

# MSX

## truuks en tips deel 6

Hans Klopper en Marcel Le Belle





**MSX**  
**Truuks en tips**  
**Deel 6**





# **MSX**

## **truuks en tips deel 6**

**Hans Klopper en Marcel Le Belle**

**uitgeverij STARK-TEXEL**

postbus 302 - 1794 ZG Oosterend tel. 02223 - 661

CIP-GEGEVENS KONINKLIJKE BIBLIOTHEEK, DEN HAAG

MSX

MSX truuks en tips. – Oosterend: Stark-Textel

Deel 6 / Hans Klopper en Marcel le Belle.

ISBN 90 6398 879 6

SISO 365.3 UDC 681.3.06

Trefw.: programmeren (computer) / MSX (computer)

1e druk 1986

ISBN 90 6398 879 6

© uitgeverij Stark-Textel, Oosterend Nh.

Niets uit deze uitgave mag worden vermenigvuldigd en/of openbaar gemaakt door middel van druk, fotokopie, microfilm of op welke andere wijze ook, zonder voorafgaande schriftelijke toestemming van de uitgever.

No part of this book may be reproduced in any form, by print, photo-print, microfilm or any other means, without prior written permission from the publisher.

Ondanks alle aan de samenstelling van de tekst bestede zorg kan noch de redactie noch de uitgever aansprakelijkheid aanvaarden voor eventuele schade die zou kunnen voortvloeien uit enige fout die in deze uitgave zou kunnen voorkomen.

MSX is een handelsmerk van Microsoft.

## Inhoudsopgave

**pagina**

### **Utilities**

Volledig aanpasbare karakterset	8
Inverse karakters	11
Inverse letters in screen 1	13
Printers en MSX	14
Printer routine	16
Diskdrive uitschakelen	17
Password invoeren	19
BSAVE een Basic programma	21
De echte insert mode	27
Onleesbaar?	29
Knipperende karakters	31

### **Peeks en pokes**

De allespoker	36
Swinging letters	38
Cirkel problemen	40
Spiegel het beeld	42
De onmogelijke reset	44
Een platte cursor	46
Snelle schermopbouw in screen 2	48

### **Schermwisroutines**

Alternatief schermwissen routine 1	51
routine 2	52
routine 3	53
routine 4	54
routine 5	55
routine 6	56
routine 7	57
routine 8	58

### **Programma's en tips**

De torens van Hanoi	60
Speciale MSX klok	65

Flitsend .....	69
Omrekenen km/u naar m/s .....	70
FOR...NEXT nader bekeken .....	71
Funktietoetsen .....	72

## **Inleiding**

Het zesde deel van de reeks "MSX Truiks en tips"... Het komt niet vaak voor dat een reeks zo lang door loopt. Daar de reacties op deze reeks zeer positief zijn hebben wij echter besloten om deel 6 ook uit te brengen. Avonden van hard werken gingen aan dit deel vooraf. Het is ook niet zo eenvoudig om te peilen waar de belangstelling c.q. behoeften van de gemiddelde MSX gebruiker liggen.

We zijn er naar ons inzien echter toch weer ingeslaagd om een reeks zeer krachtige programma's te combineren.

Een programma waar we zelf toch wel een beetje trots op zijn, is de 'kleuren correctie routine'. Deze routine zorgt er voor, wanneer u de voor- en achtergrondkleur per ongeluk gelijk kiest, dat de voorgrondkleur wordt gecorrigeerd naar wit en de achtergrondkleur naar zwart.

Verder kunt u in dit deeltje natuurlijk nog genoeg andere leuke routines terugvinden.

## **Dankwoord**

Onze dank gaat ook dit keer weer uit naar Electronics Nederland (en dan met name de heer H. van Wijk) die ons de SV 838 X' Press 16 ter beschikking heeft gesteld voor de tekstverwerking.

Benedicte wordt wederom bedankt als zijnde een grote steun bij het schrijven van dit boekje.

Tenslotte bedanken wij Kees en Joke, Alfred en José en verder alle mensen die ons hebben geholpen.



# Utilities

Utility programma's stellen u in staat om meer uit uw MSX computer te halen. Wij hebben er ook voor dit deeltje weer enige van bijzondere allure weten te creëren.

Zo bent u nu in staat om twee verschillende karaktersets in het Video RAM te bewaren en door middel van één simpele instructie te wisselen.

Het wordt zelfs mogelijk gemaakt om de cursor van vorm te veranderen.

Wij zijn er van overtuigd, dat u veel gemak van deze hulpmiddelen kunt hebben als u ze op de juiste wijze toepast. Mochten en na het lezen van dit deel nog dingen zijn die nog niet aan de orde zijn gekomen (of in de voorgaande delen), dan kunt u rustig een briefje schrijven naar de uitgever.

Misschien kunnen we het probleem dan onder de aandacht brengen in een ander deel.

Adres: uitgeverij Stark-Textel  
Postbus 302  
1794 ZG Oosterend

# Volledig aanpasbare karakterset

De karakterset van MSX computers bevindt zich in het ROM en wordt na het opstarten gekopiëerd naar het Video RAM. De Video RAM indeling onder schermmode 0 (screen 0) is dan als volgt:

decimaal	hexadecimaal	beschrijving
0000-0959	0000H-03BFH	karakter posities op scherm
2048-4096	0800H-1000H	karkater vorm tabel

Een karakter is opgebouwd uit 8 bytes. De opbouw van een karakter is daarom te vergelijken met een 8\*8 sprite. Er zijn 256 verschillende karakters aanwezig. Het adres in het Video RAM waar zich de eerste byte van een bepaald karakter bevindt, is te vinden op de volgende manier:

`PRINT &H800+ASC*8 = adres`

Hierbij is "ASC" gelijk aan de ASCII waarde van het karakter in kwestie.

Met het statement `VPOKE` bent u nu in staat om de inhoud van het Video RAM te veranderen, waardoor het bijvoorbeeld mogelijk wordt om een letter te veranderen in een vorm die u beter aan staat.

Van de formule "`&H800+ASC*8`" wordt gebruik gemaakt in het volgende programma in de regels 370-440 en 500-570. Wat hier gebeurt is vrij uniek, want normaliter is het namelijk niet mogelijk om de vorm van de cursor te veranderen. Nu is dit echter wel mogelijk, omdat we het `BASE(2)` adres hebben verschoven van `&H800` naar `&H2000`.

Experimenteer wat met uw nieuwe kennis, want er worden heel wat nieuwe mogelijkheden geschapen.



```

10 REM *****
20 REM *
30 REM * Twee verschillende *
40 REM * karaktersets in het *
50 REM * VRAM - normaal &h800 *
60 REM * tweede &h2000 *
70 REM *
80 REM *****
90 :
100 SCREEN0
110 DEFINT A-Z
120 TELLER=0
130 LOCATE 5,5:PRINT"Even geduld A.U.B."
140 FOR I=&H800 TO &H1000
150 A=VPEEK(I) :TELLER=TELLER+1
160 :
170 VPOKE TELLER+&H2000,A
180 VPOKE 35,RND(1)*255
190 NEXT
200 :
210 REM
220 PRINT"*****"
230 PRINT"*
240 PRINT"* Er zijn nu twee ver- *
250 PRINT"* schillende karaktersets *
260 PRINT"* beschikbaar. *
270 PRINT"*
280 PRINT"*****"
290 :
300 PRINT:PRINT"BASE(2)=&h800 = normaal
310 PRINT:PRINT"BASE(2)=&h2000 = normaal
+
320 :
330 REM *****
340 REM *** SPATIE AANPASSEN ***
350 REM *****

```

```
360 :
370 VPOKE 0+&H2000+32*8,&B11111100
380 VPOKE 1+&H2000+32*8,&B10000100
390 VPOKE 2+&H2000+32*8,&B10000100
400 VPOKE 3+&H2000+32*8,&B10000100
410 VPOKE 4+&H2000+32*8,&B10000100
420 VPOKE 5+&H2000+32*8,&B10000100
430 VPOKE 6+&H2000+32*8,&B10000100
440 VPOKE 7+&H2000+32*8,&B11111100
450 :
460 REM *****
470 REM *** CURSOR AANPASSEN ***
480 REM *****
490 :
500 VPOKE 0+&H2000+255*8,&B11001100
510 VPOKE 1+&H2000+255*8,&B01001000
520 VPOKE 2+&H2000+255*8,&B00000000
530 VPOKE 3+&H2000+255*8,&B00110000
540 VPOKE 4+&H2000+255*8,&B10000100
550 VPOKE 5+&H2000+255*8,&B10000100
560 VPOKE 6+&H2000+255*8,&B01111000
570 VPOKE 7+&H2000+255*8,&B00000000
580 :
590 END
```

# Inverse karakters

Vele manieren zijn er te bedenken om een MSX computer te voorzien van een inverse karakterset. In "MSX verder uitge diept" heb ik reeds laten zien, dat het mogelijk is om in de huidige karakterset een inverse set te kopiëren. Hierdoor werd het mogelijk om normaal en inverse tekst door elkaar op het scherm te gebruiken.

Wat echter bij deze routine niet mogelijk was, was het inverse weergeven van de vele MSX graphic symbolen.

Om nu toch tot een volledige inverse karakterset te komen, hebben we de gehele karakterset geïnverteerd en deze gekopieerd naar &H2000 in het Video RAM.

Normaliter verwijst BASE(2) naar adres &H800 in het Video RAM waar zich de karakterset bevindt. Door nu dit zogenaamde BASE-adres te veranderen in &H2000 wordt de inverse-set geactiveerd. De routine werkt alleen in scherm-mode 0 (screen 0).

P.S. Na een screen 0-8 opdracht is de inverse-set weer verdwenen en levert de opdracht BASE(2) = &H2000 veel onzin op.

```
1 REM *****
2 REM * *
3 REM * Twee verschillende *
4 REM * karaktersets in het *
5 REM * VRAM - normaal *
6 REM * inverse *
7 REM * *
8 REM *****
9 ;
10 SCREEN0:WIDTH40
20 DEFINT A-Z
30 TELLER=0
40 LOCATE 5,5:PRINT"Even geduld A.U.B."
50 FOR I=&H800 TO &H1000
```

```

60 A=VPEEK(I) :TELLER=TELLER+1
70 A=AXOR255
80 VPOKE TELLER+&H2000,A
90 VPOKE 35,RND(1)*255
100 NEXT
110 :
120 REM
130 PRINT"*****"
140 PRINT"*                                     *"
150 PRINT"* Er zijn nu twee ver-         *"
160 PRINT"* schillende karaktersets *
170 PRINT"* beschikbaar.                 *"
180 PRINT"*                                     *"
190 PRINT"*****"
200 :
210 PRINT:PRINT"BASE(2)=&h800 = normaal
220 PRINT:PRINT"BASE(2)=&h2000 = inverse
230 :
240 :
250 REM *****
260 REM *                                     *
270 REM *   SPATIE AANPASSEN           *
280 REM *                                     *
290 REM *****
300 :
310 FOR I=0TO7
320 VPOKE I+&H2000+32*8,0
330 NEXT
340 END

```

# Inverse letters in screen 1

Bij het veranderen van de kleur in schermmode 1, wordt onder andere de gehele kleurentabel aangepast. Deze tabel loopt van 2000H tot 20F1H.

