 1982 onderwerp van een programma-prijsvraag voor VIC-20 bezitters."

(A. Ernst, Waalwijk)

Oplossing van dhr. Kuijpers :

```
100 DIM X(2)
110 X(0)=1
120 FOR T=0 TO 63
130 :C=0
140 :FOR I=0 TO NX
150 ::W=2*X(I)+C
160 ::C=0
170 ::IF W>=1E9 THEN C=1:W=W-1E9
180 ::X(I)=W
190 :NEXT
200 :IF C=1 THEN NX=NX+1 :X(NX)=1
210 NEXT
220 X(0)=X(0)-1
230 PRINT STR$(X(NX));
240 FOR I=NX-1 TO 0 STEP -1
250 :X$=STR$(X(I))
260 :X$=RIGHT$(X$,LEN(X$)-1)
270 :PRINT RIGHT$("00000000"+X$,9);
280 NEXT
```

TOTAAL

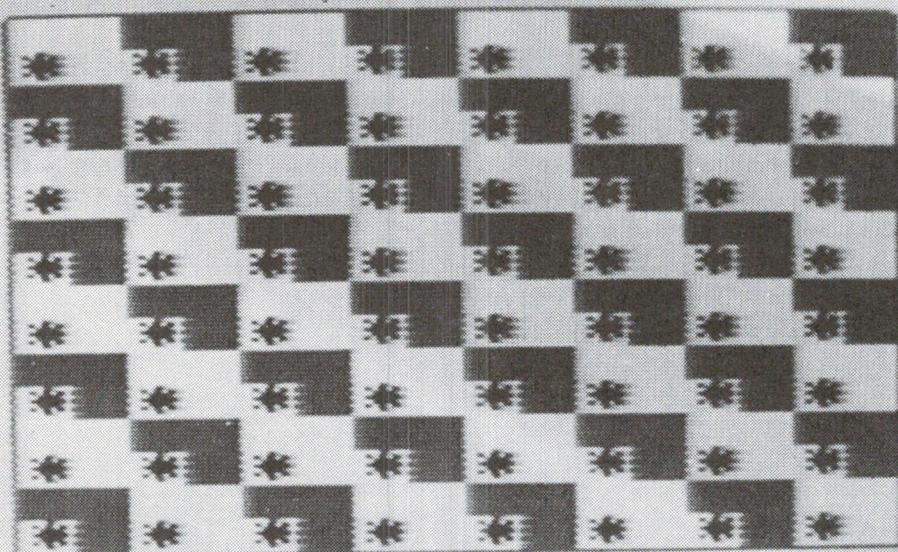


FOTO VAN DE WINNENDE OPLOSSING
VAN DE PRIJSVRAAG !

VRAAGT_U_MAAR_!

Op deze bladzijden worden vragen behandeld die door de lezers gesteld zijn. De vragen mogen betrekking hebben op alles rond de VIC-20, inclusief problemen met zelf gemaakte programma's voor de VIC-20. Het lijkt ons echter verstandig om vragen naar bijvoorbeeld de reparatie duur van een apparaat rechtstreeks aan uw dealer of de importeur te stellen.
Vragen kunnen worden opgestuurd aan:

VIC-gebruikersclub
Postbus 12972
1100 AZ Amsterdam

Graag een aan uzelf geadresseerde envelop met postzegel erop bijsluiten. Het is ook handig als u uw telefoonnummer even opgeeft.

Wij ontvingen o.a. de volgende vragen:

1.

De klok (TI\$) wordt verstoord door I/O op de cassettereorder. Hoe komt dit en is het wat aan te doen ?

Het is inderdaad zo dat het openen en sluiten van bestanden o cassette de klok beïnvloedt. Dit is waarschijnlijk een fout in het operating system van de VIC-20. Tot nog toe is ons niet bekend of daar iets aan te doen is.

2.

Zijn er al programmeerboeken uit. Zo ja, welke? Hoe duur? Waar te verkrijgen?

-'Informatieboek voor programmeurs VIC-20',

adviesprijs F 59,-

-'Learn computer programming with the COMMODORE VIC',
adviesprijs F 21,-

Beide boeken zijn bij de VIC-20 dealers verkrijgbaar.

3.

Kunt u mij informeren hoe de TAB-functie voor de printer in een programma verwerkt moet worden.

Voorbeeld: 20

PRINT#4,CHR\$(16)"08 HALLO"

Deze instructie laat de printer 'HALLO' printen beginnend op de 9e positie. CHR\$(16) is de aanduiding voor de TAB-functie, die overigens in de handleiding de POS-code wordt genoemd. 08 duidt de positie aan waar moet worden begonnen met printen. De posities tellen vanaf 00. De beginpositie moet altijd in twee cijfers direct volgend op CHR\$(16) opgegeven worden. Zie verder de handleiding van de printer, blz. 26/27 'Print position determinatie'.

4.

Waar vind ik een duidelijke omschrijving van de functietoetsen F1 t/m F8?

De karaktercode's van de functie-toetsen vindt u op blz. 146 van de VIC-20 handleiding. Het gebruik hiervan wordt behandeld in een VIC-TIP elcers in de nummer van 'VIC PRIMEURS'.

5.

Is het mogelijk om met 16K uitbreidings 2 programma's met oplopende nummering te koppelen?

Met een 16K uitbreiding is dat niet zondermeer mogelijk. In de volgende VIC PRIMEURS zullen we daarvoor een hulpprogramma beschrijven (wat overigens met elke geheugen grote zal werken).

Met de 'VIC-1212 programmers aid' als uitbreiding is dit wel mogelijk namelijk met de opdracht MERGE.

6. Mij is nog niet duidelijk hoe POKE-instrukties moeten worden gewijzigd wanneer de uitbreidingscassettes worden toegepast?

Bij de 3K-uitbreiding blijven de POKE-adressen ongewijzigd. Bij de 8K- en 16K-uitbreiding liggen de tekens van het scherm niet meer op adres 7680 t/m 8191 maar op adres 4096 t/m 4607. De kleuren van de tekens op het scherm verhuizen van 38400 t/m 38911 naar 37888 t/m 38244.

7. Kunt u mij uitleggen wat er bedoeld wordt met 'garbage collecting problem' in VIC-computing blz. 15?

De VIC heeft geheugenruimte nodig voor variabelen. Sommige variabelen zijn tijdelijk binnen een programma, of wijzigen van lengte etc. Op den duur zullen

de variabelen dus verspreid staan in het geheugen met lege plekken ertussen. Als de computer nu, terwijl hij een programma uitvoert, een groot stuk aaneengesloten geheugen nodig heeft, dat niet beschikbaar is, gaat hij variabelen in het geheugen opnieuw rangschikken totdat de vrije ruimte aaneengesloten is. Dit proces heet 'GARBAGE COLLECTING' en het probleem is dat dit soms erg lang kan duren (tot enkele minuten toe). Dit 'opruimen' wordt trouwens ook uitgevoerd bij de FRE(0)-instructie.

8. Is er iets van bekend of er een damspel, een schaakspel of een bridgespel voor de VIC uitkomt?

Van een damspel is ons nog niets bekend. Aan een schaakprogramma wordt naar ons weten gewerkt, het zal een programma in ROM-cassette worden. De volgende firma verkoopt een bridgespel:

Electronica 2000
Afd. MICROSOLVE
Chrysantenstraat 4
1031 HT Amsterdam
020-360901

Gregoriaanse-kalander

Wij ontvingen de volgende brief van een onze leden en daar de inhoud van deze brief ons voor iedereen interessant lijkt hebben we hem geheel in 'VIC PRIMEURS' opgenomen. Met natuurlijk een woord van dank voor de Putters voor zijn inzending.

Geachte redactie,

Sinds enige weken ben ik de entousiaste bezitter van een VIC-20 en ben na de bestudering van de handleiding al direct begonnen met het maken van programma's. Hoewel het mijn bedoeling is (als penningmeester van een vereniging) de computer hoofdzakelijk te gebruiken voor administratieve doeleinden, waarvoor de VIC-20 zich uitermate goed leent, is het altijd leuk om wat bijzonders te hebben.

Aangezien er al voldoende kleur- en animatieprogramma's in omloop of zelf te maken zijn wil ik voor de liefhebbers een tip geven voor het maken van een (in geprogrammeerde rekenmachines vaak toegepast) programma voor het berekenen van de dag van de week (ook wel eeuwigdurende kalender genoemd) waarvoor al sinds lang een formule bekend is, die echter voor normale toepassingen zo tijdervarend is dat hij alleen in rekenmachines en computers kan worden gebruikt.

Het programma bepaalt de dag van de week sinds 1582, dus gebaseerd op de Gregoriaanse kalender.

Ik geef hier alleen de essentiële programmapunten, die door elke programmeur van lijnnummers en verdere verfraaiing kan worden voorzien.

De input is D,M,J dwz bijv. 5,12,1983

Het antwoord luidt; 5 12 1983 is een maandag (dat weet u dan ook vast). De betrokken formules werken als volgt. Eerst wordt de FAKTOR van iedere datum berekend. Met behulp van deze FAKTOR wordt de juiste dag berekend. Het te maken programma is klein en altijd leuk als tussendoortje.

Voor januari en februari is de FAKTOR-berekening verschillend van die van maart t/m december. Schrikkeljaren worden automatisch meeberkend. U weet het wellicht: eeuwjaren zijn geen schrikkeljaren tenzij ze bij deling door 400 een heel getal opleveren. 1600 en 2000 zijn dus wel schrikkeljaren (u ziet 'VIC PRIMEURS' is ook goed voor u algemene opvoeding, red.)

Nu het algoritme:

```
INPUT D,M,J : REM DUS DAG,MAAND,JAAR
IF M<3 THEN F=365*j+d+31*(M-1)+INT((J-1)/4)
IF M<3 THEN F=F-INT(.75*(INT(((J-1)/100)+1)))
IF M>2 THEN F=365*j+d+31*(M-1)-INT(.4*M+2.3)
IF M>2 THEN F=F+INT(J/4)-INT(.75*(INT(J/100)+1))
W=F-INT(F/7)*7
IF W=0 THEN PRINT"ZATERDAG"(of D$="ZATERDAG")
enz.
enz.
IF W=6 THEN PRINT"VRIJDAG"
END(of GOTO)
```

N.B.

Ik heb voor de zekerheid de formule steeds in 2 delen gesplitst, omdat ik bang was dat hij voor een regel te complex zou worden, maar misschien is dat helemaal niet zo.

Denk bij het intypen aan de juiste aantalen haakjes. Er moeten altijd evenveel "(" haakjes als ")" haakjes zijn.

Nog een aardige tip is misschien het maken van een klok op de televisie (zoals beschreven in de handleiding op blz. 114), als u dit combineert met het slaan van de uren en bovendien het geven van blieps (1,2 of 3) op ieder 1e, 2e resp. 3e kwartier.

Hoogachtend

H. Putters
Amersfoort

PROGRAMMABEURS

=====

Via het ruilen van programma kunt u een grote programma bibliotheek opbouwen , maar waarschijnlijk veel belangrijker nieuwe programmeervormen ontdekken.

VIC-primeurs wil iedereen daarbij behulpzaam zijn , vandaar deze programma(ruil)beurs.U hanteert de navolgende procedure.

Als u een programma ontwikkelt hebt , waarmee u denkt de lezers van Primeurs een plezier te doen , stuur het dan op naar:

De VIC-gebruikersclub

Postbus 12972

1100 AZ AMSTERDAM

Denk nu niet dat jouw programma niet goed genoeg is , want tenslotte zijn we bijna allemaal amateurs en willen we alleen maar van elkaar leren.

Nadat het programma is ingestuurd komt het terecht op de ruillijst in de volgende editie van VIC-primeurs.

U kunt dan een programma uit de ruillijst van de laatste Primeurs kiezen.Dit programma wordt u dan toegezonden op een nieuwe cassette

Om deze procedure goed te laten verlopen en mogelijk misbruik tegen te gaan zijn de volgende regels van kracht:

1. Het programma moet op een cassette ingestuurd worden.

2. Het programma moet origineel zijn en niet reeds in een identieke vorm op de ruillijst staan

3. De volgende gegevens moeten bij de inzending worden vermeld:

- NAAM , ADRES , WOONPLAATS , TELEFOON
- SERIENUMMER van uw VIC-20
- De titel van uw programma
- een daarbij behorende omschrijving(kort en simpel)
- De naam van het programma , waarmee u wilt ruilen

4. Bijsluiting van Fl.5,- voor portokosten

5. Lidmaatschap van de gebruikersclub

6. Het programma moet in het nederland zijn . Het mag dus geen vertaalwerk zijn uit het buitenland.

Deze regel zijn opgesteld om ieder produkt toch zoveel als mogelijk te beschermen.Velen maken al gebruik van deze ruilmogelijkheid en uit de reakties mogen we aannemen naar tevredenheid.

Je zult er veel plezier aan beleven.

UIT_DE_RUILLIJST:_SOLITAIR

De ruillijst is uitgebreid met enige inzendingen. We geven hieronder de listing van een van deze inzendingen, nl. SOLITAIR, van dhr. Henk Bierman. Deze inzending is eruit gelicht omdat het spel SOLITAIR via dit programma aangenaam is te spelen en geeft indien gewenst de juiste oplossing. De maker kan een beloning in de vorm van een ROM-spel tegemoet zien.

We hadden wel een tweetal opmerkingen bij het programma:

- er zit een geluidsroutine in die niet wordt gebruikt (regel 4000 ev.)
- een iets duidelijker aanwijzing voor het gebruik van de toetsen en het zichtbaar maken van wat je ingetypt zou zeker wenselijk zijn.

```
5 GOSUB7000
10 DIMAX(7,7)
20 FORI=1TO7:FORJ=1TO7:AX(I,J)=1:NEXT:NEXT
150 AX(4,4)=0
151 TI$="000000"
155 GOSUB500
160 GETI$
161 IFI$="S"THEN900
162 IFI$="H"THEN6000
165 IFVAL(I$)<10RVAL(I$)>7THEN160
170 GETJ$: IFVAL(J$)<10RVAL(J$)>7THEN170
180 I=VAL(I$):J=VAL(J$)
190 X=I:Y=J
200 GOSUB700
210 IFFLAG=1THEN300
220 GETT$: IFT$<>"J"ANDT$<>"H"ANDT$<>"I"ANDT$<>"N"THEN220
230 IFT$="J"THENGOSUB1000
240 IFT$="H"THENGOSUB1200
250 IFT$="I"THENGOSUB1400
260 IFT$="N"THENGOSUB1600
270 GOTO160
300 FLAG=0
310 PRINT"NOG EENS PROBEREN !"
315 FORP=1TO999:NEXT
317 PRINT"J" :PRINT"J"
320 GOTO160
500 PRINT"J"
505 PRINT" 1234567":PRINT
510 PRINT" 1 ○●●●○"
520 PRINT" 2 ○●●●○"
530 PRINT" 3 ●●●●●●●"
540 PRINT" 4 ●●●●●●●"
550 PRINT" 5 ●●●●●●●"
560 PRINT" 6 ○●●●○"
570 PRINT" 7 ○●●●○"
```

```

580 PRINT
595 PRINT:PRINT:PRINT
600 RETURN
700 IFX<10RX>70RY<10RY>7THEN780
710 IFX<3ANDY<3THEN780
720 IFX<3ANDY>5THEN780
730 IFX>5ANDY<3THEN780
740 IFX>5ANDY>5THEN780
750 GOTO790
780 FLAG=1
790 RETURN
900 PRINT"U GEBRUIKTE"TELLER;"BEURTEN"
910 PRINT:PRINT"GEBRUIKTE TIJD:";TI$
960 END
1000 Q=I-2
1010 R=I-1
1030 X=I:Y=Q
1040 GOSUB700
1050 IF FLAG=1THEN1080
1055 IFAX(Q,J)=10RAZ(R,J)=0THEN1080
1060 GOSUB1800
1070 GOTO1100
1080 FLAG=0
1090 PRINT"ONJUISTE GETALLEN !"
1093 FORP=1TO999:NEXT
1095 PRINT"J . . ."J"
1100 RETURN
1200 Q=I+2
1210 R=I+1
1230 X=I:Y=Q
1240 GOSUB700
1250 IF FLAG=1THEN1280
1255 IFAX(Q,J)=10RAZ(R,J)=0THEN1280
1260 GOSUB1800
1270 GOTO1300
1280 FLAG=0
1290 PRINT"ONJUISTE GETALLEN !"
1293 FORP=1TO999:NEXT
1295 PRINT"J . . ."J"
1300 RETURN
1400 Q=J-2
1410 R=J-1
1430 X=I:Y=R
1440 GOSUB700
1450 IF FLAG=1THEN1480
1455 IFAX(I,Q)=10RAZ(I,R)=0THEN1480
1460 GOSUB2000
1470 GOTO1500
1480 FLAG=0
1490 PRINT"ONJUISTE GETALLEN !"
1493 FORP=1TO999:NEXT
1495 PRINT"J . . ."J"
1500 RETURN
1600 Q=J+2
1610 R=J+1
1630 X=I:Y=Q
1640 GOSUB700
1650 IF FLAG=1THEN1680
1655 IFAX(I,Q)=10RAZ(I,R)=0THEN1480

