

Tracer

onmisbaar voor
MSX Basic programmeurs



Copyright 1992 by
MSX Software Team

```
MSX BASIC versio
Copyright 1990 b
21564 Bytes free
disk BASIC versi
Ok
10 for f#1 to 10
20 print f; "x 10 = ", f*10
30 next #
run
1 x 10 = 10
2 x 10 = 20
3 x 10 = 30
4 x 10 = 40
5 x 10 = 50
6 x 10 = 60
30/20/30/20/30/20/30/20/30/20/
PRINT F; "x 10 ="; F*10
eX1 7
seT N cLR N every new Line
[1] ----- [2] ----- [3] -----
cNt 1 sEc ----- Key Ct/Sh
Clr Dsp stP ? eVn Inf Mem tYp
Off Hot Ful Stm Lin Brk trUe
color list run
```

Tracer

MSX Software Team

```
eX1  
seT N cLR N every new Line  
[1] ----- [2] ----- [3] -----  
cNt      1 sEc ----- Key Ct/Sh  
Clr Dsp stP ? eVn Inf Mem tYp  
Off Hot Ful Stm Lin Brk trUe
```

Tracer floppy label

Tracer, een volgprogramma voor MSX Basic

| EVENT TRAP | ACTIVITY | key1 | key2 |
|------------|----------|------|----------|
| key1 | 100stop | key2 | 200 on |
| key3 | ---- off | key4 | ---- off |
| key5 | ---- off | key6 | ---- off |
| key7 | ---- off | key8 | 300 off |
| key9 | 400 off | key0 | ---- off |
| stop | ---- off | prt | ---- off |
| trg0 | ---- off | trg1 | ---- off |
| trg2 | ---- off | trg3 | ---- off |
| trg4 | ---- off | intr | 100 off |

Volg uw Basic programma's
(of die van anderen!)

- Alle informatie gemakkelijk toegankelijk
- Ook over grafische schermen
- Schermuitvoer wordt niet verstoord
- Oproepbaar met Hotkey
- TRON compatible
- Step mode, single stappen door Basic
- Meerdere regelnummers tegelijk in beeld
- Kost geen Basic geheugen (afgezien dat wat MemMan gebruikt)
- Direct aanspreekbaar met CMD commando's
- Instelbare Hotkey om conflicten te vermijden
- Op tekstschermen blijft originele scherm zichtbaar
- 3 dynamisch instelbare Breakpoints
- Automatisch verschijnen en verdwijnen
- Overzicht gebruikt geheugen
- VDP(24) bestendig
- Verplaatsbaar window
- De Basic stack wordt zichtbaar
- Eigen, snelle, detokenizer
- Knippen/inverse (screen 0) bestendig
- Werkt vlekkeloos met andere MemMan TSR's
- Overzicht event traps
- Teller voor regels, breakpoints en statements
- Ingebouwde expressie evaluator
- 6 verschillende 'automodes'
- voorbereid op MemMan 2.4...

| EXPRESSION EVALUATOR |
|-----------------------|
| ex1 12 |
| 12 |
| ex2 MID\$("TEST",2,2) |
| "ES" |
| ex3 SIN(TAN(2)) |
| -.81720966124778 |

| OTHER INFORMATION | | | |
|-------------------|-----|-----------|-----|
| ML1 | ML2 | ML3 | ML8 |
| ML4 1234h | ML5 | ML6 | ML9 |
| ML7 | ML8 | ML9 | |
| MaxFiles: | | | 1 |
| CursorX: | | | 1 |
| CursorY: | | | 2 |
| DATAline: | | | 0 |
| NEXT F | 10 | DOT line: | 10 |

| USED MEMORY OVERVIEW | Area | start | end | length |
|----------------------|-------|-------|-------|--------|
| Program | 8001h | 802Fh | | 47 |
| Variables | 8030h | 8071h | | 66 |
| Array's | 8072h | | | 8 |
| Free/Stack | 8072h | D44Ch | 21467 | |
| Strings | D44Dh | D514h | | 200 |
| TOTAL | 8001h | D514h | | 21700 |

| Overview DEF STR/INT/SNG/DBL |
|------------------------------|
| Strings (#) |
| ----- |
| Integers (%) |
| -----M-----S-----X-- |
| Single precision (!) |
| ----- |
| Double precision (#) |
| ABCDEFGHIJKL-NOPQR-TUUV-VZ |



Tracer

MSX Software Team

```
ex3  
del H oIR N every new Line  
[1] ----- [2] ----- [3] -----  
cnt 1 sec  
clr Dep stP ? eUn Int floa tYp  
off Hot Ful Str Lin Brk true
```

Tracer, het volgprogramma voor MSX Basic is geschreven door Robbert Wethmar, gedurende eind 1991 en begin 1992. Er zitten naar schatting meer dan 300 uren werk in. En dan hebben we het nog niet over het maken van de handleiding en het testen, dat door een aantal mensen zeer grondig gedaan is. Misschien vindt u het bedrag dat u voor dit programma betaalde te veel en dat iedereen over deze TSR zou moeten kunnen beschikken. Voor niets. U zal de enige niet zijn. De Tracer is niet beveiligd tegen kopiëren. Zelfs deze handleiding kan zo onder het kopieerapparaat. Ook als u dat doet zult u helaas de enige niet zijn. Bedenkt u zich echter wel dat dergelijke praktijken de ontwikkeling van MSX software niet bevorderen, maar juist tegengaan.

Vandaar dat we u vragen *niet* mee te werken aan de verspreiding van MST producten in het algemeen en dit programma in het bijzonder. Maak Sysops erop attent als er MST producten in hun lijsten verschijnen. Spreek vrienden en collega's aan als ze illegale kopieën gebruiken. Alleen op die manier blijft het leuk programma's voor de MSX te ontwikkelen!