Met behulp van het VPOKE statement is het mogelijk om slechts een klein deel te veranderen. We kunnen dan alle hoofdletters een andere kleur geven dan de kleine letters. De eerst genoemde worden dan waargenomen als inverse karakters.

```
1 REM *****
2 REM *
3 REM * Inverse hoofdletters in *
4 REM * scherm mode 1 (SCREEN1) *
5 REM *
6 REM * Kleur in regel 110 *
7 REM *
8 REM *****
9 :
10 COLOR 3,8
20 SCREEN1
30 READ Y
40 FOR X=BASE(6)+&HBT0BASE(6)+&H8 STEP-1
50 VPOKE X,Y
60 NEXT
70 :
80 FORX=32T0255:PRINTCHR$(X);:NEXT
90 :
100 END
110 DATA 5
```



# Printers en MSX

Computers zijn leuk speelgoed, maar dat ze meer voorstellen dan slechts speelgoed weet u als gebruiker maar al te goed. Uw computer kunt u inzetten voor vrijwel elke klus. Zo kunnen er uitgebreide berekeningen worden gemaakt door middel van geavanceerde spreadsheets. Teksten worden tegenwoordig ook al niet meer door een typemachine geproduceerd, maar door een tekstverwerker.

Voor de MSX computers bestaan er talloze tekstverwerkers. Deze tekstverwerkers werken in combinatie met een printer. Hier komen echter de problemen. MSX computers beschikken zoals u reeds zult weten, over een zeer uitgebreide internationale karakterset. Standaard printers zijn echter alleen in staat om de gestandaardiseerde ASCII symbolen af te drukken. Alle andere speciale MSX symbolen geven vaak zeer vreemde resultaten.

Uw computer kan van te voren worden gewaarschuwd of u gebruik maakt van zo'n printer door middel van een simpele POKE opdracht. Het adres waarin de juiste waarde dient te worden gePOKEd is &HF417. Als de printer niet aan de MSX standaard voldoet dient dit adres een waarde te bevatten, welke niet gelijk is aan nul. Er worden dan in plaats van speciale karakters spaties naar de printer gezonden.

Er zijn echter speciale MSX printers te koop die wel in staat zijn om alle symbolen die MSX computers rijk zijn te printen. Bij deze MSX compatible printers dient op adres &HF417 een 0 te staan, om aan te geven dat ook alle speciale tekens moeten worden doorgegeven aan de printer.

Een korte routine demonstreert het gebruik van adres &HF417 als printer selector.

```
10 REM *****
20 REM *
30 REM * MSX printer (Ja/Nee) *
40 REM * JA: Graphics printen *
50 REM * NEE: I.p.v. graphics *
```

```

60 REM *           worden spaties ge- *
70 REM *           print.           *
80 REM *                                           *
90 REM *****
95 :
100 COLOR 15,1:SCREEN0:WIDTH40
110 PRINT"Heeft u een MSX printer (j/n)"
;A$
120 A$=INPUT$(1):B=INSTR("jJnN",A$)
130 PRINTA$
140 IF B=0THEN120
150 ON B GOSUB 180,180,230,230
160 END
170 :
180 ' MSX printer
190 :
200 POKE &HF417,0
210 RETURN
220 :
230 ' Géén MSX printer!
240 :
250 POKE &HF417,255
260 RETURN

```

# Printer routine

Bij het printen van een tekst is het soms makkelijk om te weten waar de kop van de printer zich bevindt. Dit kan worden nagegaan door een PEEK opdracht uit te voeren op adres &HF415. Dit adres bevat de horizontale positie van de printerkop.

Het nu volgende programma demonstreert een toepassing.

```
1 REM *****
2 REM *
3 REM * Printkop positie be- *
4 REM * palen. *
5 REM * *
6 REM *****
7 :
10 SCREEN0
20 FOR X=1TO 80 :LPRINT "test";
30 LOCATE 5,5:PRINT"Printkop pos. : ";PEEK(&HF415);:'PRINTKOP POSITIE
40 NEXT
```



# Diskdrive uitschakelen

Als er een Basic programma wordt geladen van schijf, dan wordt direct de motor van de drive uitgeschakeld. Dit alles wordt bewerkstelligd door het operating system.

Na het laden van een machinetaal programma van disk wordt de diskdrive niet uitgeschakeld, omdat het operating system niet in staat wordt gesteld om z'n werk goed af te maken. Omdat het niet zo goed is als de drive motor continu blijft draaien, hebben we de volgende routine ontwikkeld, waarmee het probleem voorgoed uit de wereld is geholpen. De routine vraagt allereerst om de naam van de te laden file. Hierna wordt gekeken wat het zogenaamde startadres is van de machinetaal file. Het startadres staat op de adressen &HFCBF en &HFCC0. Hierna wordt er door middel van een FOR-NEXT loop gewacht tot de drivemotor is gestopt met draaien. Nu wordt het programma door middel van een DEFUSR opdracht geactiveerd.

Gebruik deze routine als u niet na enkele maanden drivegebruik een nieuwe motor wilt aanschaffen...

```
10 REM *****
20 REM *
30 REM *          Diskdrive uit-
40 REM * schakelen na het laden
50 REM * van een MC programma
60 REM *
70 REM *
80 REM *****
100 :
110 SCREEN0:WIDTH40:KEYOFF:DEFDBL T
120 ON ERROR GOTO 270
130 FILES
140 PRINT:PRINT
150 :
160 INPUT "Geef de filenaam";FI$
```

```
170 BLOAD FI$
180 :
190 A$=HEX$(PEEK(&HFCBF))
200 IF LEN(A$)=1 THEN A$="0"+A$
210 B$=HEX$(PEEK(&HFCC0))
220 C$=B$+A$
230 :
240 FOR T=1 TO3200:NEXT
250 DEFUSR=VAL("&h"+C$)
260 X=USR(0):END
270 RUN
```

# Password invoeren

Het is mogelijk om programma's te beveiligen met een password. Alleen als de juiste code ingegeven is, mag van het programma gebruik gemaakt worden.

Het invoeren van de geheime code moet zo voorzichtig mogelijk gebeuren. Er mag daarom geen enkele letter van het password op het scherm verschijnen.

De volgende routine maakt dit mogelijk. Er kan een password van 6 letters gevormd worden. Dit gebeurt in regel 100. In het voorbeeld is voor het password 'msx ' gekozen. Let wel, het woord 'msx' met drie spaties!

Als u dit woord verandert, dan dient u er wel op te letten dat het uit precies zes letters of cijfers bestaat.

De routine dient voor het te beveiligen hoofdprogramma geplaatst te worden. Dat kan bijvoorbeeld met het 'MERGE' statement.

```
10 REM *****
20 REM *                                     *
30 REM *  onzichtbaar een password *
40 REM *           invoeren           *
50 REM *                                     *
60 REM *****
70 ;
80 STOPON:ONSTOPGOSUB 200
90 SCREEN0:WIDTH40:KEYOFF
100 PA$="msx  "
110 ;
120 LOCATE 2,6:PRINT"Voer het password
in (6 letters)":B$=""
130 LOCATE 7,10
140 A$=INKEY$:IF A$="" THEN 140
150 B$=B$+A$:PRINT" _";:IF LEN(B$)=6 THEN
```

```
170 ELSE GOTO 140
160 :
170 IF B$=PA$ THEN PRINT:PRINT:PRINT"U h
ebt het juiste password":PRINT"ingevoerd
, u kunt nu verder":GOTO190
180 LOCATE 7,10:PRINT"          ":GOTO 120
190 END
200 RETURN
```

# BSAVE een Basic programma

MSX Basic kent vier verschillende laad/schrijf instructies, te weten:

- 1 CLOAD
- 2 LOAD/RUN
- 3 BLOAD
- 4 MERGE

ad 1. Met in de instructie CLOAD kan een geCSAVED Basic programma van tape geladen worden. Een geCSAVED Basic programma is in een speciale gecodeerde vorm (token vorm) weggeschreven. Dit heeft als voordeel dat de programmagrootte verkleind wordt.

ad 2. Met de instructie LOAD/RUN kan een geSAVED Basic programma van tape of disk geladen worden.

Als het van tape geladen wordt, dan is er sprake van een ASCII file (dit is een file die volgens de ASCII standaard gecodeerd is).

Wordt het van disk geladen, dan kan het zowel een ASCII file als een file in token vorm zijn. Dit hangt af van de manier waarop het weggeschreven is.

ad 3. Met de instructie BLOAD kan een geBSAVED (machinetaalprogramma van disk/tape geladen worden.

Een geBSAVED programma slaat de inhoud van het geheugen op. Elk aangegeven byte wordt naar disk/tape gekopieerd.

ad 4. Met MERGE is het mogelijk om twee programma's te combineren. Het ene programma dient als ASCII file op tape/disk te staan, het andere in het geheugen. Met MERGE kan de ASCII file van disk/tape met het Basic programma in het geheugen gekoppeld worden.

De volgende routine maakt op een speciale manier gebruik

van de BSAVE en BLOAD instructies.

Het is zonder enige kennis van deze instructies niet mogelijk om het volgende programma te begrijpen, daarom worden ze kort nader toegelicht.

**BSAVE:** Zoals eerder vermeld kan men met deze instructies een gespecificeerd gedeelte van het geheugen wegschrijven naar tape/disk in binaire vorm. Dit gebeurt als volgt:

BSAVE "filenaam", beginadres, eindadres, startadres

'beginadres' stelt het begin voor van het blok geheugen dat weggeschreven moet worden, het markeert het begin van het geheugenblok.

'eindadres' stelt het eind van het blok geheugen voor, dat weggeschreven moet worden, het markeert het einde van het geheugenblok.

'startadres' stelt het startadres voor. Dit heeft alleen zin bij machinetaalprogramma's. (Zie verder bij BLOAD.)

**BLOAD:** Met deze instructie kan een geBSAVED programma van disk/tape geladen worden; wordt de parameter 'r' toegevoegd (alleen bij machinetaalprogramma's heeft dit een betekenis), dan start het programma bij het 'startadres' dat opgegeven is bij de BSAVE instructie. De syntax is als volgt:

BLOAD "filenaam",r

of

BLOAD "filenaam"

Na deze instructie is het met BSAVE gespecificeerde geheugendeel weer geladen met de getallen (bytes) die tijdens het BSAVEN in het geheugen stonden.