```

```
1660 GOSUB2000
1670 GOTO1700
1680 FLAG=0
1690 PRINT"ONJUISTE GETALLEN !"
1693 FORP=1TO999:NEXT
1695 PRINT"J"                J"
1700 RETURN
1800 AX(I,J)=0
1810 AX(Q,J)=1
1820 AX(R,J)=0
1830 GOSUB2200
1840 RETURN
2000 AX(I,J)=0
2010 AX(I,Q)=1
2020 AX(I,R)=0
2030 GOSUB2200
2040 RETURN
2200 REM SCAN
2210 FORV=1TO7
2215 N=38451+22*V
2220 FORW=1TO7
2230 IFAX(V,W)=1THENPOKEW+N,0
2240 IFAX(V,W)=0THENPOKEW+N,1
2250 NEXT:NEXT
2256 TELLER=TELLER+1
2260 RETURN
2900 GOSUB10
3000 FORI=1TO7
3010 FORJ=1TO7
3020 PRINTAX(I,J);
3030 NEXT
3040 PRINT
3050 NEXT
3100 RETURN
4000 POKE36878,15
4005 FORI=1TO999:NEXT
4010 POKE36876,220
4020 FORL=1TO5:NEXTL
4030 POKE36876,0
4040 FORL=1TO5:NEXTL
4050 POKE36876,200
4060 FORL=1TO5:NEXTL
4070 POKE36876,0
4080 FORL=1TO500:NEXTL
4090 POKE36878,0
4100 END
5000 POKE36877,220
5010 FORL=15TO0STEP-1
5020 POKE36878,L
5030 FORM=1TO50
5040 NEXTM
5050 NEXTL
5060 POKE36877,0:POKE36878,0
5070 RETURN
6000 READI,J,T
6010 ONTGOOSUB1000,1200,1400,1600
6050 IFT=0THEN6100
6060 GOTO6000
```

```

6070 DATA6,4,1,5,6,3,7,5,1,7,3,4,4,5,2,7,5,1,2,5,2,3,7,3,5,7,1
6080 DATA3,4,4,3,7,3,3,2,4,1,3,2,1,5,3,4,3,1,1,3,2,6,3,1,5,1,4
6090 DATA3,1,2,5,4,3,5,1,4,3,4,4,3,6,2,5,6,3,5,4,3,5,2,1,3,2,4
6095 DATA4,4,3,2,4,2,4,5,3,4,2,4,///
6100 PRINT"ZO MOET HET NUU !!!"
6105 PRINT
6110 GOT0900
7000 PRINT"J";CHR$(14)
7005 PRINT"•TRAKS VERSCHIJNT OP UW BEELD EEN FIGUUR MET BALLETJES."
7007 PRINT
7010 PRINT" IET IS UW OPDRACHT OM IN ZOVEEL MOGELIJK BEURTEN OVER ANDERE"
7022 PRINT"BALLETJES TE SPRINGEN."
7025 PRINT
7030 PRINT"•EDERE KEER VERDIJNT DAARBIJ HET BALLETJE WAAR U OVERHEEN GAAT."
7035 PRINT:PRINT"DRUKT U OP EEN TOETS"
7037 GETQ$:IF Q$=""THEN7037
7050 PRINT"J"
7060 PRINT"/ KUNT DE BALLETJES VERPLAATSEN DOORHET RESP. INTOETSEN VAN"
7062 PRINT"2 CIJFERS VOOR "
7070 PRINT"DE POSITIE, EN EEN PIJL VOOR DE RICHTING (SRSR TOETS)."
7075 PRINT
7080 PRINT:PRINT"OILT U STOPPEN, TOETS S IN."
7085 PRINT
7090 PRINT"•IET U HET ECHT NIET MEER ZITTEN? •TART HETSPEL DAN EENS OPNIEUW"
7095 PRINT"EN TOETS EEN H IN."
7100 PRINT"R STAAT U DAN EEN AANGENAME VERRASSING TE WACHTEN!"
7110 PRINT:PRINT"•UCCES!"
7120 GET Q$:IF Q$=""THEN7120
7125 PRINT"J";CHR$(142)
7130 RETURN

```

CHANGE_SET_en_ASSEMBLER

Van de heer van Westerhoven ontvingen wij een zeer uitgebreid schrijven. De door hem ontwikkelde systeemsoftware ontvingen wij voorzien van een volledige beschrijving. Het werd vergezeld van een brief waarin de heer van Westerhoven aanbiedt zijn 'medeleden te laten profiteren van de door mij ontwikkelde systeem software'. Een loffelijk streven wat wij uiteraard met beide handen aannemen. De layout was dermate goed verzorgd dat wij de beschrijving en listings direct over konden nemen voor publicatie. Wij hopen dat er nog meer leden in onze club zitten die op deze manier aan 'VIC PRIMEURS' mee willen

schrijven. Natuurlijk willen wij dhr. van Westerhoven namens de leden en redactie vriendelijk bedanken voor dit artikel.

Een opmerking die ontbreekt in de beschrijving is dat de programma's werken op een VIC-20 voorzien van een 8 K of 16 K RAM uitbreiding.

Een overzicht van het artikel :

- a). Programma "move basic+set"
- b). Programma "change set"
- c). Systeembeschrijving voor a en b
- d). Basic assembler voor 6502 taal
- e). Systeembeschrijving voor basic assembler
- f). Toelichting op de syntax en het adresseren in de basic assembler
- g). Programmavoorbeeld in de assemblerstaal

Programma om het beginadres van Basic te verplaatsen naar een hoger gelegen gebied en tevens een tekenset naar RAM over te brengen.

```

1000 REM move basic+set
1010 LET PS=4096 : LET CS=37888 : LET PC=205
1020 LET SO=34816 : LET SN=5120 : LET BN=22
1030 FOR N=0 TO 511
1040   POKE N+SN,PEEK(N+SO)
1050 NEXT N
1060 PRINT "clh"
1070 POKE 36869,PC
1080 FOR N=0 TO 63
1090   POKE N+PS+110,N
1100   POKE N+CS+110,O
1110 NEXT N
1120 POKE BN*256,0
1130 POKE 44,BN
1140 NEW

```

BN=nieuwe pagina nummer voor Basic

Na afloop van het programma zijn alle betrokken pointers voor Basic verzet. Dit blijft gelden tot de machine uitgezet wordt.
Het vrijgekomen gebied beneden Basic kan gebruikt worden voor tekenset(s) of machineprogramma's.

Voor berekening van de kode in PC zie 'VIC COMPUTING' bladzijde 8 en 9 van februari 1982.

Handleiding:

Het over te brengen setgebied wordt aangegeven in SO.
Het nieuwe setgebied wordt aangegeven in SN (is veelvoud van 1024).

PS=picture screen.

PC=nieuwe code voor locatie 36869.

BN=nieuwe pagina nummer voor Basic

Na afloop van het programma zijn alle betrokken pointers voor Basic verzet. Dit blijft gelden tot de machine uitgezet wordt.

Het vrijgekomen gebied beneden Basic kan gebruikt worden voor

tekenset(s) of machineprogramma's.

Voor berekening van de kode in PC zie 'VIC COMPUTING' bladzijde 8 en 9 van februari 1982.

Programma om een_tekenset te veranderen

```

1000 REM change set
1010 DIM A1$(1),A2$(1),A1,A2,N%,N1%,C1%,C2%,C3%,PS,CS,PC
1020 LET PS=4096 : LET CS=37888 : LET PC=205
1030 PRINT "Ophaal teken :" ; : INPUT A1
1040 IF A1>63 THEN 1310
1050 PRINT "breng het naar :" ; : INPUT A2
1060 IF A2>63 THEN 1310
1070 LET A1=A1*8+5120 : LET A2=A2*8+5120
1080 FOR N=0 TO 7
1090   LET C1=N*22+183 : LET C2=PEEK(A1+N) : LET C3=128
1100   FOR NL=0 TO 7
1110     IF NL=0 THEN POKE C1+PS-1,103 : POKE C1+CS-1,0
1120     IF N=0 THEN POKE N1+C1+PS-22,100 : POKE N1+C1+CS-22,0
1130     IF C2>C3 THEN POKE N1+C1+PS,81 : POKE N1+C1+CS,0
1140     IF NL=7 AND C2=1 THEN POKE N1+C1+PS,81 : POKE N1+C1+CS,0
1150     IF NL=7 THEN POKE N1+C1+PS+1,101 : POKE N1+C1+CS+1,0
1160     IF N=7 THEN POKE N1+C1+PS+22,99 : POKE N1+C1+CS+22,0
1170     IF C2>C3 THEN LET C2=C2-C3
1180   LET C3 C3/2
1190 NEXT NL
1200 NEXT N
1210 INPUT A1$
1220 FOR N=0 TO 7
1230   LET C1=N*22+183 : LET C3=128 : LET C2=0
1240   FOR NL=0 TO 7
1250     IF PEEK(N1+C1+PS) 81 THEN LET C2=C2+C3
1260     LET C3=C3/2
1270     IF NL=7 THEN POKE A2+N,C2
1280   NEXT NL
1290 NEXT N
1300 GOTO 1030
1310 PRINT "Ophal" : LET C2=0
1320 POKE 36869,PC
1330 FOR N=0 TO 63
1340   LET C3=N/11 : IF INT(C3)=N/11 THEN LET C2=C2+22
1350   POKE N*2+C2+PS+110,N : POKE N*2+C2+CS+110,0
1360   NEXT N
1370 END

```

Handleiding:

PS=picture screen, CS=colour screen, PC=kode op locatie 36869.
Achter 'haal teken' wordt het nummer aangegeven van het teken dat veranderd moet worden.
Achter 'breng het naar' wordt het nummer aangegeven van het teken dat de veranderde waarde zal ontvangen.
Vervolgens kan met de cursor naar de te wijzigen plaats Bepotitioneerd worden en kan de wijziging met spatie en zwart balletje gemaakt worden.
Na het aanbrengen van de wijziging de 'return' toets drukken.
Programma kan verlaten worden door op de eerste regel alleen de 'return' toets te drukken.

De kode in PC behoort bij het gebied van de eigen set. In dit geval is dat 5120.
Voor de berekening van het setadres en bijbehorende kode zie 'VIC COMPUTING' bladzijde 8 en 9 van februari 1982

20 april 1982

20 april 1982

Systeembeschrijving voor de programma's
"Move basic+set" en "Change set"

ASSEMBLER-1

pag 1

Eij het aanzetten van de VIC wordt 'Start basic' automatisch aangelegd op de laagste geïadresseerde pagina van het RAM. Het gebruik van het RAM wordt weergegeven in 'A'. Het vrije geheugen dat overblijft is afhankelijk van de omvang van het basicprogramma en het aantal gebruikte variabelen, arrays en strings. Dit is geen ideale situatie om een eigen tekenset in systeemroutines onder te brengen. Het verdient daarom aanbeveling om een vrije ruimte te creëren buiten het basicgebied. Dit is mogelijk door 'Start basic' één of meer pagina's hoger te plaatsen. Dit kan door te kijken wat er in locatie 44 zit. Stel dat dit 18 is. Dan kan 'Start basic' verhoogd worden door in de calculator mode te typen:

POKE 44,22

NEW

Tot de VIC uitgezet wordt is 'Start basic' nu verplaatst en is een stabiele vrije ruimte verkregen.

Wanneer in deze vrije ruimte een tekenset gebracht moet worden kan na het aanzetten van de VIC het programma "Move basic+set" gebruikt worden. Hiermee wordt de situatie van 'A' machinaal omgebouwd tot 'E'. Hierna kan programma "Change set" gebruikt worden om de diverse tekens te veranderen.

Om het vrij gekomen gebied ook te kunnen vullen met systeemroutines die vanuit basic aangesprongen kunnen worden, kan het programma "Assembler-1" benut worden. Dit testaat uit 2 delen t.w.:

a) Programma's om een source file te kunnen bewerken.

b) De assembler

Tijdens het assembleren wordt de aangeboden source file bewerkt en wordt ie berekende objectkode direct op de aangewezen plaats in het RAM gezet. Na het assembleren wordt het basicprogramma ingelezen dat bij de tekenset en de systeemroutines hoort en is een funktionele programmatiektombinatie tot stand gebracht.

Om te voorkomen dat dit bijeenvoegen van komponenten steeds opnieuw moet gebeuren kan het geheel op tape gezet worden.

Bediening voor save naar tape:

- Zet 'Start basic' terug naar pagina '18' (zie A)
 - SAVE "naam",1,1

Bediening voor load vanaf de tape:

- Zorg ervoor dat 'Start basic' verschoven is naar de pagina waarop voorneem het programma geladen heeft.
- Daarna:
 - LOAD "naam",1,03

```

1000 PRINT "clhsource lengte:" ; : INPUT PL
1010 CS=37888; PS=4096 : REM assembler-1
1020 DIM P$(PL), T(5)
1030 PRINT "clh"
1040 PRINT "programma:" ; : INPUT PR
1050 IF PR>1 THEN 1030
1060 ON PR GOTO 1070,2000,4000,5000,6000,7000,8000,3000,
1070 PRINT "R#" ; : INPUT LN : PRINT "cursor 14x<";
1080 IF LN>PL OR LN=0 THEN 1050
1090 P$(LN)=""
1100 IF PR=0 AND LEN(P$(LN))=0 THEN 1030
1110 IF PR=0 THEN LN=LN+1 : PRINT "R*" ; LN ; TAB(8) ; : GOTO 1090
1120 GOTO 1070

1200 FOR LN=1 TO PL
1210 IF LEN(P$(LN))>>0 THEN NEXT LN
1220 FOR T=LN+1 TO PL
1230 IF LEN(P$(T))<>0 THEN P$(LN)=P$(T) : P$(T)=""
1240 IF T=PL THEN 2070
1250 NEXT T
1260 NEXT LN
1270 GOTO 1030

1300 T1=3 : IF PR=9 THEN T1=4
1305 OPEN 1,T1,0
1308 PRINT "R#" ; : INPUT LN : T-LN : PRINT
1309 IF PR=4 AND LN>PL THEN CLOSE 1 : GOTO 1040
1310 IF LN>PL OR LN=0 THEN CLOSE 1 : GOTO 1030
1320 PRINT "#1,R#" ; LN : TAB(10) ; P$(LN)
1325 IF LN=T+18 AND PR=4 THEN PRINT : CLOSE 1 : GOTO 1040
1330 IF LN=T+19 THEN LN=LN+1 : PRINT : GOTO 3008
1340 LN=LN+1 : GOTO 3010

1400 PRINT "R#" ; : INPUT LN : PRINT "cursor 13x<";
1405 IF LN>PL OR LN=0 OR P$(PL)<" " THEN 1030
1410 IF P$(LN)="" THEN 4060
1420 FOR T=PL TO LN+1 STEP -1
1430 IF P$(T-1)<" " THEN P$(T)=P$(T-1) : P$(T-1)=""
1440 NEXT T
1450 INPUT P$(LN) : PRINT
1460 OPEN 1,3,0 : LN=LN-9 : IF SGN(LN)=-1 THEN LN=1
1470 T=LN : GOTO 3020

1500 PRINT "naam:" ; : INPUT NA$
1510 OPEN 1,1,1,NA$
1520 FOR LN=1 TO PL
1530 IF P$(LN)="" THEN PRINT "#1,P$(LN)
1540 PRINT "#1,P$(LN)
1550 NEXT LN
1560 CLOSE 1 : GOTO 1030

```

ASSEMBLER-1

pag 2

pag 3

ASSEMBLER-1

```

6000 PRINT "naam:" ; : INPUT NA$
6010 OPEN 1,1,0,NA$ 
6020 FOR LN=1 TO PL
6030 INPUT#,P$(LN)
6040 IF P$(LN)="#*" THEN P$(LN)="": GOTO 6060
6050 NEXT LN
6060 CLOSE 1 : GOTO 1030

7000 T(5)=2 : IF PR=10 THEN T(5)=4
7002 OPEN 1,T(5),0
7005 PRINT "vanaf: " ; : INPUT FL : PRINT : IF FL=0 THEN CLOSE 1
    GOTO 1030
7010 LL=1
7020 LN=PEEK(FL)
7030 RESTORE
7040 FOR T=0 TO LN
7050 READ NA$ : NEXT T
7060 PRINT "#1,"L#" ; FL; TAB(10);
7070 T(1)=VAL(LEFT$(NA$,1)) : LN$="": IF T(1)=0 THEN T(1)=1 :
    GOTO 7150
7080 IF T(1)=1 THEN IN$=MID$(NA$,2,LEN(NA$)) : GOTO 7150
7090 T(2)=PEEK(FL+1) : T(3)=PEEK(FL+2)
7100 IF T(1)=3 THEN T(2)=T(3)*256+T(2)
7110 IF LEN(NA$)>4 AND MID$(NA$,5,1)="(" THEN IN$=MID$(NA$,2,3)+"
    T=6
7120 IF LEN(NA$)>4 AND MID$(NA$,2,3)="." THEN
    IN$=MID$(NA$,2,3)+": T=5
7125 IF LEN(NA$)=4 THEN IN$=MID$(NA$,2,3)+" " : T=0
7130 T$=STR$(T(2)) : IN$=IN$+MID$(T$,2,LEN(T$))
7140 IF T(0)>0 THEN T(2)=IN$+MID$(NA$,2,LEN(NA$))
7150 PRINT "#1,IN$:
7160 FL=FL+T(1) : LL=LL+1
7170 IF LL>20 THEN PRINT : GOTO 7005
7180 GOTO 7020

8000 LN=0
8010 FOR FL=1 TO PL : IF LEN(NA$)=0 THEN 8150
8020 NA$=P$(FL) : IF LEN(NA$)!=":" THEN LN=VAL(NA$) : GOTO 8150
8030 IF LEFT$(NA$,1)<" :" THEN LN=VAL(NA$)
8040 T$=STR$(LN)
8050 IF LEFT$(NA$,1)=T$ : GOTO 8150
8060 IF LEFT$(NA$,1)="+" THEN P$(FL)=LEFT$(NA$,3)+T$ : GOTO 8150
8070 IF LEFT$(NA$,1)="@" AND LN<256 THEN P$(FL)=LEFT$(NA$,3)+T$:
    GOTO 8150
8075 IF LEFT$(NA$,1)="K" THEN 8150
8080 IF LEFT$(NA$,1)="@" AND LN>255 THEN 8140
8085 IF LEN(NA$)=3 THEN LN=LN+1 : GOTO 8150
8090 T=5
8100 IF MID$(NA$,5,1)="(" THEN T=6
8110 IF MID$(NA$,T,1)="@" THEN LN=LN+3 : GOTO 8150
8120 IF MID$(NA$,T,1)="@" THEN LN=LN+2 : GOTO 8150
8130 IF MID$(NA$,T,1)="K" THEN LN=LN+2 : GOTO 8150
8135 IF MID$(NA$,T,1)<" :" THEN LN=LN+2 : GOTO 8150

8140 PRINT "L#"; FL; TAB(11) ; "syntax p-1"
8150 NEXT FL

8160 LN=0
8170 FOR FL=1 TO PL
8180 T(1)=0 : T(2)=0 : T(3)=0 : T=1
8185 LN$=""
8190 NA$=P$(FL) : IF LEN(NA$)=0 THEN LN=VAL(NA$) : GOTO 8500
8200 IF LEFT$(NA$,1)<" :" THEN LN=VAL(NA$) : GOTO 8500
8210 IF LEFT$(NA$,1)="+" THEN 8500
8220 IF LEFT$(NA$,1)="@" THEN 8500
8230 IF LEFT$(NA$,1)="K" THEN 8500
8240 IN$=LEFT$(NA$,3)
8260 IF LEN(NA$)=3 THEN IN$=NA$ : GOTO 8410
8270 IF M$=MID$(NA$,5,1)<" :" THEN IN$=LEFT$(NA$,3) : T=5 : GOTO 8290
8280 IN$=LEFT$(NA$,3)+" " : T=6
8290 IF LEN(NA$)>T+2 THEN IN$=IN$+MID$(NA$, (T+3), LEN(NA$))
8295 IF LEN(NA$)>T+2 AND M$=MID$(NA$,T,1)<" :" THEN
    T(2)=VAL(MID$(NA$,T,3))
    IF LEN(NA$)>T+2 AND MID$(NA$,T,1)<" :" THEN T=9 : GOTO 8380
8296 IF LEN(NA$)>T+2 AND MID$(NA$,T,1)<" :" THEN T=9
8300 FOR T=1 TO FL
8310 IF P$(T1)" " THEN 8340
8320 IF LEFT$(P$(T1),3)=MID$(NA$,T,3) THEN 8345
     NEXT T1
8330 PRINT "L#"; FL; TAB(11) ; "syntax p-2" : GOTO 8350
8340 T(2)=VAL(MID$(P$(T1),5,LEN(NA$)))
8345 IF LEFT$(P$(T1),1)="@" THEN T=3
8350 IF LEFT$(P$(T1),1)="K" THEN T=2
8360 IF LEFT$(P$(T1),1)="@" THEN T=2
8370 IF LEFT$(P$(T1),1)="K" THEN T=9
8380 IF T>3 AND T<9 THEN T=1
8390 IF T(2)>255 THEN T(3)=INT(T(2)/256)
8400 IF T(2)>255 THEN T(2)=T(2)*256+T(2)
8410 IN$=MID$(T$,2,LEN(T$))+IN$
8420 RESTORE
8430 FOR T1=0 TO 255
8440 READ NA$ : IF NA$=IN$ THEN 3470
8450 NEAT T1 : POKE LN,T(1) : LN=LN+1
8460 PRINT "L#"; FL; TAB(11) ; "syntax p-2" : GOTO 8500
8470 T(1)=T1 : POKE LN,T(1) : LN=LN+1
8480 IF T=2 THEN POKE LN,T(2) : LN=LN+1
8485 IF T=9 THEN POKE LN,T(2) : LN=LN+1
8490 IF T=3 THEN POKE LN,T(2) : LN=LN+1
8495 THEN POKE PS+19, INT(FL/100)*48 : POKE CS+19,0
8510 IF FL>9 THEN POKE PS+20,-INT(FL/100)*11*INT(FL/10)+48 :
    POKE CS+20,0
8520 POKE PS+21,-INT(FL/10)*10+FL*48 : POKE CS+21,0
8530 NEXT FL
8540 GOTO 1040