Inhoudsopgave

| | |
|---|-----------|
| Inleiding | 4 |
| De aanleiding | 5 |
| Het hoofdscherm | 6 |
| De opbouw van het hoofdscherm | 7 |
| De CMD commando's | 8 |
| CMD TRON | 8 |
| CMD TROFF | 8 |
| CMD TRACER | 8 |
| CMD CLEAR TRACER | 9 |
| Hoe, wat en waar | 10 |
| De editors | 11 |
| De automodes | 12 |
| O (Off) Automode off | 12 |
| H (Hot) Hotkey | 12 |
| F (Ful) At Full window | 12 |
| S (Stm) Every Statement | 12 |
| L (Lin) Every Line | 13 |
| B (Brk) Breakpoint | 13 |
| U (TrUe) if EXPR2=TRUE | 13 |
| Commando's in het hoofdscherm | 14 |
| 1, 2 en 3: het instellen van de Breakpoints | 14 |
| T (seT): instellen eigen karakterset | 14 |
| R (clR): instellen eigen kleuren | 15 |
| N (cNt): teller bij verschijnen | 15 |
| E (sEc): automatisch verdwijnen | 15 |
| K (Key): instellen Hotkey | 15 |
| C (Clr): wissen alle instellingen | 16 |
| D (Dsp): inschakelen originele display | 16 |
| P (stP): door het programma 'stappen' | 16 |
| ?: versie informatie | 16 |
| Hulpschermen | 17 |
| M (Mem): Used Memory Overview | 17 |
| Y (tYp): Overview DEF STR/INT/SNG/DBL | 17 |
| V (eVn): Event trap Activity | 18 |
| X (eXp): Expression evaluator | 18 |
| I (Inf): Other information | 20 |
| Problemen & oplossingen | 21 |
| Tracer en MemMan | 22 |
| Het laden van Tracer | 22 |
| Index | 23 |

Inleiding

Tracer betekent letterlijk volger. Het gaat hier dan ook om een programma om iets mee te volgen, en wel de uitvoering van een MSX Basic programma. Met behulp van Tracer kunnen de verrichtingen van MSX Basic op de voet gevolgd worden. Indien gewenst kan voor elke instructie het Tracer window verschijnen, zelfs als er op grafische schermen gewerkt wordt.

Als het Tracer window op het scherm staat is in een oogopslag te zien welke instructie er uitgevoerd zal worden en op welke regel die staat. Maar ook een lijst van regels die inmiddels uitgevoerd zijn ontbreekt niet. Bovendien is er nog allerlei andere informatie op te vragen. Bijvoorbeeld hoe het geheugen verdeeld is over het programma, de variabelen en de strings. Of bijvoorbeeld welk type de verschillende variabelen insteld zijn. En of er een ON KEY GOSUB actief is. Of welke USR routines er op dit moment gedefinieerd zijn. Kortom: te veel om op te noemen.

Door de informatie van de Tracer wordt het veel makkelijker om fouten te zoeken in Basic programma's. De loop van het programma kan immers op de voet gevolgd worden, zonder hem te verstoren. Ook is het met de Tracer eenvoudiger vreemde stukken Basic te doorgronden. Gewoon door stap voor stap te kijken wat ze doen.

Om beide zaken nog makkelijker te maken, is het mogelijk het window niet bij elke instructie te laten verschijnen, maar alleen als er ook daadwerkelijk behoefte aan is. Zo kan Tracer actief worden als er een bepaald regelnummer bereikt wordt, als er een bepaalde toetscombinatie wordt ingedrukt of bijvoorbeeld pas als er 100 instructies zijn uitgevoerd. Je kunt het bijna zo gek niet bedenken of de Tracer kan het.

Eenvoudig is het programma (mede) daardoor niet. Waarschijnlijk zult u deze handleiding vooral in het begin regelmatig raadplegen. De informatie in het window zelf is namelijk beperkt, omdat we het window zelf niet te groot wilden maken. Ook andere zaken, zoals de grootte van het TSR file speelden bij die keuze een rol.

Robbert Wethmar,
programmeur Tracer

De aanleiding

Op elke standaard MSX werkt het commando TRON, waarmee de regelnummers van elke uitgevoerde regel op het scherm getoond kunnen worden. Op die manier is het natuurlijk mogelijk de verrichtingen van een Basic programma te volgen, maar er zijn wel een aantal nadelen. Nadelen, die de Tracer door een andere aanpak niet heeft:

- TRON verstoort de schermuitvoer

Daar heeft de Tracer geen last van. Wanneer de Tracer actief wordt verschijnt er een rechthoek op het scherm met alle informatie zoals op de volgende pagina te zien is. Het is nu te allen tijde mogelijk de Tracer 'weg te drukken', waarna het scherm weer volledig in de originele staat hersteld wordt.