**Een Basic programma BSAVEN:**

Als we het begin- en eindadres van een Basic programma



kennen, dan kunnen we natuurlijk ook zo'n programma BSAVEN. Er zit echter een addertje onder het gras; schrijven we een Basic programma in binaire vorm weg en laden het weer in, dan krijgen we een foutmelding. Gelukkig is dit probleem eenvoudig op te lossen, door een aantal systeemvariabelen mee te nemen tijdens het wegschrijven.

De volgende routine schrijft ieder Basic programma in binaire vorm (BSAVE) weg. Bovendien worden er nog een aantal beveiligingen aangebracht:

- 1 Na het inladen van het programma verschijnt er onzin in beeld als er om een listing gevraagd wordt.
- 2 Na het runnen van het programma is het niet meer te listen en te breaken.
- 3 Het programma wordt opgestart door op de ESC toets te drukken (dus niet met 'RUN').

### **Hoe dient u te handelen**

Neem onderstaande routine geheel over. LET OP: houd de regelnummering aan en neem alles over inclusief de REMs en dubbele punten.

Schrijf het programma als een ASCII file weg:

Op tape door middel van SAVE"filenaam"

Op disk door middel van SAVE"filenaam",a

Laad het te BSAVEN Basic programma in het geheugen en tik het volgende in:

RENUM (return)

RENUM 1000,10 (return)

Merge de bovenstaande weggeschreven ASCII file.

Run het programma door RUN + (return) in te geven.

Er wordt gevraagd een filenaam in te geven.

Stel we hebben de filenaam 'test' 'zojuist ingevoerd, dan worden de volgende programma's weggeschreven:

- 1 test : bevat het Basic programma
- 2 test.sys : bevat de bovengenoemde systeemvariabelen
- 3 a : bevat de nodige machinetaal
- 4 d : bevat de nodige machinetaal

Het geBSAVEde programma kan nu als volgt weer geladen worden:

```
BLOAD"test":BLOAD"test.sys":BLOAD"a":BLOAD
"d" + (return)
```

Als de ESC toets ingedrukt wordt, dan begint het programma te lopen.

```
10 REM *****
20 REM * *
30 REM * GEAVANCEERDE *
40 REM * PROGRAMMA BEVEILIGING *
50 REM * *
60 REM *****
70 :
80 STOPON:ONSTOP GOSUB 550:IF PEEK(&HE000)=&HF3 THEN 1000
90 :
100 DATA FE,3A : 'CP "ESC"
110 DATA CA,00,E0 : 'JP Z,E000
120 :
130 RESTORE 100
140 T=&HFDCC
150 FOR X=0 TO 5:READA$:POKE T+X,VAL("&H"+A$):NEXT
160 :
170 CLEAR 200,&HE000
180 T=&HE000:RESTORE 390
```



```

190 READ A$:IFA$="**" THEN GOTO 200
   ELSE POKE T,VAL("&H"+A$):T=T+1:GOTO190
200 :
210 WIDTH40:CLS
220 :
230 '** EIND ADRES VAN EEN BASIC **
     ** PROGRAMMA BEPALEN.          **
240 :
250 BE=PEEK(&HF6C6)+256*PEEK(&HF6C7)
260 X$=HEX$(BE):'EINDADRES IN HEX VORM
270 :
280 INPUT"Voer de naam van het weg te sc
hrijven programma dat maximaal uit 8 l
etters mag bestaan zonder extensie";A$
290 :
300 B$=A$+".SYS"
310 AX=PEEK(&H8002):POKE&HF6CC,AX
320 AX=PEEK(&H8003):POKE&HF6CD,AX
330 AX=PEEK(&H8004):POKE&HF6CF,AX
340 AX=PEEK(&H8005):POKE&HF6D0,AX
350 AX=PEEK(&H8006):POKE&HF6D1,AX
360 AX=PEEK(&H8007):POKE&HF6D2,AX
370 POKE&H0FFF,&HDD:POKE&H0FFE,&HDD
380 :
390 DATA F3                :'DI
400 DATA 2A,CC,F6          :'LD HL,(F6CC)
410 DATA 22,02,80         :'LD (8002),HL
420 DATA 2A,CF,F6          :'LD HL,(F6CF)
430 DATA 22,04,80         :'LD (8004),HL
440 DATA 2A,D1,F6          :'LD HL,(F6D1)
450 DATA 22,06,80         :'LD (8006),HL
460 DATA FB                :'EI
470 DATA C3,AC,73,"**":'JP 73AC
480 :
490 FOR X=&H8002 TO &H8007
500 POKE X,RND(-TIME)*255:NEXT

```

```
510 :  
520 BSAVE B$, &HF6C0, &HF6D3  
530 BSAVE A$, &H8000, BE  
540 BSAVE "a", &HDFFE, &HE017:BSAVE "d", &HFD  
CC, &HFDD3:END  
550 RETURN
```

## De echte insert mode

Als u in direct mode op de insert toets drukt, dan verschijnt er een cursor die anders van vorm is. U bevindt zich dan in de zogenaamde 'insert mode'.

MSX Basic kent geen blijvende insert mode; als u bijvoorbeeld op de enter/return toets drukt, dan wordt deze mode opgeheven. Dit is vaak lastig.

Daarom volgt hier een routine die een blijvende insert mode genereert. Met behulp van de ESC toets kunt u deze mode in- c.q. uitschakelen.

Opmerking: Nadat u deze routine geladen heeft, is het niet meer mogelijk om de insert toets te gebruiken.

```
10 REM *****
20 REM *
30 REM *      INSERT MODE      *
40 REM *
50 REM *   VIA DE ESC TOETS   *
60 REM *
70 REM *****
80 :
90 POKE &HFDC3,&H0:POKE &HFDC4,&HE2
   :POKE &HFDC2,&HCD
100 :
110 DATA 21,A8,FC:'LD HL,#F3DC
120 DATA 46      :'LD B,(HL)
130 DATA 3A,30,E2:'LD A,(E230)
140 DATA B8      :'CP B
150 DATA C2,0C,E2:'JP NZ,E20C
160 DATA C9      :'RET
170 DATA 32,A8,FC:'LD (#F3DC),A
180 DATA 32,AA,FC:'LD (#FCAA),A
190 DATA C9      :'RET
200 DATA "***"
```

```

210 :
220 CLEAR 25,&HE200
230 T =&HE200
240 READ A$:IF A$="**" THEN 250 ELSE POK
E T,VAL("&H"+A$):T=T+1:GOTO 240
250 POKE&HE230,0
260 :
270 DATA FE,3a      :'CP 3a
280 DATA c0        :'ret nz
290 DATA f5        :'PUSH AF
300 DATA 3A,30,E2:'LD A,(E230)
310 DATA EE,1      :'XOR 1
320 DATA 32,30,E2:'LD (E230),A
330 DATA f1        :'POP AF
340 DATA C9        :'RET
350 DATA "**"
360 :
370 T=&HE220:RESTORE 270
380 READ A$:IF A$="**" THEN 390 ELSE POK
E T,VAL("&H"+A$):T=T+1:GOTO 380
390 POKE&HFDCC,&H20:POKE&HFDCE,&HE2:POKE
&HFDCC,&HCD

```

# Onleesbaar???

De allereerste computers werkten slechts met 2 standaard kleuren. Deze computers kenden dan ook niet de mogelijkheid om de voor- en achtergrondkleur naar eigen smaak in te stellen.

MSX computers zijn echter ontwikkeld naar de laatste stand van de techniek. De MSX-1 reeks beschikt over 16 kleuren en de MSX2 computers kunnen zelfs kiezen uit een reeks van 512 kleuren. De schermen 0 en 1 worden normaliter gebruikt voor tekst. Deze teksten kunnen worden weergegeven in een van de 16 beschikbare kleuren. De achtergrondkleur is tevens instelbaar. Ook hier is de MSX gebruiker in staat om uit een palet van 15 kleuren te kiezen.

Wat u als gebruiker vast wel eens heeft meegemaakt, is het feit dat door wat voor reden dan ook, de voor- en achtergrondkleur dezelfde waarden aannamen.

Bijvoorbeeld: COLOR 3,3,1

Dit zou resulteren in een geheel groen beeld, waarop geen letter is te lezen. Dat komt doordat de voorgrondkleur en de achtergrondkleur dezelfde waarde (3 = groen) hebben aangenomen.

Om dit probleem enigszins op te vangen, hebben wij voor u een machinecode routine ontwikkeld die voorgoed afrekent met dit probleem.

De werking van deze routine berust op een vergelijking tussen de waarden van de voor- en achtergrondkleur. Als deze gelijk zijn dan wordt de voorgrondkleur wit (= 15) en de achtergrondkleur zwart (1).

De waarde van de voorgrondkleur bevindt zich op adres &HF3E9.

De waarde van de achtergrondkleur bevindt zich op adres &HF3EA.

De routine is geschreven in machinetaal, dus ze blijft in het

geheugen aanwezig na een NEW opdracht. Dit betekent dat het onzichtbaar-euvel in principe tot het verleden zal behoren.

```
100 REM *****
110 REM *
120 REM * ACHTERGROND EN VOOR- *
130 REM * GROND KLEUR VERSCHIL- *
140 REM * LEND HOUDEN. *
150 REM *
160 REM *****
170 :
180 POKE &HFF0C,&HC3:POKE &HFF0D,&HO
:POKE &HFF0E,&HC0
190 :
200 DATA 21,EA,F3:'LD HL,#F3EA - bakclr
210 DATA 46 : 'LD B,(HL)
220 DATA 3A,E9,F3:'LD A(#F3E9) - forclr
230 DATA B8 : 'CP B
240 DATA CA,0D,C0:'JP Z,COLOR 15,1
250 DATA C9 : 'RET
260 DATA 3E,0F : 'COLOR 15,1:LD A,15
270 DATA 3E,0F : 'LD A,15
280 DATA 32,E9,F3:'LD (#F3E9),A
290 DATA 3E,01 : 'LD A,1
300 DATA 32,EA,F3:'LD (#F3EA),A
310 DATA CD,62,00:'CALL #0062
320 DATA C9 : 'RET
330 DATA "**"
340 :
350 CLEAR 25,&HBFFF
360 T =&HC000
370 READ A$:IF A$="**" THEN 380 ELSE POK
E T,VAL("&H"+A$):T=T+1:GOTO 370
380 DEFUSR =&HC000:X=USR(0)
```



# Knipperende karakters

Wat wij zeker een gemis vinden bij de MSX computers is de mogelijkheid om karakters te laten knipperen. U kunt natuurlijk een karakter aan en uit zetten, maar met meerdere karakters gaat dat al gauw te langzaam.

Om een oplossing te kunnen bieden voor dit gemis, hebben we een routine ontwikkeld die willekeurige stukken tekst op het scherm kan laten knipperen.

We maken hierbij gebruik van het feit dat de karakterset zich in het Video RAM bevindt en dat tevens de gegevens omtrent de karakter positie zich in dit RAM bevinden.