```

```

10000 DATA 1brk,2ora(.x),0,0,0,2ora,2asl,0
10008 DATA 1php,9ora,iasl,0,0,3ora,3asl,0
10016 DATA 9bpl,2ora().y,0,0,0,2ora.x,2asl.x,0
10024 DATA 1clc,3ora,y,0,0,0,3ora,x,3asl.x,0
10032 DATA 3jsr,zand(.x),0,0,2bit,zand,2rol,0
10040 DATA 1plp,zand,lrol,0,3bit,3and,3rol,0
10048 DATA 9bmi,2and().y,0,0,0,2and.x,2rol.x,0
10056 DATA 1sec,3ani.y,0,0,0,3and.x,3rol.x,0
10064 DATA 1rti,2eor(.x),0,0,0,2eor,2lsr,0
10072 DATA 1pha,2eor,1lsr,0,3jmp,3eor,31sr,0
10080 DATA 9buc,2eir(.y),0,0,2eon.x,21sr.x,0
10088 DATA 1cli,3eir.y,0,0,0,3eir.x,31sr.x,0
10096 DATA 1rts,2aic(.x),0,0,0,2adc,2ror,0
10104 DATA 1pla,9aic,1ror,0,3jmp(),3adc,3ror,0
10112 DATA 9bvs,2aic(.y),0,0,0,2adc.x,2ror.x,0
10120 DATA 1sei,2adc.y,0,0,2adc.x,2ror.x,0
10128 DATA 0,2sta(.x),0,0,2sty,2sta,2stx,0
10136 DATA 1dey,0,1txa,0,3sty,3sta,3stx,0
10144 DATA 9bcc,2sta().y,0,0,2sty.x,2sta.x,2stx.y,0
10152 DATA 1tya,3sta.y,1txs,0,0,3sta.x,0,0
10160 DATA 9idy,2ida(.x).9ids,0,1day,2ida,2ldx,0
10168 DATA 1tay,9ida,1tax,0,3day,3lda,3ldx,0
10176 DATA 9bcs,2ida().y,0,0,2ldy.x,21da.x,2ldx.y,0
10184 DATA 1clc,3ida.y,1tsx,0,3ldy.x,3lda.x,3ldx.y,0
10192 DATA 9cpy,2cmp(.x),0,0,2cpy,2cmp,2dec,0
10200 DATA 1lny,9cmpl,1dex,0,3cpy,3cmp,3dec,0
10208 DATA 9bne,2cmp().y,0,0,2cmp.x,2dec.x,0
10216 DATA 1cld,3cmp.y,0,0,3cmp.x,3dec.x,0
10224 DATA 9cpk,2sbc(.x),0,0,2cpk,2sbc,2inc,0
10232 DATA 1inx,9sbc,1nop,0,3cpk,3sbc,3inc,0
10240 DATA 9beq,2sbc().y,0,0,0,2sbc.x,2inc.x,0
10248 DATA 1sed,3sbc.y,0,c,0,3sdc.x,3inc.x,0

```

Let op :

PS=plaats van picture screen
CS=plaats van colour screen

Onderstreepte informatie tussen "" moet nog verder uitgewerkt worden om in te kunnen geven via het keyboard.

Het basic programma "Assembler-1" heeft tot doel een source file te kunnen samenstellen en bewerken tot een logische eenheid, die door middel van de assembler omgebouwd kan worden tot een objectprogramma in het RAM.

De navolgende onderdelen zijn hiervoor beschikbaar:

- 0 = Source opdrachten ingeven met automatische regelnummering
- 1 = Source opdrachten ingeven met handmatige regelnummering
- 2 = Lege regels uit source file verwijderen
- 3 = Source file listen op het scherm
- 4 = Opricht in source file tussenvoegen
- 5 = Source file save'n naar tape
- 6 = Source file load'en vanaf tape
- 7 = Object programma in memory listen op het scherm
- 8 = Assembleren
- 9 = Source file listen op de printer
- 10 = Object programma in memory listen op de printer
- 11 = (kan aangepast worden voor het uittesten van routines)

In het programma "Assembler-1" moeten drie details aangepast worden nl.:

- PL = source lengte via input (moet veelvoud van 20 zijn)
- FS = plaats van het picture screen
- CS = plaats van het colour screen

De addressing is variabel en gaat altijd via een label.
Er zijn voor dit doel 2 soorten adreslabels:
a) f00 t/m fzz voor 2 byte adressen (0 - 65535)
b) e00 t/m ezz voor 1 byte adressen (0 - 255)

Data en displacements kunnen direct als een decimaal getal van drie cijfers aangegeven worden of door middel van een label gedefineerd worden

KOO t/m KZZ voor data/displ. van 1 byte (0 - 255)

Tijdens het assembleren worden de foutieve regels gelist met vermelding van het regelnummer.

F-1 is labelfout in pass 1

F-2 is syntaxfout in pass 2 of ontbreken van label

Bijgaand programvoorbeeld zal het gebruik van labels en adressen duidelijk maken.

Bijgaande lijst geeft de syntax weer voor iedere opdracht.

Bijlage: 1 syntaxoverzicht
1 programvoorbeeld

Toelichting voor de 6502 basic assembler

De syntax voor de instrukties kan teruggevonden worden in de DATA tabel vanaf regel 10000. Vanaf dit punt wordt per regel de informatie gegeven voor 8 instrukties.

- 1) o is een aanwijzing dat dit operatienummer niet bestaat bv. OP kode 4 = 0
- 2) De data voor een instruktie is opgebouwd uit 3 delen; 1, 2, 3 of 9)
 - a) Op de eerste plaats altijd een cijfer (1, 2, 3 of 9)
 - 1 = 1 byte instruktie zonder adres
 - 2 = 2 byte instruktie met 1 byte adres (@AA)
 - 3 = 3 byte instruktie met 2 byte adres (@AA)
 - 9 = 2 byte instruktie met 1 byte constante (KON)
 - b) Of direct decimaal getal van 3 cijfers.
 - c) Op de derde plaats de adressering van het geheugen bv.:

OP 108 = 3jmp()	JMP (@AA)	; indirect
OP 57 = 3and.y	AND @AA.y	; direct met y
OP 33 = 2and(.x)	AND (@AA.x)	; pre-indexed met x
OP 145 = 2sta().y	STA (@AA).y	; post-indexed met y
OP 80 = 9bvc	BVC KON	; constante direct
OP 136 = 1dey	DEY	; alleen OP

- b) Op de tweede plaats altijd een mnemonische kode van 3 letters bv.: AND, ADC, JMP enz.
- c) Op de derde plaats de adressering van het geheugen bv.:
- | | | |
|-------------------|-------------|----------------------|
| OP 108 = 3jmp() | JMP (@AA) | ; indirect |
| OP 57 = 3and.y | AND @AA.y | ; direct met y |
| OP 33 = 2and(.x) | AND (@AA.x) | ; pre-indexed met x |
| OP 145 = 2sta().y | STA (@AA).y | ; post-indexed met y |
| OP 80 = 9bvc | BVC KON | ; constante direct |
| OP 136 = 1dey | DEY | ; alleen OP |

Hieronder volgt een tabel die de betekenis weer geeft van de mnemonische kode en de flags die deze kode beïnvloedt.

Flags	Mnem. code	Omschrijving
SZ	LDA	Laad akku vanuit geheugen
SZ	STA	Breng akku naar geheugen
SZ	LDX	Laad register x vanuit geheugen
SZ	LDY	Breng register x naar geheugen
SZ	STY	Laad register y vanuit geheugen
SZ	ADC	Breng register y naar geheugen
SZ	AND	Tel gehugen met carry op in akku
SZ	BIT	'and' akku met geheugen (alleen STATUS)
SZ	CMP	Vergelijk akku met geheugen
SZ	EOR	'xor' akku met geheugen
SZ	ORA	'or' akku met geheugen
SZC	SBC	Trek gehugen en carry af van akku
SZ	INC	Tell 1 op bij geheugen
SZ	DEC	Trek 1 af van geheugen
SZC	CPX	Vergelijk register x met geheugen
SZC	CPY	Vergelijk register y met geheugen
SZC	ROL	Gehugen cyclisch links schuiven
SZC	ROR	Gehugen cyclisch rechts schuiven
SZC	ASL	Gehugen links schuiven
SZC	LSR	Gehugen rechts schuiven
SZC	ECC	Spring als c=0
SZC	BCS	" " c=1
SZC	BBQ	" " 2=1
SZC	BMI	" " s=1

Flags	Mnem. code	Omschrijving
S		Als bit 7=1 dan is s=1
V		v=1 als overflow is opgetreden
D		d=1 als decimal mode gebruikt wordt
I		i=1 is interrupt niet mogelijk
Z		Als inhoud van byte=nul dan z=1
C		Als v=0 instruktie zonder aktiviteit
		Betekenis van de flags
S		Als bit 7=1 dan is s=1
V		v=1 als overflow is opgetreden
D		d=1 als decimal mode gebruikt wordt
I		i=1 is interrupt niet mogelijk
Z		Als inhoud van byte=nul dan z=1
C		Als v=0 instruktie zonder aktiviteit

Programmavoorbeeld voor de 6502 basic assembler

(Print 2 maal het alfabet)

4610	LDA KST STA \ddagger CY LDX 001 LDA 000 STA \ddagger PA.x STA \ddagger CA.x INX LDA 016 STA \ddagger PA.x LDA 148 STA \ddagger CA.x	Zet 4610 in Programmateller Laad konstante KST in akku Breng akku naar adres \ddagger CY Laad 1 in register x Laad 0 in akku Breng akku naar adres \ddagger PA+x Breng akku naar adres \ddagger CA+x Verhoog register x met 1 Laad konstante 16 in akku Breng akku naar adres \ddagger PA+x Laad konstante 148 in akku Breng akku naar adres \ddagger CA+x
\ddagger VA	JSR \ddagger AL DEC \ddagger CY BMI 003 JMP \ddagger VA RTS	Adreslabel voor volgende regel Spring naar subroutine \ddagger AL Verminder geheugen \ddagger CY met 1 Spring 3 byte verder als s=1 Spring naar adres \ddagger VA Keer terug van subroutine
\ddagger AL	LDX 063	Adreslabel voor volgende regel
\ddagger VL	TXA	Laad register x met 63
\ddagger PA	STA \ddagger LC.x LDA 000	Adreslabel voor volgende regel Breng register x naar akku
\ddagger CA	STA \ddagger LC.x DEX BMI 003 JMP \ddagger VL LDX 001 LDA K88 CLC ADC \ddagger PA.x STA \ddagger PA.x LDX 001 LDA 088 CLC ADC \ddagger CA.x STA \ddagger CA.x RTS	Adreslabel voor volgende regel Adreslabel voor volgende regel Breng akku naar adres \ddagger LC+x Laad 0 in akku Adreslabel voor volgende regel Breng akku naar adres \ddagger LC+x Verminder register x met 1 Spring 3 byte verder als s=1 Spring naar adres \ddagger VL Laad register x met 1 Laad akku met konstante K88 Maak c=0 Tel inhoud van adres \ddagger PA+x en c op in akku Breng akku naar adres \ddagger PA+x Laad 1 in register x Laai 88 in akku Maak c=0 Tel inhoud adres \ddagger PA+x en c op in akku Breng akku naar adres \ddagger CA+x Keer terug van subroutine
\ddagger CY	BRK K88 88 KST 1	Adreslabel voor volgende regel Reserveer 1 byte geheugen Definitie voor de waarde van label K88 Definitie voor de waarde van label KST
4096	Zet 4096 in de programmateller	Zet 4096 in de programmateller
\ddagger LC		Adreslabel voor volgende regel

4-5-1982

DRIE_NIEUWE_ROM-CASSETTES

Tegelijkertijd met deze 'VIC PRIMEURS' komen er drie nieuwe ROM-cassettes uit voor de VIC-20. Ze zijn alle 3 ontwikkeld door het zweedse bedrijf DATATRONIC AB (zweedse importeur van COMMODORE producten). Ze zijn voorzien van een engelse handleiding.

Kort overzicht van VIC-GRAF, VIC-STAT en VIC-FORTH.

VIC-GRAF: Een ROM-cassette waarmee het mogelijk is grafieken van FUNCTIES te tekenen en te analyseren. Na het invoeren van de functie kunt u deze laten tekenen (in hoge resolutie) en de nulpunten en extreme waarden laten berekenen. Ook is het mogelijk de intergraal van een bepaald bereik te laten bepalen. De ingevoerde functie kan gewijzigd worden of vervangen worden door een nieuwe functies. Een programma dat het analyseren en begrijpen van functie met behulp van een computer vereenvoudigt en een goede aanvulling vormt bij de VIC-20 in onderwijs toepassingen.

VIC-STAT: Deze ROM-cassette is speciaal ontwikkeld voor het maken van statistische berekeningen. Het standaard BASIC van de VIC-20 wordt uitgebreid met een aantal krachtige, nieuwe statements. Grafische afbeeldingen kunnen gemaakt worden met de opdracht PLOT, 2024 punten dus 44 bij 46. Deze punten kunnen ook weer gereset worden met PLOTD en getest worden met PLOTC. Met BARV en BARTH kunt u 'staafdiagrammen' tekenen. SCALE voorziet alles van een

schaalverdeling en de opdracht COLR zorgt voor de gewenste kleuren. Met PAPER kan het gehele scherm of een gedeelte hiervan direct op de printer gezet worden en heeft u de gegevens op papier.

Dan nu een aantal opdrachten die werken op een array met daarin de gegevens die u wilt verwerken. De opdracht STAT berekent van dit array het gemiddelde, de variantie en de standaard deviatie. LINREG en LINKO berekenen respectievelijk de lineaire-regressie coefficienten en de lineaire-correlatie. Tot slot de opdracht SO om zowel een string als numeriek array te sorteren. Al deze opdrachten kunt u in ieder gewenst BASIC-programma gebruiken en zo wordt u veel reken- en/of tekenwerk bespaard en krijgt u op een eenvoudige wijze inzicht in uw gegevens.

VIC-FORTH: Een compleet nieuwe programmeer taal voor de VIC-20. Deze versie van FORTH is special voor de VIC-20 gemaakt en komt vrijwel overeen met het standaard fig-FORTH. Hoewel FORTH ontwikkeld is voor gebruik met een disk-systeem werkt VIC-FORTH ook met cassette. Apart leverbaar is een EDITOR en ASSEMBLER op diskette. Bijgeleverde gebruiksaanwijzing behandelt de ongeveer 250 woorden van het VIC-FORTH vocabulair. Niet direct te begrijpen voor iemand die nog nooit iets over FORTH gelezen heeft. Een aanvulling hierop is de PET-FORTH handleiding een 320 pagina's dikke handleiding van het compatibele PET-FORTH (ook uitgebracht door DATATRONIC AB). Verder is er een literatuurlijstje opgenomen achterin de VIC-FORTH gebruiksaanwijzing.

Met de VIC-20 zit u nu niet langer vast aan BASIC.
Na het aanbrengen van de

De adviesprijs van VIC-STAT en VIC-GRAF is F 175,- en voor VIC FORTH is deze F 249,-. Ze zijn verkrijgbaar bij de VIC-20 dealers.

VIC-FORTH ROM-cassette in de uitbreidingsbus van de VIC-20 wordt VIC-FORTH (net als VIC-GRAF en VIC-STAT) automatisch gestart.