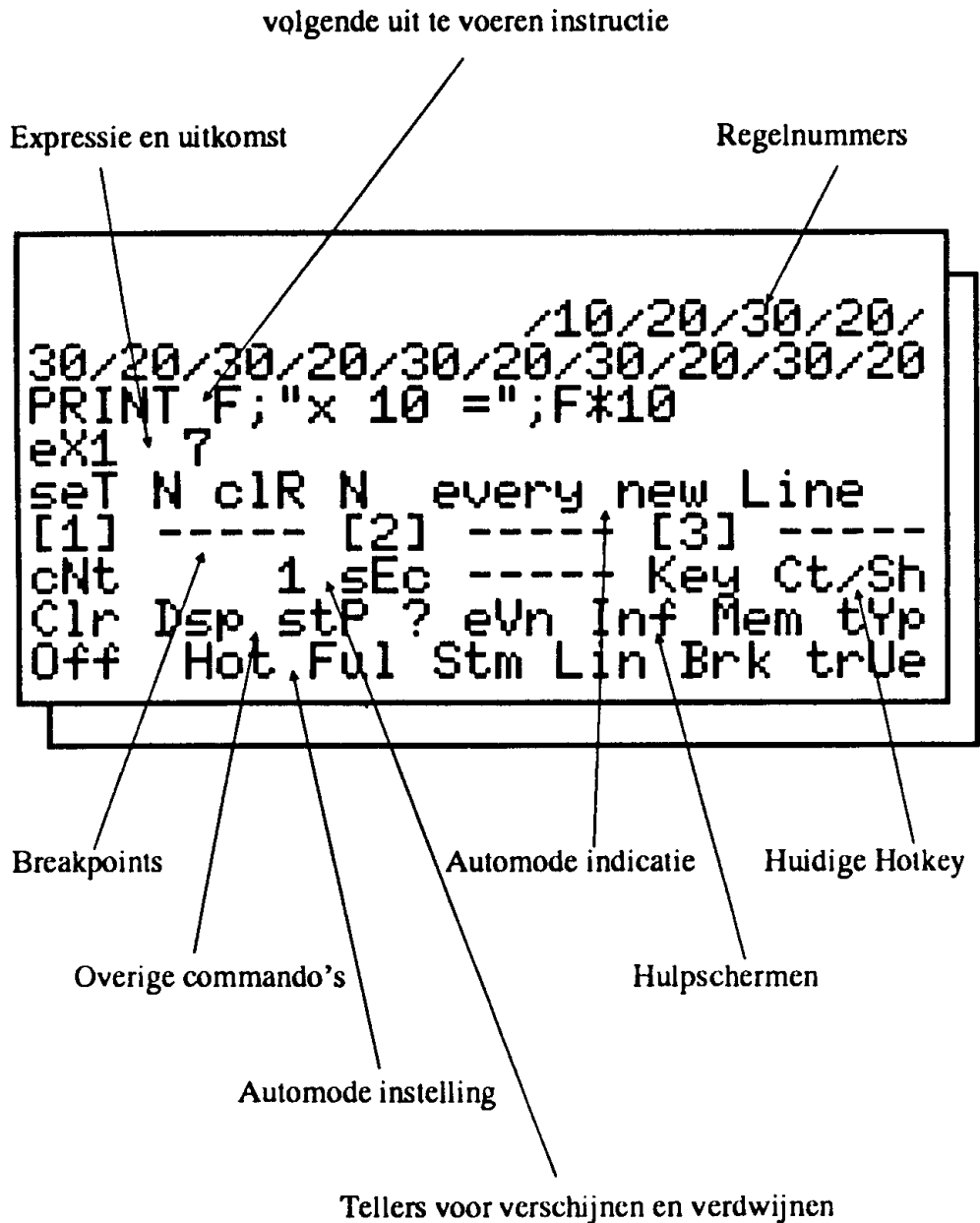
- TRON werkt niet in grafische schermen

Ook daar heeft de Tracer geen last van, hoewel ook de Tracer zijn 'window' alleen maar in een tekstscherf (screen 0 of screen 1) kan tonen. Wanneer er een grafisch scherm actief is op het moment dat de Tracer moet verschijnen zal het programma automatisch eerst screen 0 activeren. Op die manier kan de Tracer gewoon gebruikt worden. na het verlaten van de Tracer wordt het originele grafische scherm hersteld.

- TRON geeft behalve het regelnummer geen informatie

Ook hier zijn de verschillen met Tracer hemelsbreed. Als de Tracer eenmaal actief is, is niet alleen een regelnummer zichtbaar, maar ook een schat aan andere informatie. Zo is het opvragen van de waarde van een bepaalde variabele, het bekijken van de hoeveelheid vrij geheugen of bijvoorbeeld het bekijken van de default variabelen-typen geen enkel probleem.

Het hoofdscherm



De opbouw van het hoofdscherm

Op de voorgaande pagina staat het hoofdscherm van de Tracer afgebeeld. We zullen de verschillende onderdelen van dit – op het eerste gezicht nogal verwarrende – scherm nu kort doorlopen.

De eerste drie regels zijn gereserveerd voor regelnummers. Telkens als er wordt begonnen met de uitvoering van een nieuwe regel wordt er rechtsonder een regelnummer toegevoegd en schuiven de reeds aanwezige nummers door. Hierdoor wordt de 'geschiedenis' van het Basic programma zichtbaar. Het is altijd duidelijk hoe een programma op een bepaald punt terecht is gekomen.

De vierde regel van boven toont altijd de Basic instructie die op het punt staat uitgevoerd te worden. Het is namelijk ook mogelijk de Tracer bij elke instructie, dus meerdere keren per regel te laten verschijnen. Dan is een regelnummer alleen niet meer voldoende om aan te geven waar de Basic interpreter precies gebleven is.

Direct onder het eerstvolgende Basic statement staan de letters 'eX1'. Dat de X daarbij een hoofdletter is, is geen toeval: alle hoofdletters in het Tracer scherm zijn commandoletters. Door ze in te drukken terwijl de Tracer zichtbaar is wordt een commando geactiveerd. De X in dit voorbeeld staat voor eXpressie, het is namelijk mogelijk vanuit de Tracer Basic expressies te evalueren – ofwel: sommetjes uit te rekenen. Het resultaat van één van de expressies die ingevoerd kunnen worden is in het hoofdscherm te zien achter 'eX1', in dit voorbeeld dus 7. Er is een hele regel voor gereserveerd omdat het ook om een groot getal of zelfs een string zou kunnen gaan. In dit geval is de expressie echter niets anders dan de variabele F, waarvan de waarde dan ook getoond wordt.

Het onderste deel van het Tracer window bevat een – nogal cryptisch – overzicht van de mogelijke commando's en een overzicht over enkele instellingen. Wat ze precies doen en betekenen wordt verderop in deze handleiding behandeld.

De CMD commando's

Behalve automatisch kan de Tracer ook direct vanuit Basic geactiveerd worden met behulp van een aantal CMD commando's. Op deze manier kan de Tracer actief gemaakt of juist uitgeschakeld worden.

CMD TRON

Als de automode van de Tracer uitgeschakeld is door een CMD TROFF commando of door het O commando, kan hij alleen met behulp van CMD TRON of CMD CLEAR TRACER weer aangezet worden. CMD TRON doet niets anders dan de automode op 'every line' instellen, zodat de Tracer automatisch zal verschijnen als er aan een nieuwe Basic regel begonnen wordt.