Voor de duidelijkheid hebben we een volledig programma gemaakt om u een goed overzicht te geven omtrent de werking van de routine.

Veel succes bij de toepassing in uw eigen programma's.

```
10 REM *****
20 REM * *
30 REM * KNIPPERENDE KARAKTERS *
40 REM * *
50 REM * *
60 REM *****
70 :
80 SCREEN0: WIDTH40
90 DIM A (10),B (10)
100 DEFINT A-Z
110 KEYOFF
120 LOCATE 17,5:PRINT"MSX"
130 LOCATE 12,7:PRINT"TRUKS EN TIPS"
140 LOCATE 16,9:PRINT"DEEL 6"
150 :
160 REM TEKST MSX LATEN KNIPPEREN
170 :
180 S=1
```

```

190 BE=17:EI=19:T=5:GOSUB 650
200 AANTAL=5:SA=200:SU=400:GOSUB 560
210 :
220 REM TEKST TRUKS LATEN KNIPPEREN
230 :
240 S=1
250 BE=12:EI=17:T=7:GOSUB 650
260 AANTAL=10:SA=400:SU=300:GOSUB 560
270 :
280 REM TEKST EN LATEN KNIPPEREN
290 :
300 S=1
310 BE=19:EI=20:T=7:GOSUB 650
320 AANTAL=3:SA=700:SU=200:GOSUB 560
330 :
340 REM TEKST TIPS LATEN KNIPPEREN
350 :
360 S=1
370 BE=22:EI=25:T=7:GOSUB650
380 AANTAL=10:SA=300:SA=300:GOSUB 560
390 :
400 REM TEKST DEEL 3 LATEN KNIPPEREN
410 :
420 S=1
430 BE=16:EI=21:T=9:GOSUB 650
440 AANTAL=5:SA=700:SU=200:GOSUB 560
450 END
460 :
470 :
480 REM TEKENS OPSLAAN
490 :
500 A(S)=(HD)+(VE)*40
510 B(S)=VPEEK(A(S))
520 RETURN
530 :
540 REM TEKST LATEN KNIPPEREN

```



```
550 :  
560 FOR Q=1 TO AANTAL :FOR T=1 TO S  
570 VPOKE A (T),0:NEXT:FOR M=1 TO SU:NEX  
T  
580 FOR T=1 TO S  
590 VPOKE A (T),B(T):NEXT:FOR M=1 TO SA  
600 NEXT:NEXT  
610 RETURN  
620 :  
630 REM TEKENS INLEZEN  
640 :  
650 FOR V=BE TO EI:S=S+1  
660 HD=V:VE=T:GOSUB 500  
670 NEXT  
680 RETURN
```



# Peeks en pokes

Ook in dit deel vindt u weer de nodige Peeks en Pokes terug. Het gebruik van de peek en poke opdrachten vergt een vergaande kennis van het zogenaamde systeem RAM van uw computer.

Het ligt in de traditie van de MSX Truuks en tips reeks om ruimschoots aandacht te besteden aan de peeks en pokes. In de loop der tijd zijn er al een aanzienlijk aantal de revue gepasseerd.

In dit deel hebben we een programma opgenomen, dat u in staat stelt om ook zogenaamde 'updated variables' te poken. De tekst bij het programma zal u duidelijk maken hoe het een en ander in zijn werk gaat.

# De allespoker

In de Truiks en tips reeks maken we vaak gebruik van enkele slimme PEEKS en POKES. De meeste POKES vanaf adres &H8000 geven wat dit betreft geen enkel probleem. Enkele adressen die zich in het systeem RAM bevinden, worden echter door de computer om een bepaalde tijd van een bepaalde waarde voorzien. De inhoud van deze adressen kan daarom wel worden veranderd, maar kort daarop neemt dit adres weer zijn oude waarde in.

Als u op deze plaatsen toch een andere waarde wilt POKEn, dan biedt de volgende routine uitkomst. Er wordt regelmatig naar een machinecode routine gesprongen die een door u gekozen waarde in het door u gekozen adres POKEt.

Een voorbeeld van een "updated" adres is &HFBD9. Dit adres komt in het programma voor in regel 150 en in regel 180.

Regel 150: Hier wordt slechts een PEEK gemaakt naar het adres, om aan te geven dat de inhoud van het adres ook daadwerkelijk is veranderd.

Regel 180: Hier wordt het adres gedefinieerd in de machinetaalroutine. In ons voorbeeld hebben we adres &HFBD9 gebruikt. U kunt zelf natuurlijk elk ander adres invoeren. U dient er rekening mee te houden dat de essentiële informatie zich bevindt in de DATA regels. Het adres &HFBD9 staat daar in tweeën gehakt, en omgekeerd. Als u nu bijvoorbeeld adres &HFC00 wilt POKEn, dan dient u dit in tweeën te hakken (FC en 00). Dit dient u om te draaien. Als resultaat hebben we dan de hexadecimale waarden 00 en FC.

De DATA regel 180 bevat dus nu:

```
180 DATA 21,00,FC:LD HL,#FC00
```

Succes bij de toepassing van de routine.

```

10 REM *****
20 REM * *
30 REM * POKEN OP ELK RAM ADRES *
40 REM * D.M.V. EEN MC-ROUTINE *
50 REM * *
60 REM * ADRES IN LIJN 180!!!! *
70 REM * *
80 REM *****
90 ;
95 INTERVAL ON
100 GOTO 240 : 'i.p.v. GOSUB
110 COLOR 15,1:SCREEN0:WIDTH40
120 INPUT "Welke waarde wenst u te poken
?";REGELNR
125 IF REGELNR>=256 THEN CLS:GOTO 120
130 POKE &HFEC5,REGELNR
140 CLS:PRINT "WAARDE: ";PEEK(&HFEC5)
150 PRINTPEEK(&HFBD9):GOTO 500
160 ;
170 : 'ORG #C000
180 DATA 21,D9,FB: 'LD HL,#Fbd9
190 DATA 3A,C5,FE: 'LD A,(#FEC5)
200 DATA 77 : 'LD (HL),A
210 DATA C9 : 'RET
220 DATA "**" : 'END
230 ;
240 CLEAR 200,&HBFFF
250 T=&HC000
260 READA$: IFA$="**" THEN 270 ELSE POKET, VAL
("&H"+A$): T=T+1: GOTO 260
270 DEFUSR=&HC000
280 GOTO 110 : 'i.p.v. RETURN
500 POKE &HFD9A,&HC3: POKE &HFD9B,0
: POKE &HFD9C,&HCO 'INTERRUPT HOOK
600 FOR X=1 TO 500: PRINTPEEK(&HFBD9) " ";: N
EXT : 'BEWIJS NIET ALTIJD GELIJK AAN GEWE
NSTE WAARDE

```

# Swinging letters

Het volgende programma had ik al een tijdje liggen, maar het was nooit echt voltooid. Hierin heb ik i.v.m. de Truuks en tips reeks verandering gebracht.

Het geheel bestaat uit twee delen. Het eerste programma bestaat voor het overgrote deel uit machinetaal. Er wordt voor gezorgd, dat de kolommen in screen 1 gaan scrollen.

Adres 50003 bevat de waarde van de te scrollen tekstkolom.

Het tweede programma is een voorbeeld van een toepassing van bovengenoemde routine.

Experimenteert u zelf en ontdek nieuwe toepassingen.

```
10 SCREEN0
20 CLEAR 200,&HC000
30 PA=&HF470:DEFUSR1=PA
40 FOR X=&H9D08 TO &H9D36:READA$:POKEX,V
AL("&H"+A$):NEXT
50 DATA 3a,53,c3 :' ld a,(50003)
60 DATA 3c      :' inc a
70 DATA 21,e0,1a :' ld hl,6880
80 DATA 3d      :' loop: DEC A
90 DATA ca,17,9d :' jp z,loop1
100 DATA 23     :' inc hl
110 DATA c3,0f,9d:' jp loop
120 DATA cd,4a,00:' call 74
130 DATA f5     :' push af
140 DATA 01,17,00:' ld bc,23
150 DATA 11,20,00:' ld de,32
160 DATA ed,52  :' sbc hl,de
170 DATA cd,4a,00:' call 74
180 DATA 19     :' add hl,de
190 DATA cd,4d,00:' call 77
200 DATA ed,52  :' sbc hl,de
```

```

210 DATA 0b      :' dec bc
220 DATA 79      :' ld a,c
230 DATA b0      :' or b
240 DATA c2,21,9d:' jp nz,loop2
250 DATA f1      :' pop 77
260 DATA cd,4d,00:' call 77
270 DATA c9      :' ret
280 :
290 ' copyright H.Klopper (c) 1985
300 :

```

```

10 KEY OFF
20 DEFUSR=40200!
30 COLOR 15,1,1
40 SCREEN1
50 PRINT"Dit is een demonstratie van ":P
RINT"een machinetaal programma.
60 PRINT"Het programma gooit de hele":PR
INT"boel op het scherm door":PRINT"de wa
r. Aanschouw 'de ":PRINT"swinging letter
s' op het":PRINT"scherf."
70 PRINT"De groetjes, en verder veel":PR
INT"computerplezier.....
....."
80 FORT=1T07000:NEXT
90 POKE 50003!,20
100 'FORX=1T02
110 X=USR(1)
120 FOR T=1T025:NEXT
130 POKE 50003!,RND(1)*32+1
140 'NEXT
150 AA=AA+1:IFAA=250THENGOTO160ELSE GOTO
110
160 RUN

```



# Cirkel problemen

Het commando CIRCLE stelt u in staat om in no time een cirkel in SCREEN 2 of hoger te tekenen. Als voorbeeld voor ons circle probleem tekenen we een cirkel in screen 2.

```
10 SCREEN 2
20 CIRCLE (128,96),60,15
30 GOTO 30
```

Dit is volgens alle handleidingen een mooie *ronde* cirkel. Dat dat echter niet het geval is kunt u direkt zien als u een rond voorwerp tegen uw scherm aanhoudt en de omtrekken vergelijkt met de omtrekken van de zo genaamde cirkel op het scherm.

Dat deze cirkel meer op een ei lijkt, dan op een rond geheel, komt door de schermresolutie. Doordat er in horizontale richting 256 pixels en in verticale richting maar 192 pixels zijn, ontstaat er een verkeerde verhouding.

De cirkel wordt als het ware uit elkaar getrokken in horizontale richting. Om dit probleem uit de wereld te helpen, kunt u een POKE opdracht gebruiken. Deze POKE opdracht verandert de zo genaamde reciproke van de aspekt ratio van de cirkel. Dat zijn een heleboel moeilijke woorden, maar het zegt zoveel, als het aanpassen van de verhoudingen.