MUZIEK_!

Hierbij de listing van een programma dat het u mogelijk maakt wonderschone klanken aan uw VIC-20 te ontlokken. U kunt er mee piano, luit en orgel spelen over drie oktaven en ook een melodietje opnemen en weer afspeLEN. Alles wordt trouwens door het programma zelf uitgelegd.

Het programma maakt gebruik van twee manieren om informatie van het toetsenbord binnen te krijgen, n.l. via GET en CHR\$ voor de piano en via PEEK(197) voor het orgel en de bijbehorende volumeregeling. Daar CHR\$ en PEEK(197) niet dezelfde waarde opleveren voor dezelfde toets zijn er twee DATA-reeksen nodig om van een toets naar een toonhoogte te komen (r. 1000-1020 en r. 1100-1120). Het beperkte

geheugen van de standaard VIC laat geen twee DIM-instructies toe van de benodigde lengte, dus is er volstaan met een. Deze reeks wordt nu bij de keuze van een instrument gevuld met de DATA-reeks die bij de invoermethode van dat instrument hoort.

Bij gebruik van wat geheugenuitbreidung kunnen vooral het orgel en de opneemmogelijkheid nog wat uitgebreid worden. Het opnemen zou een langere reeks tot zijn beschikking kunnen krijgen om een langere melodie in te kunnen spelen. Tevens zou hier de keuze tussen instrumenten en registers toegevoegd kunnen worden alsook de mogelijkheid voor het inspelen van een tweede en derde stem.

Intypen dus en VEEL PLEZIER !

```
1 MU$="MUZIEK ":"PRINT MU$ : PRINT "HILHORST":PRINT "MET DIT PROGRAMMA KUNT"
2 PRINT "U PIANO, LUIT EN ORGEL SPELEN EN EEN MELODIE VASTLEGGEN EN AFSPELEN"
3 FOR F=0 TO 5000:NEXT F
5 MX=110:DIM TL%(64),BD%(MX),BT%(MX)
10 C=36876
20 S2=36875
30 V=36878
40 POKE S2,0
135 A$=" ":PRINT MU$
137 PRINT "P"=PIANO"
139 PRINT "L"=LUIT"
141 PRINT "O"=ORGEL"
143 PRINT "OP"=OPNEMEN"
145 PRINT "AF"=AFSPELEN"
167 PRINT "Q"=STOPPEN"
170 INPUT A$
180 IFA$="P"THENGOSUB200
190 IFA$="L"THENGOSUB300
195 IFA$="O"THENGOSUB400
196 IFA$="OP"THENGOSUB500
197 IFA$="AF"THENGOSUB600
198 IFA$="Q"THEN3000
199 GOTO 135
```

```

500 PRINT "JOOPNEMEN ■": GOSUB5000: GOSUB4000
508 FORF=0TOMX: BTX(F)=0: BDW(F)=0:NEXTF
509 POKEV,10:POKE197,64:V1=10
510 I=0:N=64
520 NV=N
530 N=PEEK(197):W=W+5
540 IFN>640RN=15THEN590
543 IFN=64ANDNC>NVTHENBDW(I)=W:W=0:BTX(I)=TLX(NV):I=I+1:POKES2,0:GOT0520
544 IFI>MXTHEN590
545 IFN=64THENPOKES2,0:GOT0530
548 IFN=620RN=63THENGOT0530
550 POKES2,TLX(N)
553 IFNV>NTHENBDW(I)=W:W=0:BTX(I)=TLX(NV):I=I+1
560 IFI>MXTHEN590
580 GOT0520
590 POKES2,0:POKEY,0
595 FORF=0TO20:GETOP$:NEXTF
599 RETURN
600 PRINT "JOOPFSPELEN ■"
610 FORF=1TOMX
620 FORW=0TOBDW(F)
630 POKES2,BTX(F)
640 POKEY,10
650 NEXTW
670 NEXTF
690 POKEY,0:POKES2,0
699 RETURN
1000 DATA 215,0,239,212,240,0,203,0,143,151,0,167,179,187,0,199,241,0
1010 DATA 0,0,0,0,209,221,232,228,226,159,229,0,233,195,236,238,0
1020 DATA 237,235,201,207,135,163,0,175,191,231,147,225,183,223,0,217,0,219
1100 DATA ,151,167,187,199,,217,,,147,163,183,195,207,215,,,221,226,,236,,,,22
5,231
1110 DATA 235,239,,,223,228,232,237,240,,,229,233,238,241,,,135,159,175,191,2
01,209
1120 DATA 219,,143,,179,,203,212,,,0
3000 END

4000 PRINT "KOE TOETSEN ZIJN", "(ALS OP EEN PIANO):"
4005 PRINT "A A2 B B3 C C2 D D3 E E2 F F3 G G2 H H3 S S2 Q W ER T Y UI O PG * +"
4010 PRINT "Z Z2 X X3 V V2 N N3 M M2 . ,"
4020 PRINT "L LAGE C: C2M", "H HOGE C: C,M", "RETURN = STOPPEN"
4099 RETURN
5000 RESTORE
5005 FORF=0TO52:READTLX(F):NEXTF
5010 FORF=0TO64:READTLX(F):NEXTF
5099 RETURN
5100 RESTORE
5110 FORF=0TO52:READTLX(F):NEXTF
5199 RETURN

```

```

200 PRINT"J10PIANO ■":GOSUB5100:GOSUB4000
220 GETP$:IFP$=""THEN220
230 POKE$2,0
235 N=ASC(P$)-42
240 IFN<0ORND>52THEN290
269 Z=15
270 POKE$2,TLX(N):POKEC,TLX(N)
271 FORM=0TO20
272 POKEY,Z
273 IFZ>0THENZ=Z-(1-1/Z)
274 GETP$:IFP$<>""THEN230
277 FORL=0TO3:NEXT
278 NEXT
279 POKEY,0
280 GOT0220
290 POKE$2,0:POKEC,0
299 RETURN
300 PRINT"J10SLUIT ■":GOSUB5100:GOSUB4000
320 GETL$:IFL$=""THEN320
330 POKEC,0
335 N=ASC(L$)-42
340 IFN<0ORND>52THEN390
369 Z=8
370 POKEC,TLX(N)
371 FORM=0TO8
372 POKEY,Z
373 IFZ>0THENZ=Z-1:GOT0375
374 POKEC,0
375 GETL$:IFL$<>""THEN330
377 FORL=0TO12:NEXT
378 NEXT
379 POKEY,0
380 GOT0320
390 POKEC,0
399 RETURN

400 GOSUB5000
402 FORI=0TO20:GETO$:NEXTI:O$=" " :OG$="J10ORGEL ■"
403 PRINTOG$:PRINT"J10REGISTERS: ",,"RIB = BAS      ",,"AT = TENOR      ",,"AS = SOP
RAANJ10"
404 PRINT"J10000000RETURN■ = STOPPENTIIII"
405 INPUTO$:IFO$="B"THENSX=36874:GOT0409
406 IFO$="T"THENSX=36875:GOT0409
407 IFO$="S"THENSX=36876:GOT0409
408 GOT0490
409 FORF=0TO200:NEXTF:POKEY,10:POKE197,64
410 PRINTOG$:GOSUB4000:PRINT"J10F?■ = HARDER",,"J10HOME■ = ZACHTER")
430 N=PEEK(197)
440 IFN>640RN=15THEN402
445 IFN=64THENPOKESX,0:GOT0430
446 IFN=63ANDV1<15THENV1=V1+1:POKEY,V1:GOT0430
447 IFN=62ANDV1>0THENV1=V1-1:POKEY,V1:GOT0430
448 IFN=620RN=63THENGOT0430
450 POKESX,TLX(N)
480 GOT0430
490 POKESX,0:POKEY,0
499 RETURN

```

VICKIES

Uit veel vragen is ons inmiddels gebleken dat velen van jullie , maar ook VIC-dealers behoeft te hebben aan een mededelingen rubriek. Met de derde editie van VIC-primeurs voldoen we graag aan deze vraag.

Met VICKIES kun je zelf-ontwikkelde programma's en / of hardware aanbieden. Je kunt vragen stellen of een speciaal produkt opvragen. Misschien wil je met een andere VIC gebruiker bij jou in de buurt in contact komen of zelfs een nieuwe gebruikersclub opzetten , waarbij je de lezers van VIC-primeurs in jouw omgeving aanspoort om lid te worden. Ook problemen met het ontwikkelen van software kun je de lezers van VIC-primeurs voorleggen. In het kort:De algemene probleembak voor allerlei en iedereen.
"Ja, maar wat kost dat nou?": vraag je je misschien af.VICKIES , beste vrienden , zijn gratis. Maak er dus gebruik van. Stuur jouw vragen op naar:
VIC-gebruikersclub - Postbus 12972 - 1100 AZ , AMSTERDAM.

Goed , daar gaan we dan. De eerste VICKIES.

NIEUW voor VIC huiscomputer. Nu leverbaar RTTY adaptor met diverse baudsnelheden voor amateur en commercieel gebruik. Probleemloos aan te sluiten op uw telex-converter.
Technisch bureau H&B
Burgerhoutsestraat 127
4702 BD ROOSENDAAL
Tel : 01659 - 43171

VICKIES

VICKIES

VICKIES

Programma's ontwikkeld voor de VIC

KAARTENBAK

Voor dit programma heb je 8K RAM geheugen nodig. De bedoeling is om een compleet archief op te bouwen d.m.v. bestanden. Iedere kaart kan drie gegevens bevatten.

Het programma bestaat uit 8 delen,t.w.:1.Tape laden.
2.Wijzigen. 3.Invoeren. 4.Print label. 5.Print lijst. 6.Sorteren
7.Zoeken. 8.Tape saven.
Dit programma is ook op Floppydisk verkrijgbaar. Voor meer informatie kunt u kontakt opnemen met: GROHAM BV.-GELDERMALSEN-03455-2341.

MAGAZIJN BACKORDER BESTAND

Dit programma is alleen op floppydisk verkrijgbaar. De titel van dit programma spreekt voor zichzelf. Het programma gebruikt 16K RAM en kost FL.159,-. Bestellingen plaatst u bij GROHAM BV. - GELDERMALSEN - 03455 - 2341.

Zowel het kaartenbak- als het orderbestandprogramma wordt geleverd met een nederlandse gebruiksaanwijzing.

Dat de VIC daadwerkelijk in heel nederland gebruikt wordt , blijkt wel uit de volgende mededeling van een zeer aktieve dealer uit Friesland.

TEKSTWERKER

Dit programma is verkrijgbaar op cassette en gebruikt 16K RAM. Regellengte en de dikte van de letters kan vooraf worden bepaald. Het programma wordt geleverd met een complete

uitdraai op papier van datzelfde programma.

U kunt het programma bestellen door Fl. 17,50 over te maken op postgiro 919700 t.n.v.: :

Radio v. d. Galien

Willemstraat 26

DAMWOUDE

Tel: 05111 - 1396

De prijs is inclusief portokosten.

FREQUENTIE programma

Dit programma is geschreven voor het opslaan van frequenties voor scanners en wereldontvangers. Het komplette bestand van frequenties kunt u uitprinten op papier. Ook kunt u in het bestand selekteren. Als extra service ontvangt u met dit programma een cassette met scannerfrequenties in uw eigen omgeving.

U kunt het programma bestellen door Fl. 17,50 over te maken op postgiro 919700 t.n.v.: :

Radio v.d. Galien

Willemstraat 26

DAMWOUDE

Tel : 05111 - 1396

Wij wensen u veel plezier met deze twee programma's.

VIC-20 SOFTWARE

Wij ontvingen een uitgebreide folder van INFOBOEK Zeeland, waarin een uitgebreid verhaal stond m.b.t. het software door dit bedrijf ontwikkeld. Hieronder volgt een samenvatting.

TEKSTVERWERKING

Door middel van dit programma kan iedere VIC gebruiker nu zijn brieven en/of rapporten opstellen en eindeloos korrigeren en manipuleren voordat het resultaat wordt uitgeprint op de VIC-1515 printer.

MAILING

Met dit programma kan iedere vereniging nu zijn ledenlijst bijhouden, sorteren, muteren en bewaren voor later gebruik. Dit pakket kan 100 adressen per keer verwerken. Ieder adres bevat de volgende gegevens: NAAM en VOORLETTERS - ADRES - POSTCODE en WOONPLAATS - een extra gegeven (Bijv. telefoonnummer), wat NIET wordt uitgeprint op een adreslabel.

MINI VISICALC

Ingewikkelde berekeningen, budgetsamenstellingen, boekhoudvergelijkingen worden met dit programma kinderspel. Het programma werkt volgens RPN notatie, een manier van rekenen, die vooral bekend is bij de gebruikers van HP rekenmachines. U heeft 4 bewerkingenregister en 10 opslagregisters ter beschikking.

GROOTBOEK-PROEF-en SALDIBALANS

Dit pakket is een tussenschakel tussen het eenvoudig bijhouden van de dagboeken van een bedrijf en de ingewikkelde formulering van het invullen de eindbalans. Dit programma maakt vanuit uw dagboeken namelijk een grootboek, waarna u de rekeningen kunt laten uitprinten. Na het voltooien van het grootboek kan het programma op de VIC-1515 printer een volledige proef- en saldibalans afdrukken.

BIORITME

Dit is ons eerste programma ,
dat niet op het serieuze
zakelijke vlak ligt.

Het programma genereert op de
printer overzichten van uw
kritische dagen en een dagelijks
overzicht van uw emotionele ,
fysieke en intellectuele
gesteldheid

ALGEMENE INFO over BOVENSTAANDE PROGRAMMA'S.

Alle programma draaien met een
geheugen uitbreiding van 16K RAM
en maken gebruik van een
printer.

De programma's zijn leverbaar op
cassette en floppy.

Prijzen:

Tekstverwerking	Fl. 135,-
Mailing	Fl. 135,-
Mini Visicalc	Fl. 85,-
Grootboek	Fl. 150,-
Bioritme	Fl. 85,-

De VIC-gebruikersclubs

=====

De VIC-gebruikersclub , verantwoordelijk
voor de uitgave van VIC-primeurs , wil op
een aantal plaatsen in Nederland lokale
VIC-gebruikersclubs opzetten. Deze lokale
clubs zullen worden opgezet met de
medewerking van HANDIC BENELUX BV en
zullen leden in hun omgeving behulpzaam
zijn met alle voorkomende vragen ,
informatie en programma ontwikkeling.
Natuurlijk kun je allerlei kontaktvormen
bedenken , zoals de organisatie van
bijeenkomsten , clubverkopen ,
demonstraties , enz.,enz.Ook het houden
van lezingen en cursussen blijkt
succesvol te verlopen.De aktiviteiten
zullen worden ondersteund en in sommige
gevallen begeleidt door de overkoepelende
VIC-gebruikersclub.

Wij zoeken nog een aantal aktieve
VIC-eigenaren , die weten dat je
gezamelijk veel meer uit de computer kunt
halen dan alleen , en die dus zo'n lokale
club zouden willen opzetten. Als u
daartoe in de gelegenheid bent , laat het
ons weten en bel met:
HANDIC BENELUX BV
Nicky Moeken
072 - 337644

Misschien heb je t.a.v. het organiseren
van zo'n club nog een aantal suggesties ,
pak dan de pen en schrijf ons:
VIC-gebruikersclub
postbus 12972
1100 AZ AMSTERDAM

De VIC-gebruikersclub.



VICKIES

VICKIES

TELEX en MORSE decoder/converter voor de VIC-20 Computer.

Met de komst van de PET computer in de VS, vroegen enkele zend-amateurs zich af of het ook mogelijk was de computer voor hun hobby te gebruiken. De zend amateurs zijn altijd al trendsetters geweest in de electronica. Velen werken op laboratoria van grote firma's en komen daar in aanraking met de ontwikkeling van nieuwe elektronische produkten. Voorbeeld daarvan is de ontwerper van de 6502 microprocessor en de PET computer, Chuck Peddle, die nu een eigen bedrijf heeft en computer van het merk SIRIUS maakt. In de VS werkte Ron Lodewyck aan de cursus BASIC voor de PET 2001 computer. Bij hem is toen het idee geboren om een interface te maken tussen de PET en zijn Radiozend/ontvanger. Ron heeft nu een eigen bedrijf en er zijn dan ook Telex en Morse decoders voor de CBM 3000 serie, CBM 8000, VIC-20, TRS-80 model 1 en 3, Exidy sorcerer, de ITT-2020 en de Apple II computer. Met de komst van de VIC-20, kwamen er al snel vragen. Kan ik ook zo'n programma op de VIC-20 krijgen? Velen zeiden dat dit niet kon, voor ons was dat een uitdaging om het wel voor elkaar te krijgen, mede dankzij onze jarenlange ervaring met de decoders van Ron Lodewyck die Computer World in zijn assortiment heeft. Het resultaat is dan ook een interface kaart met daarop het 4 K machinetaal programma in Eprom en twee converters voor Telex en Morse.

Het gebruik is eenvoudig, alles is op een kaart gebouwd, welke de grootte van een ROM pack heeft. Er zijn aansluitingen voor zend/ontvang schakelaar, hiermee bedient men vanaf het keyboard de zender (PTT). Ook gaan er twee draadjes naar de ontvanger uitgang bv. de luidspreker of oortelefoon uitgang.

Wat kan ik er nu mee doen? Moet ik een zendexamen afleggen? Is het ingewikkeld voor mij, moet ik soms een electronicus zijn om de interface te kunnen gebruiken? Op al deze vragen kan ik Nee antwoorden. Bij het ontwerpen zijn we uitgegaan van mensen zonder specifieke elektronica kennis. De gebruiksaanwijzing is dan ook zodanig dat iedereen het kan begrijpen.

Als men de radio aanzet hoort men op de banden (KG,LG,FM), diverse stations, de een kan men verstaan, de ander spreekt een voor ons onbekende taal. Zendamateurs daarentegen spreken maar een taal (engels). Persberichten worden uitgezonden in een code. Iedere letter bestaat dan uit 5 toontjes, deze toontjes worden door de interface omgezet naar een door de computer leesbaar schrift. Hetzelfde gebeurt bij morse ontvangst. Met deze interface kan men ook zenden, dus de tekst in typen vanaf het keyboard en in de juiste code aan de zender aanbieden.