CMD TROFF

Dit commando schakelt de Automode uit. De Tracer zal hierna niet meer automatisch verschijnen, maar slechts met een CMD TRACER commando opgeroepen kunnen worden.

De Tracer kan met het MemMan commando (CMD) TK volledig uit het geheugen verwijderd worden. Vanuit Basic gaat dat met:

CMD TK ("RWL Tracer")

LET OP: In MemMan 2.31 zit nog een kleine beperking in CMD TK. De Tracer wordt wel goed verwijderd, maar in de 'hook' in het systeemgebied blijft een sprong naar MemMan aanwezig. Hierdoor blijft Basic traag werken, bij iedere instructie wordt namelijk heel even MemMan actief. In de praktijk geeft dit echter geen problemen.

Wie ook de sprong naar MemMan wil verwijderen om Basic op de maximale snelheid te laten lopen kan de Tracer vanuit MSXDOS verwijderen met TK.COM. Dat programma heeft de genoemde beperking namelijk niet.

In de volgende versie van MemMan zal dit 'probleem' opgelost zijn.

CMD TRACER

Met dit commando kan de Tracer direct aangeropen worden. Dat werkt natuurlijk zowel in direct mode als vanuit een programma. Dat laatste zou handig kunnen zijn om de Tracer te activeren op een cruciaal punt in het programma. Om te voorkomen dat de Tracer altijd geactiveerd wordt – ook als u helemaal niet in het Tracen van het programma geïnteresseerd bent – kan natuurlijk gebruikt gemaakt worden van het IF statement, bijvoorbeeld:

IF DEBUG = 1 THEN CMD TRACER

Door nu voorin het programma de variabele DEBUG de waarde 1 te geven wordt de Tracer aangeroepen, ook als er meerdere van dergelijke IF-statements in het programma voorkomen.

Nog mooier is het misschien om gebruik te maken van de in MemMan 2.3 ingebouwde functies voor TSR's. Daarmee kan de aanwezigheid van een TSR getest worden. Bijvoorbeeld:

IF ATTR\$ FT("RWL Tracer") THEN CMD TRACER

Met bovenstaande instructie wordt de Tracer slechts aangeroepen als de TSR ook daadwerkelijk aanwezig is. Combinaties zijn natuurlijk ook mogelijk.

CMD CLEAR TRACER

Dit commando wist alle instellingen van de Tracer. Alle ingevoerde expressies, breakpoints en tellers worden gewist, de regelnummers worden verwijderd, kleuren en karakterset worden niet meer door de Tracer ingesteld. Kortom: alles is weer precies zoals het was toen de Tracer ingeladen werd.

Hoe, wat en waar

De Tracer werkt dus altijd in een window, een stukje scherm dat verschijnt zodra de Tracer actief wordt en daarna weer spoorloos verdwijnt. Dit window kan te allen tijde met de cursortoetsen bewogen worden. Op die manier is het mogelijk de uitvoer van het Basic programma toch zichtbaar te maken; het Tracer window beslaat namelijk niet het hele scherm.

De keuze voor het window was een bewuste. Ten eerste maakt dat het mogelijk (een deel van) de uitvoer van het Basic programma en de informatie van de Tracer *tegelijktijd* in beeld te hebben. Dat kan in veel gevallen zeer waardevol zijn. De andere reden heeft eigenlijk niets met Tracer, maar meer met programmeren in het algemeen te maken. Met de komst van MemMan 2 zijn TSR's op de MSX mogelijk geworden, waarvan er vele gebruik zouden willen maken van soortgelijke windows als de Tracer gebruikt. De Tracer is in dat opzicht een soort testcase. Gewoon kijken wat er mogelijk is.

Alle commando's die er in de Tracer gegeven kunnen worden bestaan uit één enkele toetsdruk, meestal een letter. Dat maakt de bediening niet eenvoudig, maar met enige ervaring wel snel. Om het geheel nog enigszins werkbaar te maken zijn te commandoletters vrijwel altijd op het scherm zichtbaar als hoofdletters. De – meestal drie letters lange – afkortingen staan meestal voor een commando.

Omdat er echter met ruimte gewoekerd moest worden staan er ook een aantal werkende toetsen niet op het scherm. De belangrijkste daarvan is de *spatiebalk* die net als Escape en Ctrl-Stop gebruikt kan worden om de Tracer te verlaten. Als de Tracer verlaten wordt terwijl er een Basic programma loopt, wordt automatisch de volgende instructie uitgevoerd. Wanneer er een automode is ingesteld – en dat is standaard het geval – zal het window na enige tijd automatisch opnieuw verschijnen, tenzij de uitvoering van het programma om welke reden dan ook stopt.

De editors

Op verschillende plaatsen in de Tracer is het mogelijk gegevens in te voeren. Het kan daarbij gaan om getallen of regelnummers, maar ook om bijvoorbeeld een expressie. Wie MSX Basic gewend is zal weinig moeite hebben met het editten van gegevens in de Tracer. De betekenissen van de toetsen zijn namelijk rechtstreeks van Basic afgeleid:

| | |
|---------------|--|
| Shift-Home | Maakt het hele invoerveld schoon. |
| Home | Beweegt de cursor naar het begin van het veld. |
| BS | Wist het laatst ingevoerde teken. |
| Del | Wist het teken waar de cursor op staat. |
| Cursor links | Beweegt de cursor een stap naar links. |
| Cursor rechts | Beweegt de cursor een stap naar rechts. |

Bij het editten van de Breakpoints zijn ook de cursortoetsen omhoog en omlaag actief om naar de vorige of volgende regel in het Basic programma te stappen.

De automodes

De Tracer kan op twee manieren actief worden: via een CMD TRACER commando vanuit Basic of automatisch als er een Basic programma loopt. Die laatste mogelijkheid zal in de praktijk waarschijnlijk veel gebruikt worden. Er zijn dan ook verschillende manieren om de Tracer automatisch te laten verschijnen die we hier stuk voor stuk zullen behandelen.