De gegevens omtrent de aspekt ratio bevinden zich op de adressen &HF40B en &HF40D in het RAM geheugen. Op deze adressen dient u de waarde 206 te POKEn. Onderstaand programma demonstreert een toepassing van deze POKEs.

```
10 REM *****
20 REM *
30 REM * RONDE CIRCLE I.P.V. *
40 REM * HET BERUCHTE MSX "EI" *
50 REM *
60 REM *****
70 :
```

```
100 COLOR 3,3:SCREEN2
110 :
120 POKE &HF40B,206 : 'RECIPROKE V/D
130 POKE &HF40D,206 : 'ASPECT RATIO
140 :
150 CIRCLE (128,96),80,15
160 PAINT (128,96),15
170 GOTO 170
```

# Spiegel het beeld

De volgende truuk maakt op een leuke manier gebruik van de VPOKE en VPEEK statements. Allereerst wordt het gehele scherm regel voor regel (screen 0, locaties 0-959) in een array gelezen. Daarna wordt het horizontaal en vertikaal gespiegeld weer terug gezet.

Hoewel het programma geheel uit Basic statements bestaat, is het toch snel.

U kunt deze routine natuurlijk ook in zelf ontwikkelde programma's opnemen.

Opmerkingen: Zorg er voor dat er wel tekst op het scherm staat alvorens het programma te runnen.

Het programma werkt alleen in schermmode 1.

De schermbreedte dient op 40 ingesteld te worden.

```
10 REM *****
20 REM *
30 REM * DEZE ROUTINE GEEFT DE *
40 REM * INHOUD VAN SCREEN0 IN *
50 REM * SPIEGELBEELD WEER. *
60 REM *
70 REM *****
80 :
90 WIDTH40:KEYOFF
100 DEFINT A,X:DIM A(1000)
110 :
120 FOR X=0 TO 959
130 A(X)=VPEEK(X)
140 NEXT
150 :
160 FOR X=959 TO 0 STEP -1
170 VPOKE 959-X,A(X)
180 NEXT
190 :
```

```
200 FOR T=1 TO 2000:NEXT
210 FOR X=0 TO 959
220 VPOKE X,A(X)
230 NEXT
240 :
250 GOTO 120
```

# De onmogelijke reset

Vaak worden programma's 'gekraakt' met behulp van de RESET knop. Na een RESET bevindt het programma zich nog steeds in het geheugen. Het is dan meestal een koud kunstje om het programma op disk te zetten.

De volgende routine maakt dit onmogelijk; na een RESET zal de computer 'ophangen'. Uit en aan zetten is dan de enige mogelijkheid. Dat betekent wel dat het programma verdwenen is.

## Het gebruik van de routine

- 1 Laad het onderstaande programma in en run het
- 2 Voer een filenaam in
- 3 Het machinetaalprogramma wordt nu naar tape/disk geschreven
- 4 Het machinetaalprogramma kan weer geladen worden door middel van de instructie:

BLOAD"filenaam",r

Deze routine kunt u voor elk hoofdprogramma plaatsen zodat een RESET onmogelijk wordt.

N.B.: Het programma werkt niet op MSX2 computers.

```
10 REM *****
20 REM *
30 REM * Deze routine maakt *
40 REM *
50 REM * een reset onmogelijk *
60 REM *
70 REM *****
80 ;
90 LOCATE ,,0
```

```

100 CLEAR 200,&HE200
110 X=INP(&HA8):POKE&HE201,X
120 X=X AND 240
130 X=X+X/16
140 POKE&HE200,X
150 :
160 DATA F3          :'DI
170 DATA 3A,00,E2   :'LD A,(E200H)
180 DATA D3,A8      :'OUT (A8H),A
190 DATA 3E,41      :'LD A,41H
200 DATA 32,00,40   :'LD (4000H),A
210 DATA 3E,42      :'LD A,42H
220 DATA 32,01,40   :'LD (4001H),A
230 DATA 3E,00      :'LD A,00H
240 DATA 32,02,40   :'LD (4002H),A
250 DATA 3E,00      :'LD A,00H
260 DATA 32,03,40   :'LD (4003H),A
270 DATA 3A,01,E2   :'LD A,(E201H)
280 DATA D3,A8      :'OUT (A8H),A
290 DATA FB         :'EI
300 DATA C9         :'RET
310 DATA "**"
320 :
330 T=&HE202
340 READ A$:IF A$="**" THEN 350 ELSE POK
E T,VAL("&H"+A$):T=T+1:GOTO340
350 DEFUSR=&HE202
360 X=USR(0):SCREEN0
370 :
380 PRINT"wilt u de machinetaal BSAVEN?"
390 A$=INPUT$(1)
400 IF A$="n" THEN END
410 INPUT"Onder welke naam";A$
420 BSAVE A$,&HE200,&HE225,&HE202

```

# Een platte cursor

De volgende routine maakt de vierkante cursor plat als u zich in direct mode bevindt.

Met de volgende POKE instructie kunt u de cursor weer normaal maken:

```
POKE&HFDC2,&HC9
```

Deze POKE zet de gebruikte 'hook' weer op C9H, de code voor return.

```
10 INPUT "test
100 REM *****
110 REM *
120 REM * een platte cursor *
130 REM *
140 REM * versie 2 *
150 REM *
160 REM *****
170 :
180 POKE &HFDC3,&H0:POKE &HFDC4,&HE2
:POKE &HFDC2,&HCD
190 :
200 DATA 21,AA,FC:'LD HL,#FCAA
210 DATA 46 : 'LD B,(HL)
220 DATA 3E,1 : 'LD A,1
230 DATA B8 : 'CP B
240 DATA c2,0B,E2:'JP NC,E20B
250 DATA C9 : 'RET
300 DATA 32,AA,FC:'LD (#FCAA),A
320 DATA C9 : 'RET
330 DATA "**"
340 :
350 CLEAR 25,&HE200
```



```
360 T = &HE200
370 READ A$: IF A$ = "**" THEN 380 ELSE POK
E T, VAL("&H"+A$): T = T + 1: GOTO 370
```

# Snelle schermopbouw in screen 2

Schermmode 2 wordt bij de MSX computers van huis uit gebruikt als grafisch scherm. Er bestaat een hele reeks opdrachten om grafische voorstellingen te creëren.

Een van deze grafische instructies is "LINE". Hiermee is het onder andere mogelijk, om blokken met een bepaalde kleur te vullen. Dit gaat vrij snel, maar voor ons gaat het nooit snel genoeg.

Het is door middel van een slimme truuk mogelijk om zeer snel stukken beeldscherm met informatie te vullen. Allereerst dient u te weten dat SCREEN2 is opgebouwd uit drie delen van 256 bytes lengte. Elk deel verwijst naar een iegen plaats in de nametable. Elke waarde van een karakter in dit deel van de nametable wordt gedefiniëerd door de eerste 2048 bytes van de karakervormtabel.

Het middelste deel van het scherm komt overeen met het 256e tot en met het 511e byte van de nametable.

Bytes 512 tot en met 767 komen overeen met het derde deel van het scherm.

Dit alles komt er op neer, dat als je een bepaalde positie op het scherm een waarde geeft tussen 0 en 255, dan zal op die plaats het karakter verschijnen dat op die karakterpositie van het betreffende schermdeel staat.

Het vullen van het scherm met een bepaalde waarde kan gebeuren door middel van de VPOKE instructie (regel 150-190). Het zijn echter 3\*256 bytes, dus dat zal enige tijd vergen. Om deze reden hebben we een machinetaalroutine toegevoegd, welke de juiste waarden VPOKEt.

Het programma demonstreert nu het verschil tussen het LINE commando en bovengenoemde methode met behulp van machinetaal.

```

10 COLOR 15,1,1:SCREEN2
20 CLOSE:OPEN"GRP:"AS1
30 PRESET(10,10):
  PRINT#1,"VLAG IN BASIC
40 GOSUB 460
50 LINE(0,0)-(255,64),6,BF
60 LINE(0,65)-(255,130),15,BF
70 LINE(0,131)-(255,191),4,BF
80 FOR T=1TO500:NEXT
90 COLOR 15,1,1:SCREEN2
100 CLOSE:OPEN"GRP:"AS1
110 COLOR 15:PSET(10,10):
  PRINT#1,"VLAG IN MACHINETAAL....
120 GOSUB 460
130 GOSUB 250
140 :
150 REM *****
160 REM * FOR X=&H1800 TO &H1AFD *
170 REM * VPOKE X,1 *
180 REM * NEXT *
190 REM *****
200 :
210 LINE (8,0)-(15,7),6,BF
220 LINE (8,63)-(15,71),15,BF
230 LINE (8,126)-(15,135),4,BF
235 X=USR(0)
240 GOTO 240
250 DEFUSR=&HD000
260 :
261 T=&HD000
262 READ A$:IFA$="**"THEN RETURN ELSE PO
KE T,VAL("&H"+A$):T=T+1:GOTO 262
264 :
270 DATA F3 : 'DI
275 DATA 21,00,18: 'LD HL,#1800
280 DATA CD,DF,07: 'CALL #07DF

```

```

285 DATA E5      : 'PUSH HL
290 DATA E1      : 'POP  HL
295 DATA 01,00,03: 'LD  BC,#0300
300 DATA 3E,01   : 'LD  A,1
305 DATA D3,98   : 'OUT (#98),A
310 DATA 0B      : 'DEC  BC
315 DATA 78      : 'LD  A,B
320 DATA B1      : 'OR   C
325 DATA 20,F7   : 'JR  NZ,LOOP
330 DATA FB      : 'EI
335 DATA C9      : 'RET
336 DATA "**"     : 'END
340 ;
360 RETURN
460 ;           FOR TIJD=1TO300:NEXT
560 RETURN

```

# Alternatief schermwissen

De volgende routines maken het mogelijk om in screenmode 0 het scherm op een andere manier te wissen dan met CLS. Iedere routine wist op een eigenwijze manier het scherm. Bovendien kunnen ze ook voor andere toepassingen gebruikt worden.

```
10 REM *****
20 REM * * * * *
30 REM * WIS HET SCHERM *
40 REM * * * * *
50 REM * ROUTINE 1 *
60 REM * * * * *
70 REM *****
80 :
90 DEFINT A-Z
100 DIMA(25)
105 :
110 FOR T=0 TO 39
120 FOR Q=1 TO 2:S=0
130 LOCATE 0,0:PRINTSPACE$(80)
135 :
140 FOR X=0 TO 39*24 STEP 40:S=S+1
150 A(S)=VPEEK(X+T):NEXT
160 S=0
170 :
180 FOR X=80 TO 39*24STEP40:S=S+1
190 VPOKE X+T,A(S):M=M+1
195 :
200 NEXT:NEXT
210 NEXT
220 GOTO 110
```

```

10 REM *****
20 REM *
30 REM *      WIS HET SCHERM      *
40 REM *
50 REM *      ROUTINE 2          *
60 REM *
70 REM *****
80 :
90 DEFINT A-Z
100 DIMA(25):FOR Q=0 TO 100
110 :
120 T=RND(-TIME)*40+1:S=0
130 LOCATE 0,0:PRINTSPACE$(80)
140 :
150 FOR X=0 TO 39*20 STEP 40:S=S+1
160 A(S)=VPEEK(X+T):NEXT
170 S=0
180 :
190 FOR X=80 TO 39*20STEP40:S=S+1
200 VPOKE X+T,A(S):M=M+1
210 NEXT:NEXT