Het programma voorziet ook in de mogelijkheid om automatisch berichten te ontvangen. Hiervoor zendt het tegenstation de roepnaam uit met een code d.i. bv. PAOVICKKK (de code is KKK). Daarop zendt de VIC, die dit ontvangt, automatisch een antwoord uit. Het tegenstation zendt nu zijn bericht naar de onbemande VIC. Als de bezitter van de onbemande VIC thuis komt en zijn TV aanzet, hoort hij een toon uit de luidspreker komen ten teken dat er een bericht voor hem is. Bij het indrukken van de toets "pijl links", wordt de toon uitgezet en wordt het bericht zichtbaar op het TV scherm.

Wilt u de Telex en Morse decoder zien werken, dan kunt u bij iedere dealer die de VIC-20 verkoopt terecht. De prijs van de decoder/converter is F 595,-- incl.btw. Hiermee is de VIC-20 een geduchte konkurrent geworden van de bestaande decoders die alleen maar Telex en Morse kunnen decoderen. Zoals U weet kan de VIC-20 nog veel meer. Binnenkort hoort u meer van ons over nieuwe produkten voor de VIC-20, zoals de MODEM. Hiermee kunt u computer programma's verzenden via de telefoonlijn naar andere gebruikers. In het VIC computercentrum te Rotterdam komt een computer te staan die u kunt bellen. Via de modem kunt u dan programma's achterlaten of er andere programma's vanaf halen. Zelfs kunt u een bericht voor een andere VIC gebruiker achterlaten. Ook zijn we bezig om de VIC nederlands te laten praten, als U het niet gelooft, ga dan maar kijken bij het VIC Computer Centrum te Rotterdam.

Happy computing,

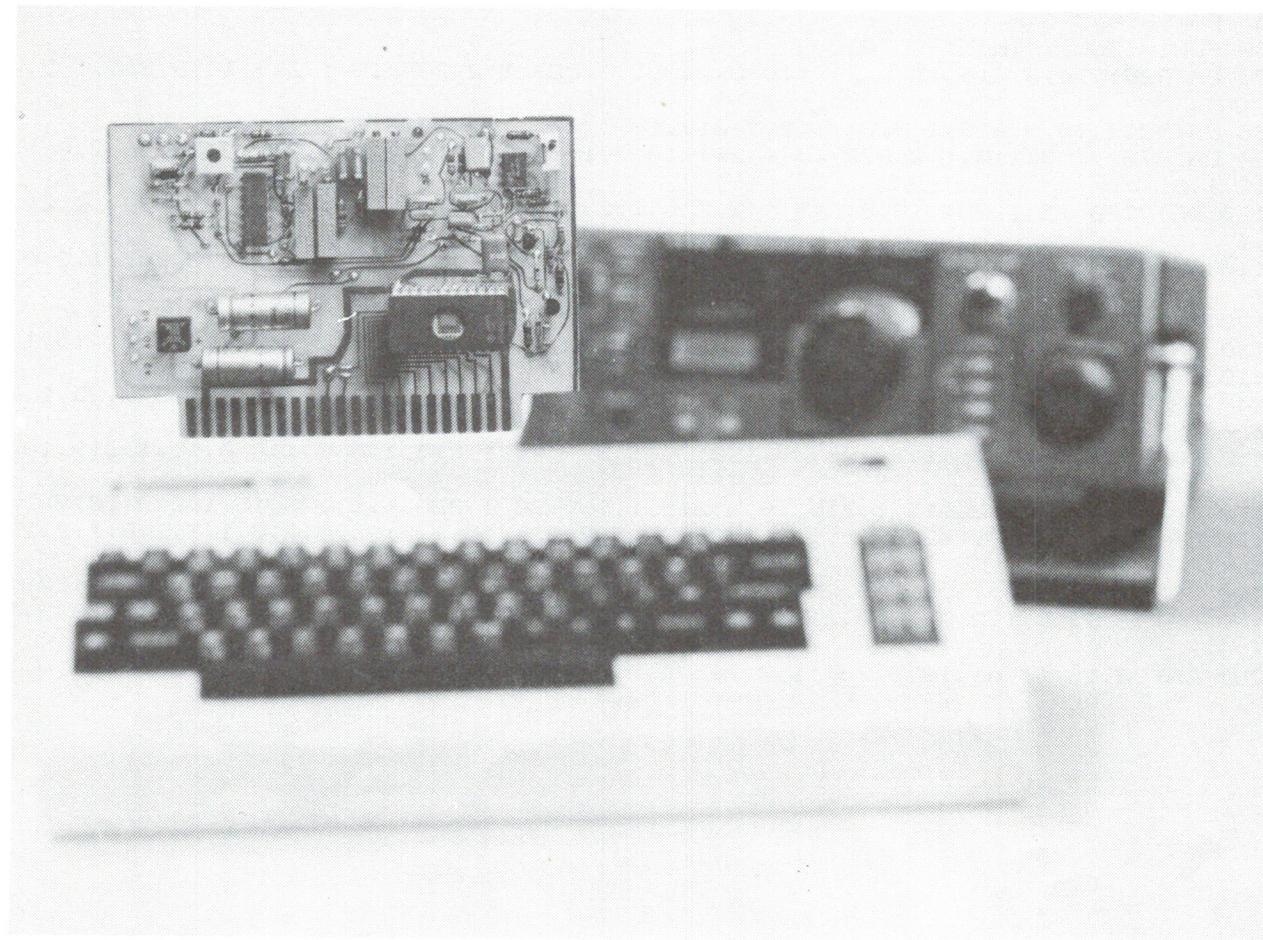
N. Karssemeijer.

VIC Computer Centrum

Keerweer 12

3012 KB Rotterdam

Tel.: 010 - 13 78 23



BEATLES : een meer tonige versie van PENNY LANE

Hieronder een listing van het liedje PENNY LANE van de BEATLES. Er wordt gebruik gemaakt van alle drie toongenerators in de VIC-20, en kan als voorbeeld dienen voor het schrijven van andere liedjes. De volgorde van gegevens in de DATA-statements is TOONHOOGTE, LENGTE en GENERATOR-nummer enz, enz, enz. Data -2 wordt gebruikt om vanaf het begin te herhalen en data -1 geeft het einde aan en het programma draait nu netjes de volume-regelaar in de VIC-20 dicht.

Veel luisterplezier !

```
10 POKE36878,6
20 READA
23 IF A=-2 AND X=1 THEN 20
25 IF A=-2 THEN CLR : X=1 : GOTO 20
30 IF A<>-1 THEN 40
35 FOR I=PEEK(36878)TO0STEP-.05:POKE36878,I:NEXT:POKE36874,0:POKE36875,0:POKE36876,0:END
40 READB,C
50 POKE36873+C,A
60 FOR I=1TOB*13:NEXT:GOTO 20
70 DATA 201,10,3,215,15,3,219,10,3,223,1,3,223,1,2,235,15,1,219,10,3,215,1,3,0,1
5,1
80 DATA 212,10,3,215,1,3,231,15,1,212,10,3,207,1,3,0,15,1,201,10,3
90 DATA 207,1,3,215,1,2,237,15,1,201,10,3,195,1,3,219,1,2,0,15,1,215,10,2,0,1,3,
212,1,2
100 DATA 228,15,1,201,10,3,215,1,3,225,1,2,0,15,1,219,10,3,223,1,3,223,1,2,235,1
5,1
110 DATA 219,10,3,215,1,3,0,15,1,212,10,3,215,1,3,231,15,1,201,10,3,207,1,3,0,15
,1
120 DATA 215,10,3,209,1,3,221,1,2,235,90,1
130 DATA 201,1,3,0,1,2,0,5,1,215,15,3,219,10,3,0,1,3,219,1,3,228,1,2,231,15,1,21
5,10,3
140 DATA 0,1,3,215,1,3,0,15,1,212,10,3,0,1,3,215,1,3,231,25,1,219,1,3,0,25,1,0,5
,3
150 DATA 221,1,3,215,1,2,229,30,1,0,30,1,229,40,1,215,1,3,0,1,2,0,15,1,219,10,3,
221,1,3
160 DATA 212,1,2,228,15,1,215,10,3,219,1,3,0,30,1,228,1,2,228,15,1,219,10,2
170 DATA 0,1,3,228,1,2,0,25,1,0,1,2,228,1,2,234,15,1,219,10,2,0,1,1,228,1,2,234,
25,1
180 DATA 0,1,2,228,1,2,235,25,1,215,1,3,223,1,2,237,15,1,219,10,3,223,1,3,223,1,
190 DATA 235,15,1,219,10,3,215,1,3,0,15,1,212,10,3,215,1,3,231,15,1,212,10,3,207
,1,3
200 DATA 0,15,1,201,10,3,207,1,3,215,1,2,237,15,1,201,10,3,195,1,3,219,1,2,0,15
,1
```

210 DATA215,10,2,0,1,3,212,1,2,228,15,1,201,10,3,215,1,3,225,1,2,0,15,1,219,10,
3
220 DATA223,1,3,223,1,2,235,15,1,219,10,3,215,1,3,0,15,1,212,10,3,215,1,3,231,1
5,1
230 DATA201,10,3,207,1,3,0,15,1,215,10,3,209,1,3,221,1,2,235,90,1,201,1,3,0,1,2
,0,5,1
240 DATA215,15,3,219,10,3,0,1,3,219,1,3,228,1,2,231,15,1,215,10,3,0,1,3,215,1,3
,0,15,1
250 DATA212,10,3,215,1,3,231,20,1,219,1,3,0,20,1,0,5,3,221,1,3,215,1,2,229,35,1
,0,30,1
260 DATA229,30,1,215,1,3,0,1,2,0,15,1,219,10,3,221,1,3,212,1,2,228,15,1,215,10,
3
270 DATA219,1,3,0,25,1,0,1,3,225,1,2,228,25,1,219,1,3,0,15,1,207,10,3,215,1,3,2
31,1,2
280 DATA225,22,1,0,5,1,225,22,1,0,5,1,225,30,1,219,1,3,209,1,2,0,15,1,221,1,3,2
15,10,2
290 DATA225,1,3,219,1,2,232,38,1,221,1,3,215,10,2,219,1,3,209,25,2,221,1,3,215,
25,2
300 DATA0,1,1,225,1,3,219,1,2,232,38,1,221,1,3,215,10,2,0,1,1,219,1,3,209,1,2,2
32,25,1
310 DATA215,1,3,225,1,2,237,25,1,209,1,3,228,1,2,238,25,1,225,1,2,0,25,1,221,1,
2
320 DATA232,25,1,219,1,2,0,25,1,221,1,2,238,25,1,215,1,2,0,25,1,219,1,2,232,25,
1
330 DATA221,1,2,0,25,1,225,1,3,219,1,2,232,38,1,221,1,3,215,10,2,219,1,3,209,25
,2
340 DATA221,1,3,215,25,2,0,1,1,225,1,3,219,1,2,232,38,1,221,1,3,215,10,2,0,1,1,
219,1,3
350 DATA209,1,2,232,25,1,215,1,3,225,1,2,237,25,1,209,1,3,228,1,2,238,25,1,215,
25,3
360 DATA219,1,3,237,25,1,221,1,3,235,25,1,219,1,3,212,1,2,228,25,1,212,1,3,225,
25,2
370 DATA0,1,1,201,1,3,212,1,2,228,25,1,0,1,3,-2
375 DATA223,1,3,215,1,2,0,15,1,225,1,3,219,10,2,228,1,3
380 DATA223,1,2,235,38,1,225,1,3,219,10,2,223,1,3,215,25,2,225,1,3,219,25,2,228
,1,3
390 DATA212,1,2,239,38,1,225,1,3,207,10,2,223,1,3,201,25,2,219,1,3,212,25,2,215
,1,3
400 DATA207,1,2,240,38,1,201,1,3,212,10,2,195,1,3,207,25,2,201,1,3,212,25,2,207
,1,3
410 DATA215,38,2,201,1,3,212,10,2,195,1,3,207,25,2,0,1,1,240,25,1,228,1,3,223,1
,2
420 DATA235,38,1,225,1,3,219,10,2,223,1,3,215,25,2,225,1,3,219,25,2,228,1,3,212
,1,2
430 DATA239,38,1,225,1,3,207,10,2,223,1,3,201,25,2,219,1,3,212,25,2,207,1,3,215
,1,2
440 DATA240,38,1,201,1,3,212,10,2,195,1,3,207,25,2,201,1,3,212,25,2,215,1,3,207
,25,2
450 DATA0,25,1,228,13,3,225,13,3,228,13,3,221,13,3,223,1,3,201,1,2,235,25,1,207
,25,2
460 DATA212,13,2,215,18,2,201,23,2,195,28,2,191,30,2
1000 DATA-1

'VIC REVEALED'

Dit is de titel van een 267 pagina's tellend boek welke in de Engelse taal geschreven is. Zoals de titel zegt wordt in dit boek de VIC onthuld en dat vooral in hardware-zaken. In het voorwoord schrijft de auteur dat het boek een verzameling van ervaringen met de VIC is, hoe en waarom hij werkt en hoe deze zaken gebruikt moeten worden tot het schrijven van betere programma's.

Het boek is in 5 secties verdeeld, elke sectie omvat een van de functionele onderdelen waarin een standaard VIC uitgesplitst kan worden.

Dit boek is geschreven voor de Europese versie van de VIC-20 en wel die ontworpen is voor het gebruik op Europese PAL TV toestellen. De VIC-CHIP die voor dit PAL standaard gebruikt wordt is de 6561 terwijl voor de andere standaards zoals het Amerikaanse 525 lijnensysteem de 6560 gebruikt wordt.

De 5 secties waarin het boek verdeeld is zijn:

1. De 6502 microprocessor.
2. VIC system software.
3. De 6561 video interface chip.
4. De 6522 VIA en de USERPORT.
5. VIC in/output functies.

Verder staan er achterin het boek een zestal aanhangsels, waaronder het schema van de VIC-20 en een gedeelte van de Machinetaalmonitor instructies.

In de eerste sectie waarin de 6502 microprocessor besproken wordt, komen o.a. ook aan bod: de interne organisatie van de 6502; geheugen organisatie en de 6502 instruktie cyclus; processor status-register en het

gebruik van flags; stack-register en zijn toepassing; interrupts en initialisatie; machinetaal op de VIC en het schrijven van machinetaal programma's. In dit hoofdstuk wordt o.a. besproken het verschil tussen RAM en ROM geheugen en ook de zeropage. Verder hoe een Basicprogramma geschreven kan worden in combinatie met machinetaal subroutines.

In de tweede sectie, VIC system software, worden ondermeer de volgende punten behandeld: de organisatie van het geheugen; VIC systeem variabelen met een tabel hiervan; VIC gebruikers geheugen; data opslag en de door de gebruiker op te roepen Kernel routines.

In het gedeelte over de organisatie van het geheugen wordt heel duidelijk uiteengezet welke locaties bestemd zijn voor: de Kernel ROM en de Basic ROM; 8K locaties voor uitbreiding RAM/ROM geheugen; scherm RAM enzovoorts.

Er komt niet helemaal duidelijk naar voren tot hoever het geheugen uitgebreid kan worden. Er wordt wel over een tot 32K adresseerbaar geheugen gesproken, maar niet dat daarbinnen het Basic gebruikers geheugen tot maximaal 29K uittebreiden is.

De systeem variabelen tabel is keurig op volgorde afgedrukt in zowel de Hexadecimale als de Decimale waarde.

Verder wordt er duidelijk besproken hoe een Basic programma in het geheugen opgeslagen wordt.

De derde sectie bespreekt het gedeelte waar het apparaat zijn naam aan ontleent heeft; te weten de Video Interface Chip (6561).

In deze sectie wordt ondermeer het VIC ic doorgelicht alsmede de interne registers van de 6561, de besturing van de afmetingen van het scherm, kleurbesturing en de 6561 geluidsgenerators.

In het gedeelte over de interne registers van de 6561 staan wat leuke testprogrammaatjes om bijvoorbeeld het beeld te laten verschuiven of zelfs om het beeld groter te laten worden. Hieronder een programmaatje om het scherm horizontaal te verschuiven.

```
10 FOR Q=0T040
20 POKE36864,Q
30 FOR X=0T01000:NEXT X
40 NEXT Q
50 POKE36864,12
```

Ook komt naar voren hoe je je eigen karakters kunt maken met behulp van de User Definable charactermode. Dit wordt allemaal aangevuld door een aantal voorbeeld programma's. Verder staat er een voorbeeld programma waarin de grafiek van een functie geplot wordt in de High-resolution mode van de VIC. In het hoofdstukje over de kleurbesturing wordt verteld hoe de kleuren gemaakt worden en dat ze onderverdeeld zijn in de scherm- en in de randkleuren. In het gedeelte over de geluidsgeneratoren van de 6561 worden de 3 audio oscillatoren, de ruisgenerator en de volume besturing besproken. Het hoofdstuk wordt besloten met wat voorbeeldprogramma's hiervoor.

Sectie 4, de 6522 VIA's en de Userpoort, behandelt o.a.: de VIC in/output poorten; de werking van die poorten; het schuitregister van de 6522 en de functiebesturing.

De in/output poorten zijn onderverdeelt in twee VIA's te weten VIA 1 en VIA 2. Alle in- en outputlijnen worden overzichtelijk besproken zodat U hierin precies kunt terugvinden hoe Uw VIC met de "buitenwereld" communiceert en ook langs welke weg dit gebeurt. Alle aansluitingen van de I/O poorten zijn ook ter verduidelijking geplaatst.

In de vijfde sectie worden o.a. de volgende hoofdzaken behandeld: de cassettepoort; het keyboard; RS 232 seriele communicatie; joysticks; geheugen-uitbreidings connector en de seriele IEEE poort.

In het gedeelte over de cassette-poort wordt o.a. het lezen van en het schrijven naar de band behandeld. Ook is er een grafiek getekend, die de signalen laat zien die aan de uitgang van de cassette-poort staan als deze naar de cassettereorder schrijft alsmede de timing hiervan.

In het hoofdstukje over het keyboard wordt o.a. puntsgewijs uiteengezet wat er allemaal voor komt kijken wanneer het keyboard wordt afgescand. Ook is er een tekening geplaatst van hoe het keyboard in een matrix is onderverdeeld.