O (Off) Automode off

Een beetje flauw: in deze stand zal de Tracer *niet* automatisch verschijnen. Wanneer deze stand ingeschakeld wordt verschijnt de vraag of u zeker van uw zaak bent. Als de Tracer uit staat kan hij namelijk alleen maar weer geactiveerd worden door een CMD TRACER commando, of indirect met behulp van CMD TRON of CMD CLEAR TRACER.

H (Hot) Hotkey

In feite staat de automode in deze stand ook uit. De Tracer zal echter wel vlak voor de uitvoering van elke Basic instructie testen of de Hotkey (standaard Ctrl-Shift) ingedrukt is. Als dat het geval is zal het Tracer window verschijnen. De Hotkey kan met het commando K veranderd worden. Overigens werkt de Hotkey ook in alle volgende automodes.

LET OP: de Tracer controleert de Hotkey alleen vlak voor de uitvoering van een Basic commando. Daardoor is het bijvoorbeeld onmogelijk de Tracer op te roepen tijdens de uitvoering van een INPUT opdracht.

F (Ful) At Full window

Soms gaat het er niet zozeer om wat er precies gebeurt maar volgens welke route Basic door het programma loopt. Ofwel: welke regelnummers worden er precies uitgevoerd? Door de automode op 'At Full window' te zetten verschijnt de Tracer elke keer als het gebied voor de regelnummers dreigt over te lopen. Bij het verlaten van de Tracer met de spatiebalk worden in deze mode alle regelnummers uit de Tracer gewist, om ruimte te maken voor de volgende reeks. Op die manier kunt u precies zien welke regels er uitgevoerd zijn.

S (Stm) Every Statement

In deze mode verschijnt de Tracer vlak voor elk Basic commando dat uitgevoerd wordt. Zodoende mist u niets van de uitvoering van het programma. Nadeel is natuurlijk dat de Tracer erg vaak verschijnt, ook in delen van het programma die totaal oninteressant zijn.

L (Lin) Every Line

Wanneer deze mode geselecteerd is zal de Tracer verschijnen vlak voordat de eerste instructie van een regel uitgevoerd wordt. De Tracer verschijnt dan dus voor elke regel precies een keer.

B (Brk) Breakpoint

Met behulp van deze automode kunt u de Tracer bij de uitvoering van een bepaald regelnummer laten verschijnen. Met behulp van de cijfertoetsen 1, 2 en 3 kunnen drie breakpoints ingesteld worden. Zodra een van de ingestelde regelnummers bereikt wordt zal het Tracer window verschijnen. Een bijzondere eigenschap van de editor voor breakpoints is de mogelijkheid om met de cursortoetsen omhoog en omlaag door de regelnummers van een Basic programma te lopen. Op die manier voorkomt u dat er er een breakpoint ingesteld wordt op een regelnummer dat niet bestaat. Met name wanneer er tijdens de ontwikkeling van een programma regelmatig regels tussengevoegd worden is dit een zeer nuttige optie.

U (TrUe) if `EXPR2=TRUE`

Op deze manier is het mogelijk de tracer te laten verschijnen als aan een willekeurige voorwaarde voldaan wordt. De voorwaarde kan elke Basic expressie zijn, die u invult in expressie 2 in de Expression Evaluator. Wanneer die expressie bijvoorbeeld ingesteld wordt op: `SC=1000` verschijnt er als resultaat een 0 (onwaar) of -1 (waar), afhankelijk van de waarde van de variabele SC op dat moment. Met behulp van deze expressie is het mogelijk de Tracer automatisch te laten verschijnen als SC een waarde groter of gelijk aan 1000 gebreikt. Dit wordt voor elke Basic instructie getest.

Commando's in het hoofdscherm

De commando's die in het hoofdscherm geaccepteerd worden zijn in een aantal groepen te verdelen:

- het instellen van de automode
- het overschakelen naar de hulpschermen
- het instellen van de Breakpoints
- de Tracer controle commando's
- de Trace-commando's

De eerste categorie is reeds in een apart hoofdstuk besproken en ook aan elk van de hulpschermen (te bereiken met de toetsen X, V, I, M en Y) zal een apart hoofdstuk geweid worden. Hier gaan we dan ook in op de resterende drie groepen.

1, 2 en 3: het instellen van de Breakpoints

Wanneer de automode ingesteld wordt op 'at Breakpoint' zal de Tracer verschijnen als een bepaalde regel wordt bereikt. Er kunnen tegelijkertijd drie regels ingesteld worden waarop gecontroleerd wordt: de breakpoints 1, 2 en 3.

Deze Breakpoints zijn in het hoofdscherm constant te zien en kunnen na een druk op de corresponderende cijfertoets worden ingesteld. In de editor werken de cursortoetsen omhoog en omlaag op een bijzondere manier: ze laten het regelnummer verspringen naar respectievelijk de vorige en de volgende regel in het Basic programma. Daardoor wordt het bijvoorbeeld eenvoudig mogelijk een Breakpoint te zetten op de eerste regel na een NEXT instructie: gewoon het regelnummer van de NEXT intikken en vervolgens één regel verder stappen!

T (seT): instellen eigen karakterset

Als een Basicprogramma een eigen karakterset instelt kan het zijn dat het Tracer window onleesbaar wordt. In dat geval is het mogelijk de Tracer de standaard MSX karakterset te laten installeren, zodat de leesbaarheid gegarandeerd wordt. Bij het verlaten van de Tracer wordt de orininele set vanzelfsprekend weer hersteld.

Nadeel is dat het instellen van de MSX set nogal wat tijd kost, en in extreme gevallen zelfs fout zou kunnen gaan. Bijvoorbeeld wanneer er een combinatie van Basic en machinetaal die de VDP direct aanspreekt gebruikt wordt.

R (cIR): instellen eigen kleuren

Als een Basicprogramma de voor- en achtergrond kleur aan elkaar gelijk maakt is het Tracer window onzichtbaar. Door de optie cIR (van color) op 'Y' te zetten zal de Tracer telkens als het window verschijnt eigen kleuren instellen, namelijk witte letters op een zwarte achtergrond. Daardoor is het Tracer window altijd zichtbaar.