```

```

10 REM *****
20 REM *
30 REM *          WIS HET SCHERM          *
40 REM *
50 REM *          ROUTINE 3              *
60 REM *
70 REM *****
80 :
90 DEFINT A-Z:
100 LOCATE 0,23:PRINTSPACE$(80)
110 :
120 DIMA(25):FOR Q=0 TO 500
130 T=RND(-TIME)*40+1:S=0
140 FOR X=80TO 39*20 STEP 40:S=S+1
150 A(S)=VPEEK(X+T):NEXT
160 S=0
170 :
180 FOR X=0 TO 39*20STEP40:S=S+1
190 VPOKE X+T,A(S):M=M+1
200 NEXT:NEXT

```



```

10 REM *****
20 REM *
30 REM *          WIS HET SCHERM          *
40 REM *
50 REM *          ROUTINE 4              *
60 REM *
70 REM *****
80 ;
90 WIDTH40:KEYOFF
100 DEFINT A-Z
110 ;
120 O=23:B=0
130 FOR X=1 TO 12
140 LOCATE O,0:PRINTSPACE$(40);
145 ;
150 FOR M=1 TO 250:NEXT
160 LOCATE O,B:PRINTSPACE$(40);
170 O=O-1:B=B+1
180 NEXT

```

```
1 REM *****
2 REM *
3 REM *      WIS HET SCHERM      *
4 REM *
5 REM *      ROUTINE 5          *
6 REM *
7 REM *****
8 :
10 DEFINT A-Z
15 FOR T=0 TO 39
20 FOR X=0 TO 24*39 STEP 40
30 VPOKE X+T,0
40 NEXT:NEXT
```

```

10 REM *****
20 REM *
30 REM *          WIS HET SCHERM          *
40 REM *
50 REM *          ROUTINE 6              *
60 REM *
70 REM *****
80 :
90 DEFINT A-Z
100 L=0:R=39
110 FOR X=0 TO 19
120 FOR M=0 TO 24*39 STEP40
130 VPOKE M+L,0
140 NEXT
150 :
160 FOR M=0 TO 24*39 STEP40
170 VPOKE M+R,0
180 NEXT
190 L=L+1:R=R-1:NEXT
200 LOCATE 0,0

```

```

1 REM *****
2 REM *
3 REM *      WIS HET SCHERM      *
4 REM *
5 REM *      ROUTINE 7          *
6 REM *
7 REM *****
8 ;
10 DEFINT A-Z
15 DIMA(25)
16 FOR T=0 TO 39
17 S=0
18 ;
20 LOCATE 0,0:PRINTSPACE$(80)
30 FOR X=0 TO 39*24 STEP 40:S=S+1
40 A(S)=VPEEK(X+T):NEXT
45 FOR V=0 TO 20:S=0
50 FOR X=80+V*40 TO 39*24STEP40:S=S+1
60 VPOKE X+T,A(S):M=M+1
65 ;
70 NEXT:NEXT
80 NEXT
90 GOTO 16

```

```

10 REM *****
20 REM *                                     *
30 REM *           WIS HET SCHERM         *
40 REM *                                     *
50 REM *           ROUTINE 8             *
60 REM *                                     *
70 REM *****
80 :
90 WIDTH40:KEYOFF
100 DIM A(40)
110 FOR X=0 TO 39
120 IF CHR$(VPEEK(X+T*40))<>" " THENA$=A
$+CHR$(VPEEK(X+T*40))
130 NEXT
140 FOR T=0 TO 21
150 FOR X=1 TO39
160 LOCATE X,T:PRINTA$:LOCATE X-1,T:PRIN
T" "
170 NEXT:LOCATEX,T:PRINT" "
180 NEXT
190 CLS

```

# Programma's en tips

In dit hoofdstukje hebben we een aantal direkt bruikbare programma's opgenomen met daarnaast ook enige tips, waaronder een addertje onder het FOR-NEXT gras.

Een van de programma's is een computerversie van de torens van Hanoi. Dit spel vergt zeer veel uren van puzzelen om tot de juiste resultaten te komen. De computer doet het echter deze keer voor u. Aanschouw het vernuft van uw MSX computer met de torens van Hanoi.

# De torens van Hanoi

De Truuks en tips reeks is niet echt bedoeld om spelletjes-achtige programma's te publiceren, maar dit keer maken we toch een uitzondering. Het volgende programma is een computerversie van de torens van Hanoi.

Een korte beschrijving van het spel:

U kunt kiezen uit maximaal 16 blokjes die variëren van groot naar klein. Het is de bedoeling om alle blokjes te verplaatsen van de linker toren naar de rechter toren, zonder dat een groter blokje op een kleiner blokje komt te liggen.

Een ideale oplossing voor moeilijke problemen vormt natuurlijk uw MSX computer. Met een beetje denkwerk ontstaat snel een goed programma om uw problemen op te lossen.

Zo ook dit keer: De computer voert voor u het HANOI-probleem uit.

Voer het aantal schijven in en aanschouw het vernuft dat schuilt achter de toetsen van uw MSX computer.

```
1 REM *****
2 REM *
3 REM *   De torens van Hanoi   *
4 REM *
5 REM *       Truuks en tips   *
6 REM *       -- deel 6 --     *
7 REM *
8 REM *****
9 ;
10 SCREEN0:WIDTH40:KEYON
11 STOP ON : ON STOP GOSUB 99 : COLOR ,
0,0 : SCREEN 2,3 : OPEN "GRP:" FOR OUTPUT AS #1
```



```

12 LINE ( 0,192)-(256,192),15
13 LINE (30,192)-(30, 75),15
14 LINE (110,192)-(110, 75),15
15 LINE (190,192)-(190, 75),15
16 :
17 REM *** INVDER ***
18 :
19 DRAW"BM 0,20": PRINT#1, "Verschuiving
en:          ";B
20 DRAW"BM 0,10": PRINT#1, "Schijven
: (MAX 16) ";
21 N=VAL(INPUT$(2)) :IF N=0 OR N>16THEN
20
22 LINE (255, 19)-(216, 10),0,BF
23 PRINT #1,N : B=0
24 LINE (216, 20)-(255, 30),0,BF
25 :
26 REM *** SPRITES WEG ***
27 :
28 FOR M=16 TO N STEP-1
29 SPRITE$(M)= ""
30 NEXT M
31 :
32 RESTORE
33 :
34 REM *** SPRITES INLEZEN EN PLAATSEN *
**
35 :
36 X=30 : Y=159
37 :
38 FOR M=N TO 1 STEP-1
39 :
40 FOR S=1 TO 13
41 S$=S$+CHR$(0):T$=T$+CHR$(0):REM
*** WAARDE = 0 ***
42 NEXT S

```

```

43 :
44 FOR S1=14 TO 16
45 READ U,V
46 S$=S$+CHR$(U);T$=T$+CHR$(V)
47 NEXT S1
48 :
49 SPRITE$(M)=S$+T$
50 S$="" : T$=""
51 PUTSPRITE M,(X,Y),15,M:REM *** BEGIN
COÖRDINATEN ***
52 Y=Y-7
53 :
54 NEXT M
55 :
56 REM *** POSITIE BEREKENING ***
57 :
58 G=Y-8 : Y=159 : ST=30 : HI=110 : EN=1
90
59 GOSUB 61: REM *** BEGIN ***
60 GOTO 19: REM *** EINDE ***
61 IF N=1 THEN GOSUB 74 : RETURN: REM **
* AFBREEKVOORWAARDE ***
62 :
63 SWAP HI,EN: REM *** WISSEL HI-EN ***
64 N=N-1 : Y=Y-7 : GOSUB 61
65 SWAP EN,HI: REM *** HERWISSEL EN-HI
***
66 N=N+1 : Y=Y+7 : GOSUB 74: REM *** SCH
IJF BEWEGEN ***
67 SWAP ST,HI: REM *** WISSEL ST-HI ***
68 N=N-1 : Y=Y-7 : GOSUB 61
69 SWAP HI,ST: REM *** HERWISSEL HI-ST
***
70 N=N+1 : Y=Y+7 : RETURN: REM *** TERUG
SPRONG 610,570 ***
71 :

```

```

72 REM *** BEWEGING V/D SPRITES ***
73 :
74 IF ST < EN THEN F=1 : I=EN ELSE F=-1
   : I=EN+2
75 M=N
76 :
77 FOR C=Y TO G STEP-1
78 PUTSPRITE M,(ST,C),15,M: REM *** SPRI
TE NAAR BOVEN ***
79 NEXT C
80 :
81 FOR D=ST TO I STEP F
82 PUTSPRITE M,(D,C),15,M: REM *** SPRIT
E HORIZONTAAL ***
83 NEXT D
84 :
85 FOR E=C TO Y
86 PUTSPRITE M,(D,E),15,M: REM *** SPRIT
E NAAR BENEDEN ***
87 NEXT E
88 :
89 REM *** AANTAL PLAATSEN AANGEVEN ***
90 :
91 B=B+1
92 LINE(255,30)-(216,20),0,BF
93 PRINT#1,B
94 :
95 RETURN
96 :
97 REM *** AFBREKEN ***
98 :
99 COLOR 15,4,4 : END
100 :
101 REM *** DATA ***
102 :
103 DATA 255,255,255,255,255,255,255,25

```

4,255,254,255,254,255,252,255,252,255  
104 DATA 252,255,248,255,248,255,248,25  
5,240,255,240,255,240,255,224,255,224  
105 DATA 255,224,255,192,255,192,255,19  
2,255,128,255,128,255,128,255, 0,255  
106 DATA 0,255, 0,254, 0,254, 0,25  
4, 0,252, 0,252, 0,252, 0,248, 0  
107 DATA 248, 0,248, 0,240, 0,240,  
0,240, 0,224, 0,224, 0,224, 0,192  
108 DATA 0,192, 0,192, 0,128, 0,128  
, 0,128, 0, 0, 0, 0, 0, 0, 0

# Speciale MSX klok

De Basic functie "TIME" maakt gebruik van de interrupt in uw computer. Deze interrupt treedt bij een Europese MSX computer 50 keer per seconde op.

Het nu volgende programma maakt op een slimme manier gebruik van de "TIME" functie. Hierdoor wordt het mogelijk om een vrij nauwkeurige klok te construeren die bovendien de nodige leuke extra's te bieden heeft.

Nieuwsgierig???