Bij de besprekking van de RS 232 seriele-interface komt o.a. het openen en sluiten van een RS 232 kanaal voor.



In het hoofdstuk over de Joystick wordt uitgelegd welke geheugen locaties men hiervoor moet gebruiken. Daarnaast is ook een testprogramma geplaatst, die al in de vorige "VIC-PRIMEURS" gepubliceerd is. Het gebruik van de potentiometer joystick ofwel de Paddle, wordt ook zeer uitvoerig besproken in ditzelfde hoofdstuk.

Bij de geheugen-uitbreidings connector worden alle pinnen op een duidelijke wijze besproken zoals de BLK's met daarachter hun begin-locaties. Verder zijn twee flowcharts geplaatst welke het "zenden" en het "ontvangen" m.b.v. de IEEE poort behandelen.

Dit waren zo in vogelvlucht de 5 secties waarin het boek verdeeld is. Ter afronding van deze besprekking volgt hieronder in het kort de inhoud van de aanhangsels waarmee het boek afsluit.

In aanhangsel 1 is een duidelijk overzicht gemaakt van alle CBM-codes. In aanhangsel 2 is een zogenaamd "Wedge Program" afdrukkt met een aantal grafische-instructies, waarna aanhangsel 3 de instructieset met de timing van de 6502 behandelt.

Aanhangsel 4 is een tabel voor conversie van Hexadecimaal-Decimaal.

Het schema van de VIC-20, in aanhangsel 5, is onderverdeeld in drie delen.

Circuit 1 bestaat in grote lijnen uit de 6502 en de 6561. Opgemerkt dient te worden dat het video-circuit welke afgebeeld staat, niet die van ons PAL-systeem is, maar die voor de 6560 zoals ook gedrukt is.

Circuit 2 bestaat uit het VIA gedeelte en de voeding; circuit 3 laat het geheugen gedeelte zien zoals de RAM's (2114), de Basic en de Kernel ROM (2364) zo ook de Karakter ROM (2332). Er kunnen nog enkele verschillen voorkomen, daar er zich enige kleine productie wijzigingen hebben voorgedaan, doch veel kan dit niet zijn.

In aanhangsel 6 is een overzicht gegeven van een groot aantal Machinetaal Monitor instructies met daarbij een korte omschrijving.

Een onthulling was dit boek zeker. Het is geen boek dat je zomaar in een adem kunt uitlezen want daar staat teveel informatie voor in waarover toch echt nagedacht moet worden. Daarom is het mooi dat het boek in vijf secties is onderverdeeld.

Voor degene die na het werken met het "Informatieboek voor programmeurs" op iets andere problemen stuit dan alleen maar in het software vlak, is dit boek een mooie combinatie.

G.R.

FORTH , de taal van de jaren '80



Wat te denken van een computertaal die gestructureerd , modulair , uitbreidbaar , kompakt en overdraagbaar werkt met de mogelijkheid om assembler met "high-level" kode te combineren en dat alles in minder dan 8,5 kilobytes

..... Oh ja , en inklusief de mogelijkheid voor "makro" assembler en "text-editor". Een droom? Nee, zeker niet - De taal is FORTH.

Alhoewel niet nieuw meer kent de taal , vooral na de oprichting van de FORTH Interest GROUP , een groeiende acceptatie. Zo'n drie duizend leden over de gehele wereld propageren het gebruik van FORTH. Vooral voor het kreeeren van behendigheidsspellen is FORTH , natuurlijk door zijn snelheid , uitermate geschikt.

FORTH werd in 1969 ontwikkeld door Charles H. Moore. Hij was in die tijd werkzaam bij de "National Radio Astronomy Observatory" in de amerikaanse staat Virginia.

De traagheid van programmeren kwam hem de strot uit. De standaard talen voor zakelijk gebruik , Linkertalen , Makrotalen , FORTRAN , COBOL , PL/1 enz.enz. moesten vervangen worden door een taal FORTH. De ontwikkeling duurde ruim tien jaar. Het vlotte langzaam en voorzichtig kwamen de ideeën tot ontwikkeling. Vooral doordat alle ideeën uit een brein voortkwamen , was het zo'n lange ontwikkelingsperiode best waard. Forth is niet zoals veel konventionele talen ingevroren ,

echter bijzonder flexibel enuitbreidbaar.Zodra je begint met programmeren in FORTH , verleng je automatisch het gebruik van de taal , waardoor nieuwe waarden en mogelijkheden ontstaan.Zelfontwikkeld datastrukturen kunnen worden toegevoegd. Zo ook de standaard opbouw van programma's.Deze mogelijkheid vindt men exclusief in FORTH , en in geen enkel ander programma.

FORTH is modulair van opbouw en volledig gestructureerd.Het programmeren bestaat uit het combineren van voorgedefinieerde - en zelfgekreeerde modulen ("words") , die zich opgaand ontwikkelen.Uiteindelijk zal het gekreerde programma bestaan uit.....een

woord."Debugging" is eenvoudig ; alle modulen kunnen , voor kombinatie , gekontroleerd en / of verbeterd worden.

Het gebruik van deze mogelijkheid heeft verstrekkende gevolgen . Indien u voornamelijk zakelijke programma's schrijft , dan zult u spoedig merken dat de zelfontwikkeld uitbreidingen puur zakelijk georiënteerd zullen zijn.Gaat u interesse uit naar "process control" , dan zal de ontwikkeling van het programma zich daar naar schikken.Sterker nog , beide ontwikkelingen kunt u combineren.

Dit houdt in dat u een bibliotheek van woorden zult ontwikkelen , gericht op uw specifieke kwaliteiten en dus gebruik.U zult ontdekken dat 80% van een nieuwe ontwikkeling terug te vinden is in de bestaande bibliotheek.Hierdoor bespaart u zichzelf veel

programmeertijd.

We zijn al ingegaan op de snelheid van FORTH.Deze snelheid wordt bepaald door "THREADED CODE", een krachtige toepassing van de taal.Een ontwikkelde FORTH Macro-assembler , bijvoorbeeld , geschreven in FORTH , gebruikt over het algemeen niet meer dan 1500 BYTES.Door het principe "THREADED CODE" zijn FORTH programma's over het algemeen korter dan programma's in machinetaal.Wel is FORTH 30 tot 70% langzamer dan machinetaal.BASIC , daarintegen is ongeveer 1000% langzamer. Stel dat jouw applicatie , in een tijds-kritieke fase , een verandering vereist , dan kun je gedeelten opnieuw koderen in assembler , waarbij je dan gebruik maakt van de bijgevoegde makro-assembler.Deze is zodanig gestructureerd dat "statements" als IF -ELSE -THEN(sic),BEGIN -REPEAT,BEGIN -UNTIL,BEGIN -WHILE -REPEAT direkt gebruikt kunnen worden.

Met FORTH assembler is het niet ongebruikelijk om al direkt gebruik te maken van de kodering.Modulen , in assembler ontwikkeld , worden gelijk aan andere FORTH woorden behandeld , standaard en zelfontwikkeld.Zelf de compiler zal het verschil niet ontdekken. Forth kan in ieder gewenste nummergeheugen werken , zonder dat het de snelheid zal beïnvloeden.Alle nummergeheugens , zoals VINARY , BINARY , TRINARY , OCTAL , DECIMAAL , UNIDECIMAAL , HEXADECIMAAL kunnen worden gebruikt. FORTH kent geen floating point , alhoewel het als een optie

verkrijgbaar is.

Voor wat betreft de overdraagbaarheid zijn alle FORTH systemen vrijwel identiek, waardoor het gebruik op een ander systeem geweldig eenvoud is. De FORTH INTEREST GROUP heeft listings gepubliceerd, waardoor de taal op vrijwel alle mikroprocesoren te gebruiken is (1802, 8080, 6502, 8086/8088, 6800, 9900, PDP-11, 6809, NOVA, en meer in ontwikkeling). De kosten voor zo'n listing bedragen fl.70,-. Wel moeten deze versies geinitialiseert worden, alvorens zij op een bepaald systeem werken.

FORTH gebruikt, in tegenstelling tot PASCAL en FORTRAN, weinig geheugen. Vooral bij gebruik op de VIC-20 computer is dit een groot voordeel, omdat de meeste programma's niet meer dan 16K geheugen gebruiken. Het wordt luxe bij gebruik van 32K RAM. Het gebruik van geheugen is over het algemeen dus geen probleem met FORTH.

Misschien vraag je je nu af, waar die plotselinge interesse in een taal, die meer dan tien jaar oud is. Een paar jaar geleden was deze taal niet eens bekend bij professionele computerprogrammeurs. Die paar mensen, die er wel mee werkten waren enthousiast en tevreden en door het verspreiden van deze ervaring d.m.v. de INTEREST GROUP kwam de taal in bredere kringen terecht. Uiteindelijk werd in het BYTE nummer van augustus 1980 een complete editie aan FORTH gewijd en begint de interesse naar deze taal sterk te groeien.

Atari, de spellenfabrikant, gebruikt FORTH in plaats van MACHINETAAL om spellen te ontwikkelen. Alle nieuwe ontwikkelingen op dit gebied met name in de amusementshallen worden geschreven in FORTH. Ook in science-fiction films worden animaties en speciale ruimtemodellen gestuurd door FORTH. Onlangs gebeurde dat in de film "Battle beyond the stars". Twee RCA satellieten worden door FORTH bestuurd en ook NASA maakt veelvuldig gebruik van deze taal. LISP en PASCAL "interpreters" worden geschreven in FORTH. FORTH is ontdekt en uiteindelijk geaccepteerd.

DATATRONIC AB, moedermaatschappij van HANDIC in Nederland, heeft de taal gebruikt om Gasbesturings machines te sturen. Een zeer nauwkeurig programma voor een verantwoordelijke controle. Ikzelf heb een avonturenspel in FORTH geschreven en heb ontdekt dat de taal voor iedere aanpassing geschikt is. Iedere applicatie kan in FORTH worden geprogrammeerd.

Waar kunt u FORTH kopen en ook meer informatie opvragen. Welnu, allereerst is het natuurlijk mogelijk om de listing van het programma op te vragen bij de reeds besproken INTEREST GROUP. Het juist adres is:
Forth Interest Group
P.O. box 1105
San Carlos CA 94070
CALIFORNIA
U.S.A.

Vergeet niet de juiste mikroprocessor op te geven.

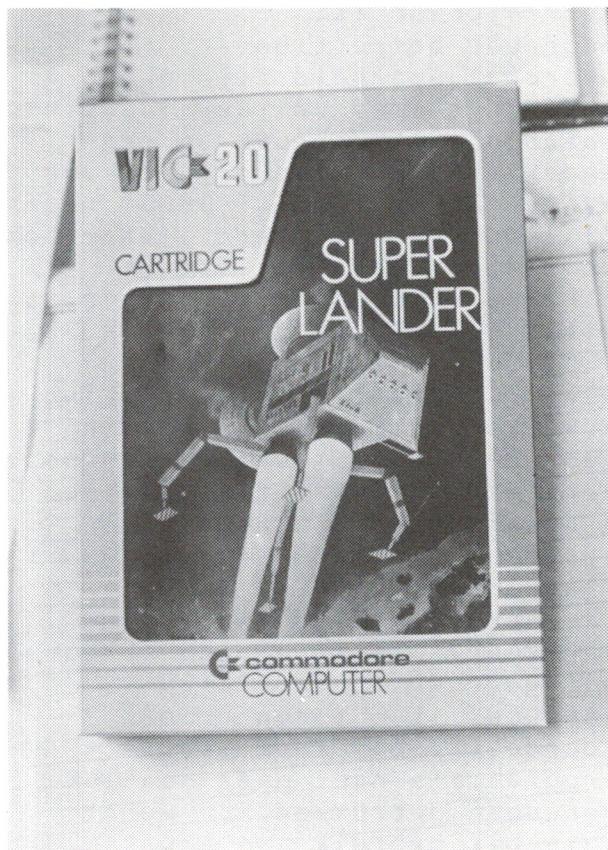
Maar je begrijpt natuurlijk wel , dat dit artikel niet in VIC-primeurs zou verschijnen als FORTH niet leverbaar zou worden voor de VIC-20.

Juist door die geweldige internationale acceptatie van FORTH heeft DATATRONIC AB besloten een speciale ROM cassette te ontwikkelen voor onze computer. In de eerste week van juli hoopt HANDIC de meeste dealers te kunnen bevoorraden met deze ROM cassette. De prijs kunt u elders in dit nummer terugvinden. De floating point cassette zal later verschijnen.

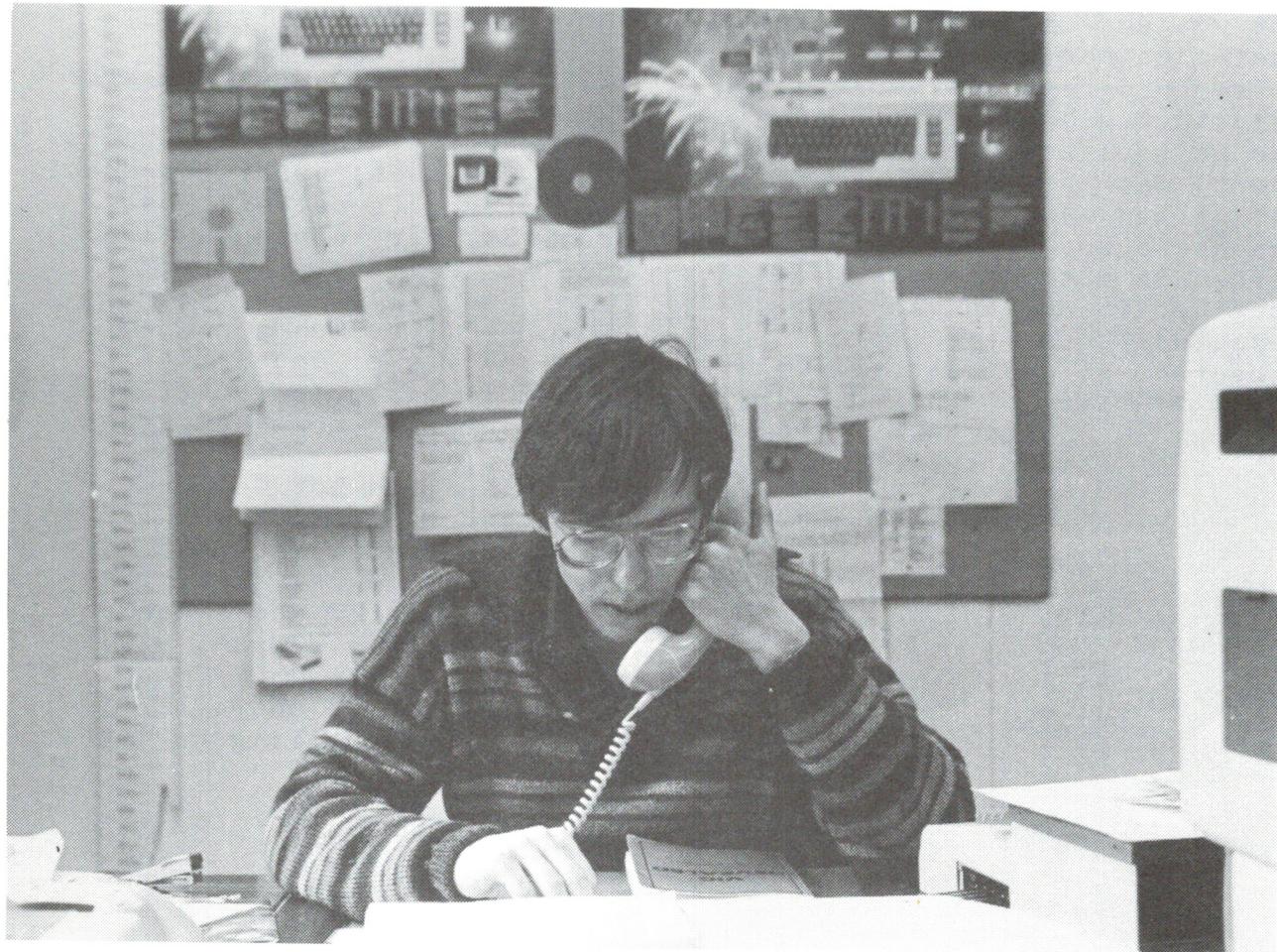
Het zal duidelijk zijn dat FORTH een bijzondere plaats inneemt tussen andere talen. Velen vragen zich zelfs af of FORTH als taal of operatingsystem bezien moet worden. Een ding is zeker ; alle wensen t.a.v. sneller en effektiever programmeren zijn in FORTH terug te vinden. Het gebruik is uniek en voegt een nieuwe dimensie toe aan de computer , waardoor iedere programmeur niet om FORTH heen kan.

Binnenkort op de VIC-20.

Dit artikel is vrij vertaald van een interview met Peter Bengsten , lid van de Interest Group en als programmeur werkzaam bij DATATRONIC AB. Onze dank daarvoor.



#####



Veel artikelen met name op het software vlak worden besproken door Onno Rietveld. Velen hebben hem gebeld om specifieke informatie te verkrijgen, omtrent de VIC-20 en of accessoires. Ook op deze foto ziet u Onno in aktie om een klant de nodige informatie te verschaffen. Onderstaand artikel is door Onno geschreven.

VIC-TIP DE FUNCTIE TOETSEN

Over het gebruik van de functietoetsen (de 4 grotere toetsen geheel rechts van het VIC-20 toetsenbord) bestaat wat onduidelijkheid. Ook is niet duidelijk waar ze nu precies beschreven worden. Het 'INFORMATIE BOEK VOOR VIC-20 PROGRAMMEURS' bevat bijvoorbeeld geen informatie over deze functietoetsen.

De 'insteekkaarten' die bedoeld worden in de beschrijving van de functietoetsen in de gebruiksaanwijzing (pag. 20) zijn de VIC-1211 (de SUPER EXPANDER) en de VIC-1212 (de PROGRAMMERS-AID). Wanneer men

gebruik maakt van deze uitbreidingen is het mogelijk met de dan toegevoegde opdracht KEY de functietoetsen een bepaalde tekst toetewijzen die uitgeprint zal worden zodra die functietoets ingedrukt wordt. Het is dan bijvoorbeeld mogelijk door het indrukken van de F4-toets 'LIST100-' op het scherm te zetten en er kan dan ook een <return> aan toegevoegd worden zodat 'LIST100-' direct uitgevoerd zal worden. Dit is alleen mogelijk buiten een programma om. KEY kan niet in een programma gebruikt worden.