LET OP: het is helaas niet altijd mogelijk om de originele kleuren goed te herstellen. Dat heeft te maken met het feit dat de palletregisters van de VDP niet uitgelezen kunnen worden. Om dat probleem te omzeilen wordt de palletinformatie door MSX Basic ook in het VRAM opgeslagen, en wel op een vast adres in de *huidige* pagina. Daardoor kan de Tracer na een SET PAGE commando het spoor kwijt raken. Hiervoor is helaas geen afdoende oplossing bekend.

N (cNt): teller bij verschijnen

Soms is het niet nodig de Tracer elke instructie of elke regel te laten zien, maar bijvoorbeeld om de vier instructies. In die gevallen kan de counter gebruikt worden om dat aan te geven. Een mogelijke toepassing is het volgen van FOR-NEXT lussen.

E (sEc): automatisch verdwijnen

Wanneer een programma stap voor stap doorlopen moet worden zonder dat u telkens het Tracer window weg wilt drukken kunt u hier instellen na hoeveel tijd de Tracer automatisch weer moet verdwijnen. De tijd kan opgegeven worden in tienden van seconden.

Wanneer er een tijd ingesteld is loopt de teller in het hoofdscherm zichtbaar terug. Wanneer de 0 bereikt wordt zal het window verdwijnen. Indien er voor die tijd een commando wordt gegeven of een toets ingedrukt wordt, wordt het commando uitgevoerd en de teller gereset.

K (Key): instellen Hotkey

Tenzij de automode 'Off' staat kan de Tracer – als er een Basic programma loopt – ook opgeroepen worden door een Hotkey. Standaard staat die Hotkey ingesteld op Ctrl-Shift. Omdat die instelling problemen zou kunnen geven met andere TSR's of zelfs het lopende Basic programma (bijvoorbeeld wanneer de Shift toets uitgelezen wordt) is het mogelijk een andere toetscombinatie als Hotkey te kiezen.

C (Clr): wissen alle instellingen

In feite is deze optie gelijk aan het commando CMD CLEAR TRACER. Na een waarschuwing worden alle instellingen gewist en in de default stand gezet. Het kan zijn dat de Tracer daardoor onzichtbaar wordt, namelijk als de voor- en de achtergrondkleur gelijk zijn. Dat kan dan met een druk op Select of R (clR) opgelost worden.

D (Dsp): inschakelen originele display

Door een druk op de D of Return-toets is het mogelijk het window tijdelijk te laten verdwijnen en zo het originele scherm te bekeken. Vooral als het om een grafisch scherm gaat is dat nuttig. Een willekeurige toets haalt het window weer terug.

P (stP): door het programma 'stappen'

Een druk op P verlaat de Tracer net zoals de spatiebalk of bijvoorbeeld Escape dat doen. Er is echter één bijzonderheid: de volgende keer dat de Tracer actief wordt, wordt niet automatisch het window getoond. De Tracer wordt wel actief en wacht op een commando, maar het originele scherm blijft zichtbaar. Op die manier kan er stap voor stap door een programma gelopen worden zonder dat het window telkens verschijnt. Wanneer er tijdens het wachten op een toets op de P gedrukt wordt, begint de uitvoering van de volgende Basic instructie en zal het window de volgende keer weer niet verschijnen. Elke andere toets haalt het window alsnog naar voren.

?: versie informatie

Een druk op het vraagteken geeft een scherm met informatie over de Tracer. Hier is onder andere het versienummer te vinden.

Hulpschermen

Naast het hoofdscherm zijn er nog een aantal informatieschermen in de Tracer te vinden. Ze kunnen allemaal vanuit het hoofdscherm opgeroepen worden. Overigens is het – hoewel het op het scherm niet aangegeven wordt – ook mogelijk uit de verschillende informatieschermen rechtstreeks naar een ander scherm over te schakelen door de corresponderende letter in te toetsen. Zodra er een toets ingedrukt wordt die geen betekenis heeft schakeld de Tracer terug naar het hoofdscherm.

M (Mem): Used Memory Overview

Dit scherm geeft een overzicht van het gebruikte geheugen. De begin- en eindadressen alsmede de lengte van elk gebied wordt getoond.

LET OP 1: De lengte van het gebied voor de strings is de totale lengte, er wordt geen onderscheid

gemaakt tussen daadwerkelijk gebruikt geheugen en nog vrije delen. De hoeveelheid vrij stringgeheugen kan opgevraagd worden met de functie FRE(""), die natuurlijk ook werkt uit de expressie evaluator van de Tracer.

LET OP 2: Het vrije gebied bevat ook de stack, die bijvoorbeeld bij veel geneste FOR-NEXT lussen en GOSUB's (zie ook het I (Inf) scherm) een flinke omvang kan bereiken. De hoeveelheid daadwerkelijk vrij geheugen kan opgevraagd worden met de functie FRE(0), die natuurlijk ook werkt uit de expressie evaluator van de Tracer.

| USED MEMORY OVERVIEW | | | |
|----------------------|--------------------|-----|--------------|
| Area | start | end | length |
| Program | 8001h-802Fh | | 47 |
| Variables | 8030h-8071h | | 66 |
| Array's | 8072h | | 0 |
| Free/Stack | 8072h-D44Ch | | 21467 |
| Strings | D44Dh-D514h | | 200 |
| T O T A L | 8001h-D514h | | 21780 |

Y (tYp): Overview DEF STR/INT/SNG/DBL

Dit scherm geeft een overzicht van de standaard types van de variabelen. Er zijn in MSX Basic vier types variabelen AB, die ook naast elkaar gebruikt kunnen worden: namelijk een integer (%), een single (!), een double (#) en een string

(\$). Standaard zal de naam AB altijd voor een double aangezien worden, maar met een DEF instructie kan dat veranderd worden. In dit scherm is altijd te zien wat de huidige instelling is. Het voorbeeld hiernaast laat zien hoe alle variabelen beginnend met de letters M, S of X als integers gedefinieerd zijn.