Probeer het programma dan snel...

```
1 REM *****
2 REM *
3 REM * Speciale MSX klok met *
4 REM * effecten en diverse *
5 REM * manieren van tijdswee- *
6 REM * gave.... *
7 REM * *
8 REM *****
9 :
10 COLOR 15,1:SCREEN0:WIDTH40:KEY OFF
20 LOCATE 8,5:PRINT"MSX TRUKS EN TIPS 6
30 LOCATE 8,7:PRINT" SPECIAL MSX KLOK
40 LOCATE 0,10:PRINT"Voer de juiste tijd
   in (uur,mm)"
50 INPUT UUR,MIN
60 :
70 COLOR 1,14,14:SCREEN1:KEYOFF:WIDTH32
80 :
90 LOCATE 8,8:PRINT" r" : 'GRAPH+R
100 LOCATE 9,8:PRINT"_____
   _____" : 'GR
APH+-
110 LOCATE 23,8:PRINT" r"
120 DOWN=9:FOR Y=9 TO DOWN
```

```

130 LOCATE 8,Y:PRINT"|":'SHIFT GRAPH+\
140 NEXT
150 LOCATE 8,Y:PRINT" L"
160 FOR X= 9 TO 22
170 LOCATE X,Y:PRINT"—"
180 NEXT
190 LOCATE X,Y:PRINT" J"
200 'END
210 FOR Y=Y-1 TO 9 STEP -1
220 LOCATE X,Y:PRINT"| "
230 NEXT
240 :
250 : ' *** TIJDS ROUTINE ***
260 :
270 LOCATE 16,9:PRINT": "
280 LOCATE 13,9:PRINT": "
290 TIME=SEC
300 :
310 REM *****
320 REM * *
330 REM * SECONDEN BEREKENEN *
340 REM * *
350 REM *****
360 :
370 SEC=TIME/50
380 LOCATE 16, 9:PRINT SEC
390 LOCATE 1,1:PRINT"UREN : ";UUR
400 LOCATE 1,2:PRINT"MINUTEN : ";MIN
410 LOCATE 1,3:PRINT"SECONDEN: ";SEC
420 PRINT:PRINTSTRING$(32,ASC("_"))
430 GOSUB 840
440 :
450 REM *****
460 REM * *
470 REM * MINUTEN BEREKENEN *
480 REM * *

```

```

490 REM *****
500 :
510     I = SEC MOD 60
520 :
530     IF I=0 AND SEC>=60
        THEN MIN=MIN+1  ;
        SEC=0:GOTO 290
540     LOCATE 13,9:PRINTMIN
550 :
560 REM *****
570 REM *
580 REM *     UREN   BEREKENEN   *
590 REM *
600 REM *****
610 :
620 I = MIN MOD 60
630 IF I=0 AND MIN>=60
    THEN UUR=UUR+1 :MIN=0
640 LOCATE 10,9:PRINTUUR
650 :
660 REM *****
670 REM *
680 REM * 24 UURS  OVERSCHRIJDING *
690 REM *
700 REM *****
710 :
720 IF UUR = 24 THEN UUR=0:MIN=0:SEC=0:G
    OTO250
730 :
740 GOTO 310
750 :
760 REM *****
770 REM *
780 REM * SECONDEN EN MINUTEN   *
790 REM * PLOTTEN (GEDEELD DOOR *
800 REM * 2)

```



```

810 REM *
820 REM *****
830 :
840 AA=SEC\2
850 LOCATE AA,16
860 PRINT"S"
870 BB=MIN\2
880 LOCATE BB,18
890 PRINT"M"
900 CC=UUR\2
910 LOCATE CC,20
920 PRINT"U"
930 IFAA>=30 THEN FOR X=0TO32 :LOCATEX,1
6:PRINT" ";:NEXT
940 IFBB>=30 THEN FOR X=0TO32 :LOCATEX,1
8:PRINT" ";:NEXT
950 IFCC>=12 THEN FOR X=0TO32 :LOCATEX,2
0:PRINT" ";:NEXT
960 RETURN

```

# Flitsend

Een leuke routine is de volgende, welke gebruik maakt van het feit dat op plaats 255 in de karakterset zich de cursor bevindt. Door dit feit is het mogelijk om op slimme wijze het scherm te laten "flashes".

Elke tekst die u hierna op het scherm zet, blijft altijd staan. Elke demonstratie van het effect dat hierdoor optreedt, wordt gegeven door het printen van de tekst "Flashing..." in regel 90. Experimenteer zelf wat met deze routine en de leukste dingen kunnen er uit naar voren komen.

```
1 REM *****
2 REM *
3 REM * "Flitsend" scherm *
4 REM *
5 REM *****
6 :
10 STOPON:ONSTOPGOSUB170
20 COLOR 15,1:SCREEN0:KEYOFF
30 LOCATE2,22,0:PRINT"Even geduld A.U.B."
"
40 FOR X=0 TO 24*40
50 VPOKE X,255
60 NEXT
70 :
80 LOCATE 10,11:PRINT"
90 LOCATE 10,12:PRINT" Flashing... "
100 LOCATE 10,13:PRINT"
"
110 :
120 LOCATE0,0,1
130 FOR T=1TO10:NEXT
140 LOCATE0,0,0
150 FOR T=1TO10:NEXT
160 GOTO 120
170 SCREEN0:END
```

# Omrekenen km/u naar m/s

Het omrekenen van eenheden levert vaak nogal wat rekenwerk op.

Het volgende programma stelt u in staat om een snelheid die is opgegeven in kilometers per uur, om te rekenen naar meters per seconde.

```
10 '*****
20 '*                                     *
30 '* KM/U ==>> m/s                       *
40 '*                                     *
50 '*****
60 :
100 CLS
110 WIDTH40
120 PRINT"*** OMREKENEN KM/U naar M/S ***
*
130 PRINT
140 PRINT
150 PRINT"Hoeveel kilometer per uur ??"
160 :
170 INPUT V
180 VV=V\3.6
190 PRINT:PRINT"Dat is dan "VV" meter pe
r sec."
200 END
```

## FOR...NEXT nader bekeken

Het volgende programma laat zien dat de waarde van een variabele na een FOR...NEXT loop niet gelijk is aan de variabele die verwacht zou worden. In sommige gevallen is het handig om dit te weten.

```
10 REM *****
20 REM *                                     *
30 REM *       de for...next loop         *
40 REM *                                     *
50 REM *       nader bekeken              *
60 REM *                                     *
70 REM *****
80 :
90 SCREEN0:WIDTH40:KEYOFF
100 PRINT"Dit programma bevat een for...
next
110 PRINT"loop, die van 1-5000 loopt.
120 PRINT"Na deze loop wordt de waarde v
an de"
130 PRINT"gebruikte variabele geprint, d
eze"
140 PRINT"waarde is niet gelijk aan 5000
maar..."
150 :
160 FOR X=1 TO 5000:NEXT X
170 PRINT:PRINT"...gelijk aan";X;"!!!"
180 END
```

# Funktietoetsen

Als MSX computergebruiker bent u verward als het gaat om het bedieningsgemak. MSX computers zijn namelijk voorzien van een van de beste en meest uitgebreide toetsenborden die we kunnen terugvinden op huiscomputers.

Een van de goede eigenschappen is onder andere de aanwezigheid van de funktietoetsen. Deze toetsen kunt u zelf op eenvoudige wijze definiëren door middel van Basic instructies. U kunt zelfs vanuit Basic kijken welke funktietoets er werd ingedrukt (ON KEY GOSUB).

Wat echter niet standaard mogelijk is vanuit Basic, is kijken of de funktietoetsen aan staan of uit. De funktietoetsen worden aangezet door middel van de instructie

## KEY ON

De funktietoetsen worden uitgezet door middel van de instructie

## KEY OFF

Om nu te kijken of de funktietoetsen aan of uit staan kunt u gebruik maken van een PEEK opdracht. Het adres dat de informatie bevat is: &HF3DE.

Als de inhoud van dit adres gelijk is aan nul, dan staan de funktietoetsen uit (KEY OFF).

Als de inhoud van dit adres ongelijk is aan nul, dan staan de funktietoetsen aan (KEY ON).

Het nu volgende programma demonstreert het gebruik van de hiervoor genoemde PEEK opdracht.

```
10 REM *****
20 REM *
30 REM * functie toetsen aan of *
40 REM * uit...
50 REM *
```

```
60 REM *****
70 :
80 A=PEEK(&HF3DE)
85 CLS
90 IF A=0 THEN PRINT"Function keys disabled
100 :
110 IF A<>0 THEN PRINT"function keys enabled
120 END
```





## Nederlandstalige MSX handboeken

### **MSX BASIC handboek voor iedereen**, door A.C.J. Groeneveld

Een compleet nederlandstalig handboek voor iedere MSX computergebruiker. Dit handboek omvat een volledige behandeling van het MSX-basic in het Nederlands. Het handboek geeft een antwoord op elke vraag die een programmeur, van welke scholing ook, over het MSX-basic zou kunnen stellen. De volledige syntaxisbehandeling rekent af met onzekerheden of een bepaalde schrijfwijze nu wel of niet is toegestaan. De duidelijke beschrijving geeft per sleutelwoord aan, welke de functie hiervan is. De laatste mogelijk nog aanwezig onduidelijkheden worden vervolgens door de opgenomen, zinnige voorbeelden weggenomen

ISBN 90 6398 1007

### **MSX ZAKBOEKJE** door Wessel Akkermans

Een vlot geschreven naslagwerk na of naast het handboek. U vindt er o.a. in: niet computergerichte tabellen; de MSX-BASIC instructieset; diverse tabellen die het BASIC-programmeren kunnen versnellen; de Z80 instructieset; hardware-gegevens (connectoren) en een aantal programmaatjes

ISBN 90 6398 888 5

### **MSX DISK handboek voor iedereen**, door A.C.J. Groeneveld

Handboek voor diskdrivebezitters om naast het grote handboek te gebruiken. Een zeer volledige behandeling van het disk-gebeuren zelf en de specifieke disk kommando's, uitgebreid met voorbeelden, tabellen en overzichten. Het handboek is aangevuld met interessante programma's, waaronder een tekentafelprogramma en een basisprogramma voor basisonderhoud

ISBN 90 6398 407 3

### **MSX PRAKTIJKPROGRAMMA'S** door Wessel Akkermans

Praktische programma's met waar nodig eerst een stukje theorie. Erg handig bij het maken van uw programma's. Een greep uit de onderwerpen: priemgetallen; zoeken en sorteren; trefwoordenlijsten; converteren van getallen; enz.