Het volgende programmavoorbeeld laat zien hoe de functietoetsen in een programma te gebruiken zijn. Het programma zelf is niet van zo'n bijzonder interessante soort, maar het dient vooral om het werken van de toetsen te demonstreren. Het kan toegepast worden in elk programma waar door het indrukken van een bepaalde toets iets moet gebeuren. De speciale tekens in regel 130 t/m 145 hebben resp. betrekking op CTRL BLK, CTRL RED, CTRL CYN, CTRL BLU.

Korte programmatoelichting:

REGEL 100 : hier wacht de VIC-20 totdat er een toets wordt ingedrukt. GET A\$ kijkt steeds of er een toets wordt ingedrukt. A\$ is leeg als dat niet het geval is, dit kunnen we zien door A\$ te vergelijken met "".

REGEL 120 : we nemen nu de ASC-code van de ingedrukte toets, is dit geen functietoets dan printen we deze boodschap.

REGEL 130-: is het wel een functietoets dan gaan we testen welke dat dan wel geweest is en printen dit op het scherm. Bij F1, F3, F5 en F7 wordt dan ook nog wat extra's gedaan. U kunt natuurlijk achter het THEN statement iedere opdracht invullen die u wilt gebruiken.

```

50 PRINT "D"
60 TS$="FUNCTIE TOETS ";P=25
100 GET A$:IF A$=""THEN 100
120 A=ASC(A$):IF A<133 OR A>140 THEN PRINT "GEEN FUNCTIE TOETS":GOTO 100
130 IF A=133 THEN PRINT TS$;" 1":P=P+1:GOTO 200
135 IF A=134 THEN PRINT TS$;" 3":GOTO 300
140 IF A=135 THEN PRINT TS$;" 5":GOTO 350
145 IF A=136 THEN PRINT TS$;" 7":P=25:GOTO 200
160 IF A=137 THEN PRINT TS$;" 2"
165 IF A=138 THEN PRINT TS$;" 4"
170 IF A=139 THEN PRINT TS$;" 6"
175 IF A=140 THEN PRINT TS$;" 8"
190 GOTO 100
199 REM VERANDER KLEUR
200 IF P>255THEN P=0
210 IF INT(P/8)=(P/8) THEN P=P+8
220 POKE 36879,P
240 GOTO 100
299 REM SET GELUID AAN
300 POKE 36878,15:POKE36876,135
310 PRINT "GELUID AAN"
320 GOTO 100
349 REM SET GELUID UIT
350 POKE 36878,0:POKE36876,0
360 PRINT "GELUID UIT"
370 GOTO 100

```



Hoge-resolutie op de VIC-20.

DEEL 2:

hoge-resolutie en zelfgedefinieerde karakters met 8K en/of 16K uitbreiding.

Zowel in het artikel over zelfgedefinieerde karakters als in het artikel over hoge-resolutie uit de vorige 'VIC PRIMEURS' werden programma's beschreven voor een standaard VIC-20 of met een 3K RAM-uitbreiding. In dit artikel komen ongeveer dezelfde programma's aan de orde maar dan geschikt voor een VIC-20 met een 8K en/of een 16K RAM-uitbreiding. In het volgende nummer van 'VIC PRIMEURS' gaan we in deze serie een programma beschrijven voor hoge-resolutie dat op iedere geheugenversie kan werken.

Eerst iets over het geheugen van de VIC-20 en het gebruik daarvan bij hoge-resolutie en zelfgedefinieerde karakters. Het standaard 5K-geheugen van de VIC-20 loopt van locatie 4096 t/m 8191 en met een 3K RAM-uitbreiding van 1024 t/m 8191. De gegevens voor het beeldscherm worden bewaard van 7680 t/m 8191. Bij het gebruik van zelfgedefinieerde karakters worden deze in het gebied 7168 t/m 7679 opgeslagen of bij hoge-resolutie wordt 5120 t/m 7679 gebruikt. Dit betekent dat het einde van het BASIC-werkgeheugen lager komt te liggen. Dit doen we door in locatie 56 te POKEn bij de voorgaande programma's.

Werken we met een 8K en/of 16K RAM-uitbreiding dan wordt het

anders. Het BASIC-werkgeheugen loopt dan vanaf 4608 tot 16384 (8K) of 24576 (16K) of 32768 (8K en 16K). Het is nu niet mogelijk om het karakter of hoge-resolutie geheugen bijvoorbeeld te verplaatsen naar 23552 t/m 24576, daar de VIC dit gedeelte alleen maar kan gebruiken tussen 4096 t/m 8191.

Hoe nu met een 8K en/of 16K RAM-uitbreiding? In plaats van verlagen van de top van het BASIC-werkgeheugen, kunnen we ook het begin hiervan een stuk naar boven schuiven, zodat voor het BASIC-programma ruimte vrij komt voor de karakters. Het enigste verschil hiervan ten opzichte van achter het BASIC-programma is dat het niet door het programma zelf gedaan kan worden.

De indeling van het geheugen wordt nu als volgt. Eerst het schermgeheugen van 4096 t/m 4607 dan het karakter of hoge-resolutie geheugen van 5120 t/m 7167 en tot slot het BASIC-geheugen vanaf 7168 (inplaats van eerst dus 4608) t/m 16384/24576/32768. Het kleurengeheugen blijft van 37888 t/m 38399 en het stukje 4608 t/m 5119 wordt niet gebruikt.

We doen dit allemaal als volgt. Geef direct na het inschakelen van de VIC-20 de volgende opdrachten:

POKE44,28:POKE7168,0:NEW

Waarom ? Poke44,28 verhoogt het begin van BASIC naar 7168. Poke 7168,0 zet daar een nul in daar de eerste locatie van het BASIC-werkgeheugen altijd een nul moet bevatten. Dan NEW om alle andere noodzakelijk pionters voor het BASIC op hun nieuwe juiste plaats te zetten. Zolang we nu met zelfgedefinieerde karakters of hoge-resolutie werken hoeven we dit niet meer te herhalen. Nu kan het programma geschreven worden, geSAVED of geladen worden etc. Zijn we hiermee echter klaar is het verstandig de VIC-20 eenmaal aan- en uitteschakelen en nog beter is even SYS64802 te geven.

En nu dan nog de POKE opdracht voor het overschakelen naar zelfgedefinieerde karakters of hoge-resolutie, doe dit met :

POKE36869,205

Terugschakelen naar normale karakters kan met POKE36869,192.

Dan volgt nu eerst de listing van het programma op pag. 10 van de vorige 'VIC PRIMEURS' maar dan met 8K en/of 16K uitbreiding. De toelichting staat verder in het programma.

```
5 IFPEEK(44)=028 THEN LIST-25
10 REM EERSTE BEGIN BASIC
15 REM VERANDEREN MET
20 REM POKE44,28:POKE7168,0:NEW
25 REM DAARNA PROGRAMMA INLADEN
30 REM BEGINADRES NIEUWE TEKENSET
40 AD=5120
50 REM DATA ALLE 255 TEKENS OVERBRENGEN
60 FORI=0TO2047
70 POKEAD+I,PEEK(32768+I)
75 PRINT I;""
80 NEXT
90 REM DATA VAN DE S DOOR NIEUWE VERVANGEN
100 FORI=0TO7:READJ:POKEAD+19*I,J:NEXT
110 REM NIEUWE TEKENSET AKTIEVEREN
120 POKE36869,205
130 PRINT "":PRINT"S"
140 REM DATA NIEUW TEKEN
150 DATA126,66,32,16,32,66,126,0
160 REM TERUGSCHAKELEN MET POKE36869,192
```

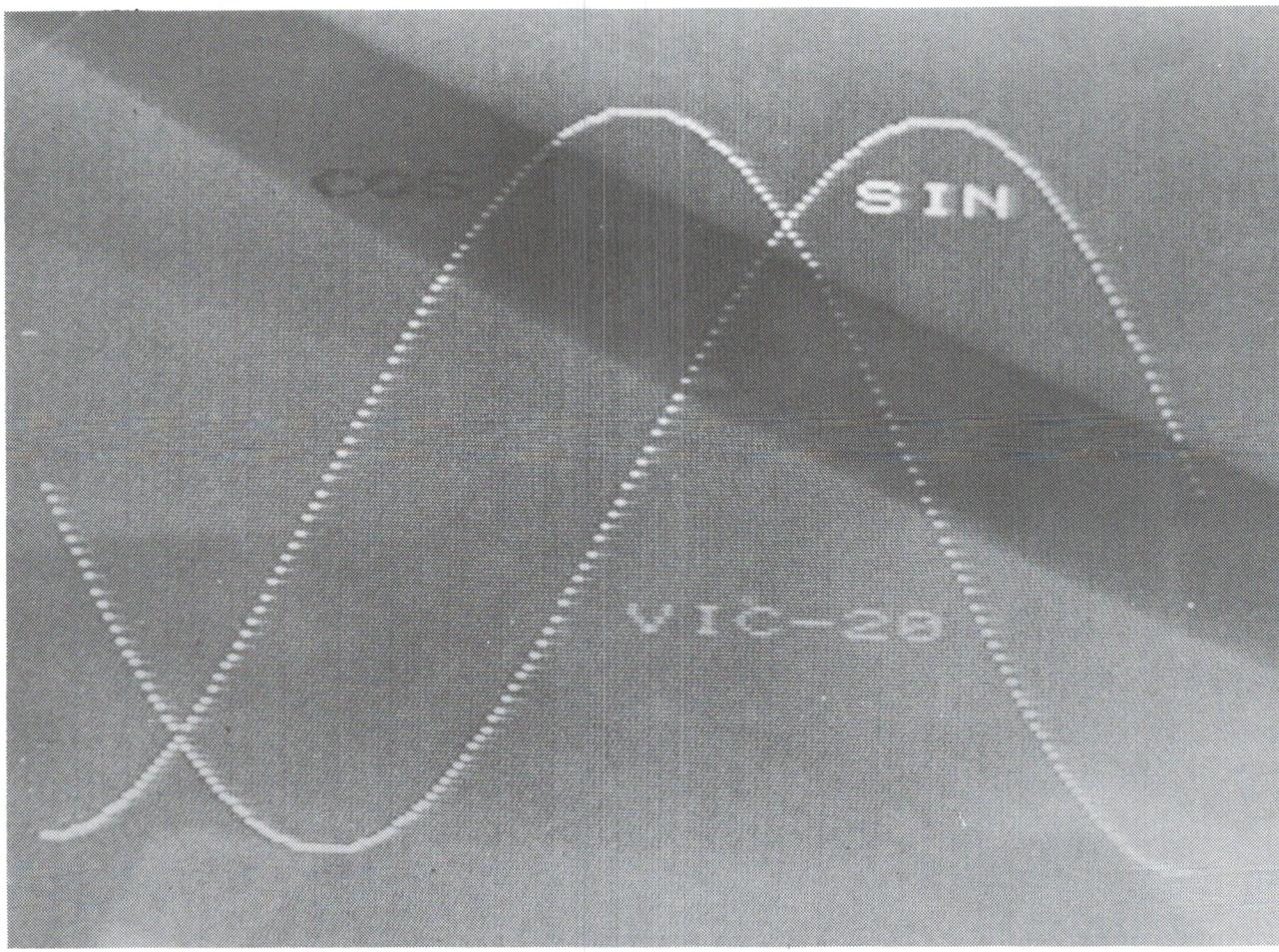
Het is nu mogelijk om 255 karakters te defineren. Karakter 0 t/m 127 zijn de normale karakter volgens de tabel op pag. 141,142 set1 van de gebruiksaanwijzing. De nummers 128 t/m 255 zijn de diapostieve karakters (129 is bijvoorbeeld diapostief A). U kunt hier elk ander karakter voor indeplaats zetten (ook niet diapostitief dus) en deze op het scherm krijgen door dan bijvoorbeeld PRINT"(rvs on)A" te geven. Verander eens 19 in regel 100 door 129 en print een diapositieve A (regel 130).

Tot slot een voorbeeld van hoge-resolutie met 8K en/of 16K RAM-uitbreidig. Het principe is hetzelfde als in het artikel over hoge-resolutie, alleen de regelnummers zijn iets veranderd en een deel van de karakters worden direct gewist in plaats van tijdens het programma (o.a. regel 205). Hier volgt de listing :

```

5 IFPEEK(44)<>28THENLIST-60
10 REM HIRES VOOR 8/16K
15 REM LET OP !
20 REM GEEF ALTIJD      VOOR HET LADEN OF IN-TYPEN
30 REM VAN DIT PROGRAMMA
40 REM EERST
50 REM POKE44,28:POKE7168,0:NEW
60 REM
100 PRINT"LEES NU TABEL IN RAM"
110 POKE 36879,42
150 POKE 36869,205
180 S=32768:T=5120
190 PRINT"LEES NU TABEL IN RAM"
200 FORI=0TO127*8+7:POKEI+T,PEEK(I+S):NEXTI
205 FORI=128*8TO255*8+7:POKEI+T,0:NEXTI:GOTO 300
210 REM SUBROUTINE
220 X%=X/8:Y%=Y/8:P=X%+Y%*22+4096: REM LET OP NIEUWE SCHERM WAARDE
230 Q=PEEK(P):IFQ<128THEN CN=CN+1:Q=127+CH:POKE P,Q
240 C=5120+Q*8+(YAND7):POKEC,PEEK(C)OR(2↑(7-(XAND7))):RETURN
300 PRINT "D"
310 FORX=1TO175:Y=92+SIN(X/28)*92:GOSUB220:NEXT
330 FORX=1TO175:Y=92+COS(X/28)*92:GOSUB220:NEXT
340 PRINT"SinusCosinusSIN"
345 PRINT"VIC-20"
350 GOTO 350

```



Het resultaat van een poging tot het maken van een scherm-foto van het hoge resolutie programma op de vorige pagina.

GEBRUIKSAANWIJZINGEN 3K, 8K en 16K RAM-cassettes

Voor de 3K, 8K en 16K RAM-geheugenuitbreidingen is nu de nederlandse gebruiksaanwijzing (eindelijk) uitgekomen. Tot voor kort werden de RAM-cassettes geleverd met een verkorte engelstalige instructie. Hieronder drukken wij de gebruiksaanwijzingen voor deze cassettes af.

VIC-1210 3K RAM-cassette

De VIC-1210 3K RAM-cassette kan direct achter in de VIC-20 of in een uitbreidings-module geplaatst worden.

Direct in de VIC-20:

1. Schakel de VIC-20 uit.
2. Druk de 3K RAM-cassette voorzichtig in de uitbreidings-poort achter in de VIC-20. Let op dat het label naar boven staat als u de cassette aanbrengt.
3. Schakel de VIC-20 aan. De volgende mededeling verschijnt nu op het scherm:

****** CBM BASIC V2 ******

6655 BYTES FREE

READY.

In een uitbreidings-module:

De VIC-1210 3K RAM-cassette kan in een van de aansluitbussen van de uitbreidings-module geplaatst worden. Voordat u de cassette plaatst, verzekert u ervan dat zowel de VIC-20 als de uit-

breidings-module uitgeschakeld zijn. Druk dan de 3K RAM-cassette voorzichtig in een van de aansluitbussen en schakel het systeem aan.

Er kan slechts één 3K RAM-cassette tegelijkertijd in de uitbreidings-module geplaatst worden.

Bij gebruik van de 3K RAM-cassette samen met een 8K- en/of 16K RAM-cassette in een uitbreidings-module, zal de 3K RAM-cassette niet door de BASIC gebruikt worden.

Zie voor meer informatie de gebruiksaanwijzing van uw uitbreidings-module.

Het is niet mogelijk het scherm- of karaktergeheugen naar deze geheugen-uitbreiding te verplaatsen.

Als u de 3K RAM-cassette gebruikt, wordt het beginadres van het BASIC programmagegeheugen verplaatst naar \$0400 (1024).

Laat de RAM-cassette niet vallen en voorkom aanraking van de metalen contacten!

N.B.

Elke wijziging aan deze RAM-cassette zonder voorafgaande toestemming van de importeur, doet de garantie vervallen.

VIC-1111 16K RAM-cassette

De VIC-1111 16K RAM-cassette kan direct achter in de VIC-20 of in een uitbreidings-module geplaatst worden.

Direct in de VIC-20:

1. Schakel de VIC-20 uit.
2. Druk de 16K RAM-cassette voorzichtig in de uitbreidings-poort achter in de VIC-20. Let op dat het label naar boven staat als u de cassette aanbrengt.
3. Schakel de VIC-20 aan. De volgende mededeling verschijnt nu op het scherm:

****** CBM BASIC V2 ******

19967 BYTES FREE

READY.

In een uitbreidings-module:

De VIC-1111 16K RAM-cassette kan in een van de aansluitbussen van de uitbreidings-module geplaatst worden. Voordat u de cassette plaatst, verzekert u ervan dat zowel de VIC-20 als de uit-

breidings-module uitgeschakeld zijn. Druk dan de 16K RAM-cassette voorzichtig in een van de aansluitbussen en schakel het systeem aan.

Er kan slechts één 16K RAM-cassette tegelijkertijd in de uitbreidings-module geplaatst worden.

Zie voor meer informatie de gebruiksaanwijzing van uw uitbreidings-module.

Als u de 16K-cassette gebruikt begint het schermgeheugen op \$1000 (4096) en het kleurengeheugen op \$9400 (37888).

Het is niet mogelijk het scherm- of karaktergeheugen naar deze geheugen-uitbreiding te verplaatsen.

Als u de 16K RAM-cassette gebruikt, wordt het beginadres van het BASIC programmagegeheugen verplaatst naar \$1200 (4608).

Laat de RAM-cassette niet vallen en voorkom aanraking van de metalen contacten!

N.B.

Elke wijziging aan deze RAM-cassette zonder voorafgaande toestemming van de importeur, doet de garantie vervallen.

VIC-1110 8K RAM-cassette

De VIC-1110 8K RAM-cassette kan direct achter in de VIC-20 of in een uitbreidingsmodule geplaatst worden.