| Overview DEF STR/INT/SNG/DBL | |
|------------------------------|----------------------------|
| Strings (\$) | |
| Integers (%) | |
| Single precision (!) | M-----S-----X-- |
| Double precision (#) | ----- |
| | ABCDEFGHIJKL-NOPQR-TUVW-YZ |

V (eVn): Event trap Activity

MSX Basic ondersteund zogenaamde events. Het is mogelijk door een enkele instructie aan het begin van een programma te bewerkstelligen dat een bepaalde subroutine automatisch wordt uitgevoerd bij – bijvoorbeeld – het indrukken van een functietoets of een vuurknop van een joystick. Ook is het mogelijk zo'n subroutine bijvoorbeeld elke 10 seconden uit te voeren.

| EVENT | TRAP | ACTIVITY | | |
|-------|------|----------|------|----------|
| key1 | 100 | stop | key2 | 200 on |
| key3 | ---- | off | key4 | ---- off |
| key5 | ---- | off | key6 | ---- off |
| key7 | ---- | off | key8 | 300 off |
| key9 | 400 | off | ky10 | ---- off |
| stop | ---- | off | sprr | ---- off |
| trg0 | ---- | off | trg1 | ---- off |
| trg2 | ---- | off | trg3 | ---- off |
| trg4 | ---- | off | intr | 100 off |

In dit scherm is van elk event te zien naar welk regelnummer er gesprongen zal worden en wat de toestand van dat event is. Voor meer informatie over events kun u het desbetreffende hoofdstuk in de Basic handleiding raadplegen.

X (eXp): Expression evaluator

De expressie evaluator is één van de meest complexe onderdelen van de Tracer. De belangrijkste oorzaak daarvan is dat er gebruik wordt gemaakt van een aantal routines in het Basic ROM die geschreven zijn voor MSX Basic en

```

EXPRESSION EVALUATOR
ex1 12
12
ex2 MID$("TEST",2,2)
"ES"
ex3 SIN(TAN(2))
-.81720966124778
    
```

niet voor TSR's als de Tracer. Hoewel de Tracer – en zeker dit deel – goed getest is op allerlei verschillende MSX computers zou het kunnen dat het op enkele exotische exemplaren toch niet helemaal vlekkeloos werkt.

In de expressie evaluator kunnen drie Basic expressies ingevoerd worden, die direct uitgerekend worden. Elke expressie die in Basic legaal is kan hier na een druk op 1, 2 of 3 worden ingevoerd.

De meest eenvoudige toepassing is natuurlijk het opvragen van de waarde van een variabele, door als expressie simpelweg de naam van die variabele in te voeren. Maar elke expressie mogelijk. In feite kan elke berekening die in Basic uitgevoerd kan worden, ook vanuit de Tracer gebeuren. De enige beperking is de maximale lengte. In Basic kan een programmaregel en daarmee een expressie maximaal zo'n 250 bytes lang zijn, terwijl er in de Tracer 'maar' 2 regels van 29 tekens beschikbaar zijn.

Het bijzondere van de eerste expressie is dat het resultaat ook in het hoofdscherm zichtbaar is. Deze expressie is dus bij uitstek geschikt om de waarde

van belangrijke variabelen te volgen. Twee variabelen tegelijkertijd volgen is mogelijk door gebruik te maken van de volgende truuk:

STR\$(X)+STR\$(Y)

Door deze instructie op de plaats van expressie 1 in te vullen verschijnen zowel de waarde van X als de waarde van Y als resultaat. Ze zijn dan ook beide te zien in het hoofdscherm.

Expressie twee heeft als bijzonderheid dat er op getest kan worden. Door de automode op 'trUe' te zetten is het mogelijk de Tracer te laten verschijnen als expressie 2 als resultaat 'waar' (ofwel de integer -1) oplevert. In dat geval zal het vaak voorkomen dat er een zogenaamde 'Booleaanse' expressie ingevuld wordt. Dus bijvoorbeeld:

S<1291

om de Tracer te laten verschijnen zodra de variabele S een waarde hoger dan 1291 bereikt. Normaal gesproken zal deze expressie de waarde 0 (ofwel onwaar) opleveren. Zodra aan de voorwaarde voldaan wordt wordt het resultaat -1.

De ingevoerde expressies worden slechts geevalueerd als dat nodig is. Als het resultaat van een expressie bekend is, onthoudt de Tracer dat. Wanneer er bijvoorbeeld tijdelijk naar het hoofdscherm gestapt wordt, wordt er de expressie niet indien nodig opnieuw geevalueerd. Als het window opnieuw verschijnt – en dus de variabelen veranderd kunnen zijn – worden alle expressies opnieuw geevalueerd.

Bijzondere gevallen zijn expressie 1 en 2. De eerste wordt altijd geevalueerd als de Tracer verschijnt, omdat het resultaat ook in het hoofdscherm zichtbaar is. De tweede wordt als de automode op 'trUe' staat bij elke Basic instructie geevalueerd om te kunnen besluiten of de Tracer moet verschijnen of niet.

LET OP 1: Wanneer er in een expressie een array-variabele genoemd wordt die in Basic nog niet gedimensioneerd is wordt die array aangemaakt, met de standaarddimensie van 10. Dat wil zeggen dat het noemen van het onbekende array-element Q(2,3,1) tot gevolg heeft dat de array Q(10,10,10) aangemaakt wordt wat als Q een dubbele precisie variabele iets meer dan $8*11*11*11 = 10468$ bytes kost. Dat wil zeggen dat vrijwel zeker een 'Out of Memory' melding zal volgen.

De andere kant van dit probleem is dat er mogelijk een Redimensioned Array melding volgt, als u een programma start terwijl één van de in dat programma gebruikte array's in de expression evaluator genoemd is. Zodra die

expressie geevalueerd wordt, wordt de array aangemaakt, waardoor hij al bestaat als de DIM instructie uitgevoerd wordt.

LET OP 2: Sommige Basic functies hebben ongewenste neveneffecten. RND(1) haalt een getal uit de RND generator, wat meestal geen probleem is. Maar een instructie als INPUT\$(1,1) leest een byte uit bestand #1, een byte die het lopende Basic programma vervolgens niet te zien krijgt!