ISBN 90 6398 437 5

### **MSX QUICK DISK handboek voor iedereen**, door A.C.J. Groeneveld

Het handboek voor iedere QUICK DISK gebruiker. Uitvoerige behandeling van de sleutelwoorden aangevuld met duidelijke voorbeelden met listing

ISBN 90 6398 254 2

### **MSX DOS handboek voor iedereen**, door A.C.J. Groeneveld

Dit handboek geeft u op een heldere wijze een totaalbeeld van de mogelijkheden van het MSX-DOS. Ook is dit handboek voorzien van een inleiding op het begrip 'operating system' en dus echt een handboek voor iedereen

ISBN 90 6398 674 2

### **MSX2 Basic handboek**

A.C.J. Groeneveld, ISBN 90 6398 221 6, 507 pagina's, prijs f 56,50  
Wie denkt over de aanschaf van een MSX2 computer, of er al een heeft, kan niet zonder het 507 pagina's tellende MSX2 Basic handboek. Alles over MSX2 Basic, de grafische- en geluidsmogelijkheden en de computer zelf. Met 288 voorbeeldprogramma's.

### **MSX2 Disk/Dos Uitbreidingshandboek**

A.C.J. Groeneveld, ISBN 90 6398 222 4, 172 pagina's, prijs f 37,50  
Omvat een volledige behandeling van het MSX2 Disk Basic en het MSX DOS operating system, voorafgegaan door een zeer duidelijke inleiding tot de fenomenen disk en operating system. Verder praktische tabellen, duidelijke afbeeldingen en zinvolle voorbeelden. Een standaardwerk dat naast elke MSX2 Disk computer zou moeten liggen.

### **MSX2 Utility-Toepassingshandboek**

A.C.J. Groeneveld, ISBN 90 6398 223 2, 144 pagina's, prijs f 29,75  
Een verzameling programma's die voor elke MSX-er onontbeerlijk zijn. Een aantal van de mogelijkheden met deze programma's: bestandsonderhoud met lijstwerk in iedere vorm op schijf en tape, staaf- en taartdiagrammen, programma's samenstellen met sprites en geluidseffecten, binair manipuleren binnen blokken op schijf. Alle programma's in dit boek zijn geschikt voor zowel MSX als MSX2 computers.

### **MSX en MSX2 mogelijkheden – Wat kan- wat mankeert - wat kan ik met mijn computer**

Wessel Akkermans, ISBN 90 6398 606 8, 100 pagina's, prijs f 29,50  
Welk type MSX computer heb ik? Doen mijn joysticks het wel goed? Werken alle toetsen naar behoren, maakt mijn schijfveenheid lees- of schrijffouten? Op al deze vragen krijgt u antwoord bij het uitvoeren van het in dit boek beschreven programma. Eigenlijk gaat het om een groot aantal afzonderlijke programma's, die als subroutine zijn opgenomen binnen een groot hoofdprogramma, van waaruit de subroutines gemakkelijk zijn te starten. Werkt zowel op MSX als op MSX2.  
De beschreven programmatuur is leerzaam voor elke MSX-er, en bovendien een waardevol gereedschap bij het onderzoeken van de goede werking van MSX en MSX2 computers voor alle gebruikers.

### **MSX Basic voor kinderen deel 1**

H.C. de Heer, ISBN 90 6398 084 1, 110 pagina's prijs f 19,50  
De bedoeling is kinderen te vertellen hoe zij een programma kunnen begrijpen, maar ook hoe zij zelf kunnen leren programmeren. Alle onderwerpen die niet beslist nodig zijn worden niet behandeld. Allereerst echter maakt de lezer kennis met het verschijnsel computer en vervolgens met de taal van de computer. Voor kinderen vanaf 8 jaar.  
deel 2 ISBN 90 6398 304 2, prijs f 24,50

## **MSX TRUUKS EN TIPS**

Als MSX-er moet je wel even wat tijd uittrekken om het MSX Basic te leren kennen, maar beheers je het eenmaal, dan kun je met een paar instructies de meest ongelofelijke dingen doen.

Programmeren is en blijft echter een kunst. Het is niet genoeg om alle woorden van een programmeertaal uit het hoofd te kennen. Nee, de kunst is om vanuit deze bouwstenen een goed functionerend en foutloos programma te schrijven. In het op een korte en krachtige wijze oplossen van problemen herkent men de ware programmeur.

En achter alle boekjes uit de serie MSX Truuks en tips gaan zulke 'ware programmeurs' schuil. MSX Truuks en tips: boordevol waardevolle truuks, tips, routines en programma's, soms in machinetaal; in elk geval altijd razend interessant. Voor iedere MSX-er een bron van lering en vermaak. Bij ieder deel opnieuw.

deel 1 ISBN 90 6398 900 8

deel 5 ISBN 90 6398 745 5

deel 2 ISBN 90 6398 340 9

deel 6 ISBN 90 6398 879 6

deel 3 ISBN 90 6398 910 5

deel 7 ISBN 90 6398 789 7

deel 4 ISBN 90 6398 897 4

## **MSX SOFTWARE PLUS**

### **MSX INTROTAPE** door A.C.J. Groeneveld

Heeft u nog maar net een MSX computer en wilt u graag weten wat hij kan en hoe u kunt leren hem te programmeren? Deze cassette introduceert MSX op een uiterst vriendelijke en onderwijzende manier. U krijgt volledige instructies om het programma te starten. Daarna volgt een demonstratie van de MSX mogelijkheden: tekenen met sprites, geluidsmogelijkheden... Tenslotte twee 'les' gedeelten. In anderhalf tot drie uur weet u wat de MSX computer is, wat hij kan, en heeft u haast ongemerkt al wat regels geprogrammeerd.

ISBN 90 6398 148 1

### **MSX SCRIPT** door Ton Weijters

Een volledig menugestuurde nederlandsstalige tekstverwerker, geschikt om efficiënt grotere of kleinere teksten te bewerken. De volledige pagina-indeling wordt door het programma verzorgd, evenals de telling, de toptitel en het invullen van regels. Ook korrigeren, zoeken, string-substitutie, blokken tekst verplaatsen, kopiëren of verwijderen, onderstrepen en vet zetten, is mogelijk.

ISBN 90 6398 189 9

### **MSX DRAWS** door A.C.J. Groeneveld

Een MSX tekenprogramma, waarmee u al binnen 10 minuten uw eerste tekening kunt maken. Draws werkt erg vriendelijk en maakt gebruik van alle grafische mogelijkheden van uw MSX computer, voor technische of creatieve tekeningen. Het effectief bereik is ruim 30000 bij 30000 puntjes met mogelijkheden als o.a. cirkels, krommen, verkleinen, verschuiven, verdraaien en substitueren.

ISBN 90 6398 754 4



## **MSX LEERBOEKEN**

door Wessel Akkermans en Piet den Heijer

De serie MSX leerboeken geeft een complete cursus MSX-BASIC programmeren, in drie delen. Deze leerboeken zijn gericht op de beginnende programmeur. De moeilijkheidsgraad van de leerstof wordt dan ook slechts geleidelijk hoger. De gebruikte voorbeelden zijn zo praktisch mogelijk gekozen. Hierdoor kunnen al in een vroeg stadium bruikbare programma's worden gemaakt. Dit zal de lezer/leerling er toe aansporen om verder te gaan. Aan het eind van ieder deel is een groot voorbeeldprogramma opgenomen. Dit programma laat zien waartoe de lezer/leerling na bestudering van het betreffende leerboek in staat zal zijn.

Bij ieder leerboek is een afzonderlijk –Opdrachten en uitwerkingen– boekje te verkrijgen. In deze boekjes staan, in volgorde van de hoofdstukken uit het leerboek, vragen en opdrachten met antwoorden en uitwerkingen. Een unieke serie leerboeken voor een ieder die meer over MSX wil weten en het betere werk met zijn computer wil maken.

MSX Basic leerboek deel 1 - ISBN 90 6398 649 1

Opdrachten bij deel 1 - ISBN 90 6398 596 7

MSX Basic leerboek deel 2 - ISBN 90 6398 769 2

Opdrachten bij deel 2 - ISBN 90 6398 556 8

MSX DOS leerboek deel 3 - ISBN 90 6398 519 3

Opdrachten bij deel 3 - ISBN 90 6398 516 9

## **MSX Verder uitgediept** door H. Klopper

Eindelijk een Nederlandstalig boek over het altijd in de mist gehulde onderwerp – PEEKS EN POKES. In dit boek staan alle belangrijke RAM en VRAM adressen. De video chip en zijn registers worden volledig uitgelegd. Maar ook hoe men een machinetaal programma van cassette naar disk kan schrijven. Bovendien een diskloader utility en een uiterst geavanceerde programma beveiliging. Tenslotte zijn er een aantal interessante programma's opgenomen, waaronder een wereldkaart, waarmee verder kan worden geëxperimenteerd. Elke MSX gebruiker kan in dit boek iets van zijn gading vinden en nieuws leren.

ISBN 90 6398 447 2

## **MSX Machinetaal handboek** door H. Klopper en M. Le Belle

Hoewel een MSX computer over een krachtig Basic beschikt, is het toch handig tijdens het programmeren de grondbeginselen van machinetaal te kennen. Daarvoor is dit boek een goede gids. De zaken worden niet puur theoretisch maar ook aan de hand van duidelijke voorbeelden, die direct bruikbaar zijn, uitvoerig uitgelegd. Enkele onderwerpen zijn verder – scroll routine –machinetaal software (ook in disk Basic) op cassette zetten –disassembler –Z80 assembler instructies –lijst van ROM-routines –alle hook-adressen –bespreking van Basic tokens en een compleet token-overzicht. Het handboek voor iedere MSX programmeur die zijn computer ten volle wil benutten.

ISBN 90 6398 735 8





## **truuks en tips deel 6**

De MSX programmeur heeft de beschikking over een ruim scala van mogelijkheden. Deze mogelijkheden zijn echter niet zomaar om te zetten in een goed programma. De MSX Truuks en tips reeks heeft als doel u in dit opzicht een ruggesteuntje te bieden.

De meeste truuks en tips kunnen in zelf ontwikkelde programma's verwerkt worden, zodat ze nog beter voor de dag komen. Tevens bevat dit boekje weer een aantal machinetaalroutines die voor de nodige geheimzinnige effecten zorgen.

Er is vrijwel geen kennis van de MSX Basic nodig om de uitgekende programma's in dit deel te kunnen gebruiken. Het is daarom niet noodzakelijk om de inhoud van deel 1 tot en met 5 te kennen.

Een kleine greep uit de inhoud:

- Letters of zinnen laten knippen
- Een Basic programma m.b.v. BSAVE wegschrijven
- Basic programma beveiliging
- Alternatieven om het scherm te wissen
- De torens van Hanoi
- Karaktersets wisselen
- Machinetaalroutine die de voor- en achtergrondkleur altijd verschillend houdt
- Machinetaalroutine die een reset onmogelijk maakt

Dit is slechts een selectie uit de vele uitgekookte truuks en tips die vooral dit deel weer bevat.

MSX Truuks en tips deel 6 geeft u meer dan ooit het gevoel de computer volledig onder controle te hebben.