Direct in de VIC-20:

1. Schakel de VIC-20 uit.
2. Druk de 8K RAM-cassette voorzichtig in de uitbreidings-poort achter in de VIC-20. Let op dat het label naar boven staat als u de cassette aanbrengt.
3. Schakel de VIC-20 aan. De volgende mededeling verschijnt nu op het scherm:

**** CBM BASIC V2 ****

11775 BYTES FREE

READY.

In een uitbreidings-module:

De VIC-1110 8K RAM-cassette kan in een van de aansluitbussen van de uitbreidings-module geplaatst worden. Voordat u de cassette plaatst, verzekert u ervan dat zowel de VIC-20 als de uitbreidings-module uitgeschakeld zijn. Druk dan de 8K RAM-cassette voorzichtig in een van de aansluitbussen en schakel het systeem aan.

Als u de 8K-cassette gebruikt begint het schermgeheugen op \$1000 (4096) en het kleurengeheugen op \$9400 (37888).

Het is niet mogelijk het scherm- of karaktergeheugen naar deze geheugenuitbreiding te verplaatsen.

Als u de 8K-cassette gebruikt, wordt het beginadres van het BASIC programmagengeheugen verplaatst naar \$1200 (4608).

Laat de RAM-cassette niet vallen en voorkom aanraking van de metalen contacten!

N.B.

Elke wijziging, behalve juist gebruik van de DIP-schakelaar, aan deze RAM-cassette zonder voorafgaande toestemming van de importeur, doet de garantie vervallen.

Als u gebruik maakt van meer dan één 8K RAM-cassette of u gebruikt een 8K RAM-cassette en een 16K RAM-cassette in een uitbreidings-module dan moet de geheugenlocatie van de 8K

RAM-cassette verplaatst worden met de in de cassette aangebrachte DIP-schakelaar.

Open eerst de RAM-cassette:

1. Verwijder de schroef achterin de cassette met een 'sterschroevendraaier'.
2. Open nu de cassette door een schroevendraaier in de 2 gleufjes achterop de cassette te plaatsen en deze voorzichtig heen en weer bewegen. Draai hierna de schroevendraaier links en rechts van de print-connector een kwartslag.

Op de print ziet u in het midden boven de print-connector de DIP-schakelaar, die het geheugen-adres bepaald.

De standen van de DIP-schakelaar:

1	2	3	4
1 2 3 4 - OFF -			
\$2000 -3FFF	\$4000 -5FFF	\$6000 -7FFF	\$A000 -BFFF

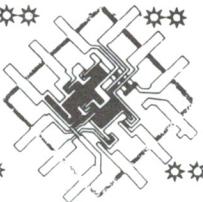
Schakel nooit meer dan één schakelaartje tegelijkertijd aan.

Normaal staat de 8K RAM-cassette geschakeld voor het BASIC bereik \$2000-3FFF. Gebruikt u een tweede 8K RAM-cassette schakel deze dan naar \$4000-5FFF. Voor drie 8K RAM-cassettes of een 16K RAM-cassette en een 8K RAM-cassette moet de laatste naar het bereik \$6000-7FFF geschakeld worden. Het bereik \$A000-B000 is een ROM gebied en kan niet voor BASIC gebruikt worden.

Gebruik nooit twee 8K RAM-cassettes voor hetzelfde geheugenbereik!

Zie voor meer informatie de gebruiksaanwijzing van uw uitbreidings-module.

***** COPY TRONICS | | | | PERSBERICHT | | | | *****



PBE/VIC Basic cursus

29-3-'82

1

Voor VIC gebruikers heeft de PET Benelux Exchange (de grootste gebruikersgroep van PET/CBM/VIC computers in de Benelux) een eenvoudige, Nederlandstalige Basic cursus beschikbaar op cassette. In een tiental lessen wordt iedereen vertrouwd gemaakt met de meest voorkomende Basic begrippen. Alle lessen staan op zichzelf en kunnen (eindeloos) worden herhaald, terwijl vanuit een les de volgende automatisch wordt geladen door het indrukken van een toets. Aan het eind zijn een drietal programmavoorbeelden opgenomen. De lessen hebben de volgende inhoud:

VIC Basic intro:	een grafische demonstratie
Les 01	: load; new; run/stop; direct mode; rekenen; ready.
Les 02	: rekenregels; regelnummers; syntax error; dubbele punt-, komma-, punt-komma functies
Les 03	: commando's; aanhalingsstekens; haakjes bij rekenkundige functies
Les 04	: haakjes en rangvolgorde (vervolg); input; stringvariabelen
Les 05	: stringvariabelen (vervolg); if-goto; if-then list; get
Les 06	: variabelen; enz
Les 07	: strings
Les 08	: string statements (asc en chr\$)
Les 09	: commando's; list; run; for-next loop; array variabele; dimensie
Les 10	: data statements; restore; step; terugtellen; matrix; nested loop; subroutines; if-then voorwaarden
Besluit	: tips, adres
Voorbeelden	: animatie, gemiddelde waarde, synoniemen.

Alle lessen werken op een standaard VIC zonder geheugenuitbreiding. Deze cursus op cassette is te bestellen, door een C60 cassette op te sturen en tegelijk fl. 25,- over te maken op girorekening 4088944 t.n.v. Copytronics, Deventer. U mag ook een giro- of betaalkaart meezendende met uw cassette. Sluit een gefrankeerde retourenveloppe (2 postzegels van 70 cent) bij, voorzien van uw naam en adres.

Inl.: Copytronics, Burg. v. Suchtelenstraat 46, 7413 XP Deventer (05700) 31895.

VIC-TIP 3_SYSies

Hierinnder volgen drie SYS-opdrachten die alledrie met het scherm te maken hebben.

1. Met de volgende POKE's en SYS is het mogelijk de cursor naar een bepaalde plaats op het beeldscherm te bewegen. Daar kan dan bijvoorbeeld iets gePRINT worden. ln is het regelnummer en loopt van 0 t/m 22, ps is de positie op die regel en loopt dus van 0 t/m 21.

POKE781,ln:POKE782,ps:SYS65520

Zo zet bijvoorbeeld;

POKE781,11:POKE782,11:SYS65520:PRINT"HALLO"

de tekst 'HALLO' op regel 12 in het midden.

2. Nu kunnen we deze regel of iedere andere schermregel leegmaken met de voldende opdracht

POKE781,ln:SYS60045

Nu kunnen we dus met;

POKE781,11:SYS60045

regel 12 uit voorgaande voorbeeld weer wissen.

3. Heeft u het te bont gemaakt op het scherm met kleur, grafische toestanden en wilde geluids-effecten, dan is het indrukken van de toetsen RUN/STOP en RESTORE vaak een uitkomst. Het programma kan dit ook zelf doen met de volgende opdracht

SYS 65234

en de rust is weergekeerd (u kunt deze opdracht bijvoorbeeld opnemen voor het END statement in het programma).

DEALERLIJST:

A

Aalten	Erba b.v.	05437- 2351
Afferden	Macom	08853- 1647
Alkmaar	TOS computercentrum	072-156540
Alkmaar	Matsunaga	072-129337
Alkmaar	Fa. v. Kampen	072-116569
Alkmaar	Sibo	072-114929
Alkmaar	Fa. v. d. Gragt	072-126046
Alkmaar	Bakker Dijk b.v.	072-114268
Alphen a/d Rijn	Groen Stereo	01720- 73083
Amersfoort	Morelisse	033- 16052
Amstelveen	Radio Valkenberg	020-432470
Amsterdam	Fa. v. Campen	020-191953
Amsterdam	Radio Vos	020-736154
Amsterdam	Attent Electra	020-934006
Amsterdam	Compu 2000	020-360903
Amsterdam	Fa. v. Kampen	020-181907
Amsterdam	Radio Valkenberg	020-184022
Amsterdam	A.R.S. Elopta	020-251922
Amsterdam	Auroro	020-264644
Amsterdam	Radio Holland	020-101972
Amsterdam	Radio Rotor	020-125759
Amsterdam	Wolfkamp	020-278931
Amsterdam	Computercollectief	020-223573
Almelo	Electronicahuis	05490- 19191
Almelo	Lee de Ruijter	05490- 18648
Almelo	Abbink	05490- 60461
St. Annaparochie	Radio Kampen	05180- 1906
Apeldoorn	v. Essen Radio	055-212485
Apeldoorn	Radio Putto	055-214106
Appelscha	Radio Oldersma	05162- 1591
Arnhem	Hupra	085-426716
Arnhem	Te Kaat	085-454518
Arnhem	Telemark	085-456838
Asten	Fa. Jeuken	04936- 3388
Assen	Audio Service Marree	05920- 13034
Assen	T.S.C.	05920- 17787
Arkel	M.C.P. b.v.	01831- 1566

B

Balk	Fa. Haantjes	05140- 2395
Beilen	Fa. de Groot	05930- 3839
Bergen op Zoom	Fa. Rein de Jong	01640- 36028
Beverwijk	Radiodokter	02510- 26292
Beverwijk	Wijkerbaan Autobedrijf	02510- 43193
Beverwijk	Westerveld	02510- 24150
Boekel	Fa. v. d. Broek	04922- 2207
Born	Fa. Wibo	04498- 51248
Den Bosch	Desiree Camp	073-138323
Den Bosch	Ben v. Dijk Electro	073-216232
Boxmeer	Fa. Repkes	08855- 3563
Breda	Elgam b.v.	076-224908
Breda	Indelec	076-142333
Breda	Breda Electra	076-226459
Breda	Fa. Cohen	076-134462
Den Burg	v. Wijngaarden	02220- 2695
Bussum	Radio Velt	02159- 17315
Bussum	Computerhouse Bussum	02159- 33700

C

Cuyk	Rutten	08850- 16344
------	--------	--------------

D

Damwoude	Fa. v. d. Galiën	05111- 1396
Delft	Electr. Centrum Delft	015-134429
Delft	E.H.S.	015-132234
Delft	Radio Reno	015-132194
Deurne	Electro Mennen	04930- 9389
Dokkum	Electr. Terpstra	05190- 4000
Doorn	Autoshop De Eend	03430- 4076
Doornspijk	Fa. v. Zeeburg	05258- 355
Dordrecht	Radiobeurs Louter b.v.	078-134918
Drachten	Fa. v. d. Meulen	05120- 12352
Drachten	T.V. Technische Dienst	05120- 17541
Drunen	Dekkers b.v. Radio & T.V.	04163- 5833

E

Ede	Sisas Holland	08380- 38075
Ede	Eylander	08380- 17548
Eindhoven	Fa. Bombeek	040-441834
Eindhoven	Ryers b.v.	040-422888
eindhoven	Compu 2000	040-445255
Eindhoven	Vogelzang	040-447955
Elburg	Bouwman radio	05250- 3777
Emmen	Crescendo	05910- 13580
Emmen	E.H.C. Micronics	05910- 13859
Enschede	Radio Nijhuis	053-315169

F

Franeker	Fa. Tinga	05170- 2525
Franeker	Fa. Ettema	05170- 2090

G

Gaanderen	Centen Electra	08350- 7241
Geleen	Cuvos Electra	04494- 47709
Geldermalsen	Graham b.v.	03455- 2341
Genemuiden	L. Roetman	05208- 1389
Gennep	v. Veenendaal	08851- 3334
Goes	IMHA	01100- 13941
Gorkum	Fa. Sommer	08130- 32644
Goor	Fa. Vrielink	05470- 4050
Gouda	Willems Expert	01820- 15248
's Gravenzande	Fa. Koenen	01748- 3136
Groningen	A.V.A. Autoshop	050-128066
Groningen	Vorstenberg comm. center	050-121524
Groningen	Telec b.v.	050-129374

H

Haaksbergen	Fa. v. Ulsen	05427- 1276
Den Haag	Quality Sellers	070-888844
Den Haag	Stuut en Bruin	070-604993
Den Haag	Rueb b.v.	070-559919
Den Haag	Radio Gerrése	070-455426
Haarlem	Fa. Kiekens	023-320485
Haarlem	Fa. v. Campen	023-311965
Haarlem	Blue Cat	023-327236
Haarlem	Computerhouse Haarlem	023-314032
Haarlem	Display Electronica	023-322421
Hilversum	Computerworld	035- 12633
Harderberg	Fa. Oostenbrink	05232- 1720
Harderwijk	Ridero	03410- 12984
Harlingen	Fa. v. d. Veen	05178- 2391
Heemskerk	Fa. v. Campen	02510- 42919
Hengelo	Radio Nijhuis	074-917567
Hengelo	Computershop Hengelo	074-428726
Heerde	Veron Electr.	05782- 1540
Heerhugowaard	Fa. Leegwater	02207- 12309
Heerlen	Vogelzang	045-716055
Heiloo	Bakker Radio	072-330262
Den Helder	Elab component supply service	02230- 30375
Den Helder	Fa. Proton	02230- 19068
Den Helder	Fa. Sibo	02230- 13310
Helmond	Fa. Westerhof	04920- 46680
Hillegom	Kall-Tronic	02520- 15605
Hoek van Holland	Electro Holland	01747- 4819
Hoofddorp	W + L Automatisering	02503- 31890
Hoofddorp	Fa. Gehrels	02503- 14965
Hoogeveen	Doeven Electronica	05280- 72221
Hoorn	Sibo	02290- 17763
Hoorn	Video 2000	02290- 14779
Hulst	Elton Autoshop	01140- 12261
Hulst	Colijn Data Systemen	01147- 834

J

Joure	Radio Rijpkema	05138- 2656
-------	----------------	-------------

K

Kampen	Elcatronic	05202- 11671
Krabbendijke	Fa. Goud	01134- 2010
Krommenie	Microcomputing	075-211619
Kloetinge	Colijn Data Systemen	01100- 14008

L

Langezwaag	Fa. v. d. Molen	05130- 88618
Lekkerkerk	Electro Slager	08105- 1869
Lichtenvoorde	Fa. Krabbenborg	05443- 2713
Lieren	Fa. v. d. Kamp	05766- 1744
Lisse	Radio Beurs	02521- 12176
Leiden	L.C.L.	071-125700
Leiden	Fa. de Groot	
Leiden	Vlasveld Electronica	071-120848
Lelystad	Rooymans	03200- 27527
Lelystad	Micron Electronics	03200- 44830

M

Maastricht	Vogelzang	043- 14169
Maastricht	Fa. Zeguers	043- 32072
Meerssen	T.V. Deusing	043-462079

N

Nijkerk	v. d. Klok Geurtsen	03494- 51382
Nijmegen	Fa. v. d. Camp	080-442747
Nijmegen	Fa. Baas Esso Service	080-441663
Nijmegen	Fa. v. d. Water	080-554182
Nijmegen	v. d. Broek	080-774322
Nijverdal	Doornbal b.v.	05486- 12297
Nijverdal	Radio Vo	05486- 12728
Nistelrode	Ben. v. Dijk	04124- 1503
Nijmegen	Macom Systems	080-553297
Nijmegen	Video Nijmegen	080-224820

O

Oosterhout	Fa. Peeters	01620- 53612
Oss	Ben v. Dijk Electro	04120- 33295

P

Purmerend	Radio Valkenberg	02990- 20727
-----------	------------------	--------------

R

Raalte	Beekman Electra	05720- 2511
Reusel	Rivago Micromix	04976- 1979
Ridderkerk	De Malle Molen	01804- 25782
Roermond	Byte Electron	04750- 10250
Roermond	Fa. Zeguers	04750- 14479
Roosendaal	H. & B Techn. Buro	01650- 34171
Rotterdam	Dil Electronics	010-854213
Rotterdam	Compu 2000	010-117524
Rotterdam	Delmee Autoshop	010-844304
Rotterdam	VIC Computercentrum	010-137823
Rotterdam	Radio Ultra	010-191680
Rotterdam	DCS Electronic	010-769900
Rotterdam	F.G. Elctronics	010-145553

S

Sassenheim	Fa. Duynstee	02522- 10398
Schagen	Sibo	02240- 8383
Scherpenzeel	Erik Jonker Electro	03497- 1321
Schiedam	Radio v. d. Bend	010-267568
Sittard	Fa. Harmens	04490- 16658
Sittard	Fa. Wibo	04490- 13070
Sneek	Blom Radio	05150- 13383
Someren	Fa. v. Ottendijk	04937- 1204
Stadskanaal	Leo Elektronics	05990- 19004
Stadskanaal	Commix	05990- 16655
Steenwijk	Gebr. v. d. Velde	05210- 13583

T

Tholen	Quist Duine	01660- 2505
Tiel	Fa. Rieuwers	03440- 13919
Tiel	T.V. Servicedienst Tiel	03440- 13907
Tilburg	Nico v. d. Braak	013-432153
Tilburg	Fa. Rosemeisl	013-323049
Tilburg	Mitchel Electronics	013-360848
Tilburg	N. v. Helfteren	013-670906
Tilburg	M. v. Helfteren	013-355605

U

Uden	Ben v. Dijk Electro	04132- 64944
Uithuize	Lambeek	05953- 1298
Utrecht	Display Electronics	030-315655
Utrecht	Radio Centrum	030-319636

V

Varsseveld	Visscher	08352- 2749
Veenendaal	Hupra	08385- 24222
Veghel	v. Aalst Electronic	04130- 41370
Venlo	Fa. Bauer	077- 17154
Vlaardingen	Radiohuis v. d. Bend	010-342481
Vlaardingen	Fa. Molenaar	010-341516
Vlissingen	Dijkhuizen	01184- 12981
Volendam	Hi-Fi Volendam	02993- 65451
Voorthuizen	Fa. v. Loon	03429- 1359

W

Waalwijk	Fa. Dekkers	04160- 34836
Wageningen	Fa. Mateman	08370- 12444
Weert	Lako b.v.	04950- 34164
Weert	Harthold Weekers	04950- 33392
Wierden	Engberts & Olthuis	05496- 1565
Wierden	Fa. v. Zoonen	02272- 1232
Winschoten	Fa. Oost	05970- 14234
Wommels	Fa. Kooistra	05159- 1341

IJ

IJmuiden	Fa. v. Campen	02550- 14516
----------	---------------	--------------

Z

Zaandam	Radio Valkenberg	075-168255
Zaandam	De Prijzenkraker	075-162562
Zaandijk	Fa. Zeltron	075-282974
Zwolle	Radio Centrum	05200- 12233
Zwolle	Electronicahuis	05200- 13804



**VIC-GEbruikers CLUB
POSTBUS 12972
1100 AZ AMSTERDAM**