I (Inf): Other information

Dit scherm bevat een mengseltje van allerlei andere informatie. Links boven is als eerste het regelnummer te zien waar bij een ERROR naar toe gesprongen zal worden. Dit met het ON ERROR

| OTHER INFORMATION | | | | |
|-------------------|-----------|-----------|-------|-------|
| ON ERROR: ----- | ML0 | ----- | ----- | |
| ML1 ----- | ML2 | ----- | ML3 | ----- |
| ML4 1234h | ML5 | ----- | ML6 | ----- |
| ML7 ----- | ML8 | ----- | ML9 | ----- |
| | MaxFiles: | | 1 | |
| | CursorX: | | 1 | |
| | CursorY: | | 2 | |
| | DATAline: | | 0 | |
| NEXT F | 10 | DOT line: | 10 | |

commando ingestelde regelnummer behoort niet tot de Event Traps omdat er niet bij elke instructie gecontroleerd wordt of er misschien een ERROR opgetreden is. Er wordt pas bij een fout gekeken of er misschien een ON ERROR-regel ingesteld was.

Verder zijn in dit scherm de DEFUSR adressen te zien, achter de afkortingen ML0 tot ML9. De adressen worden hexadecimaal weergegeven.

Het belangrijkste deel van dit scherm bevindt zich linksonder. Normaal gesproken is deze hoek leeg, maar als er iets op de Basic stack staat is dat hier te zien. De Basic instructies GOSUB en FOR zetten een blok informatie op de stack waarin onder andere het regelnummer wordt opgeslagen waarnaar teruggekeert wordt. Bij een RETURN of een NEXT wordt de stack afgezocht naar het eerste bijbehorende gegevensblok. Op die manier 'weten' RETURN en NEXT naar welke regel teruggesprongen moet worden.

Rechts van de verticale streep staan een aantal systeemvariabelen die voor de Basic programmeur van belang kunnen zijn. We zullen ze kort langslopen:

| | |
|----------|--|
| MaxFiles | Het maximale aantal files dat tegelijkertijd geopend kan worden. |
| CursorX | De X-positie van de cursor. |
| CursorY | De Y-positie van de cursor. |
| DATAline | Het regelnummer waar gezocht zal worden naar DATA bij de eerstvolgende READ instructie. |
| DOT line | Het regelnummer dat als laatste ingevoerd of gewijzigd is, of waar een fout optrad. Deze regel kan 'gelist' worden met het Basic commando LIST. (dus met een punt achter list) |



Problemen & oplossingen

Er is een leeg scherm zichtbaar, geen Tracer.

Het kan zijn dat de voor- en de achtergrondkleur aan elkaar gelijk zijn, of dat de karakterset gewist is. Druk een aantal keer op Select, mogelijk dat de Tracer dan wel weer verschijnt.

De machine hangt als ik expressies evalueer.

Tijdens het evalueren van expressies wordt het Basic ROM aangeroepen. Het is mogelijk dat de noodzakelijke routines in uw ROM op andere adressen zitten waardoor het fout loopt. De volgende mogelijkheid is echter waarschijnlijker.

Het kan zijn dat er andere TSR's actief zijn die de expressie evaluator van het Basic ROM uitbreiden. Hoewel dat over het algemeen prima zal werken zijn er een tweetal manieren waarop het fout kan gaan. Ten eerste is het mogelijk dat de 'Recursiediepte' van MemMan te klein is. De recursiediepte kan ingesteld worden met het installatieprogramma CFGMMAN.COM. Ten tweede kan het fout gaan als er (te vaak) fouten gemaakt worden in expressies. Ook in dat geval zal het vergroten van de recursiediepte helpen, maar dan slechts tijdelijk. In MemMan 2.3 is dit probleem nog niet afdoende op te lossen, in de volgende versie zal dat echter wél mogelijk zijn. De Tracer is daar al op voorbereid: wanneer MemMan 2.4 of hoger (die dus bij het uitbrengen van de Tracer nog niet uit is) aanwezig is zal dit probleem opgelost zijn.

MSX 2+ grafische schermen worden niet goed hersteld.

Als u wél een MSX2+ videochip heeft, maar geen MSX2+ ROM worden de nieuwe VDP registers niet in het RAM bewaard. Daardoor kan de Tracer ze niet herstellen.

Er verschijnt een Redimensioned Array melding als ik de Tracer gebruik, die ik normaal gesproken niet krijg.

Waarschijnlijk staat die array in de expressie evaluator genoemd. Zie 'LET OP 1' bij de paragraaf over de expressie evaluator.

Tracer en MemMan

Op de valreep nog even iets over Tracer en de Memory Manager. De Tracer is een TSR (Terminate and Stay Resident) programma dat alleen werkt onder MemMan 2.2 of hoger. Dat wil zeggen dat de Tracer alleen maar kan werken als MemMan in het geheugen aanwezig is.

De Memory Manager neemt de Tracer een hoop werk uit handen. Om te beginnen zoekt MemMan een plaats voor de Tracer in het geheugen. vrijwel altijd zal dat een nieuw segment van 16 kB zijn. Daarnaast zorgt MemMan dat de Tracer op de juiste momenten op de juiste manier wordt aangeropen. Ook daar hoeft de Tracer zich zelf geen zorgen meer over te maken.

Zonder MemMan had de Tracer ook geprogrammeerd kunnen worden. Het programma was dan echter nog groter geweest, nog ingewikkelder en had in ieder geval niet samengewerkt met de Printerbuffer PB.TSR, de MemMan RAMdisk van de TSR verzameldisk, de DOS2 RAMdisk en vele andere TSR's zoals het in MCM gepubliceerde ICP7.TSR en MOUSOR.TSR. Kortom: de mogelijkheden en daarmee het nut van het programma waren beperkt geweest.

Het laden van Tracer

Op de schrijf staat een MemMan die speciaal geconfigureerd is voor de Tracer. Wanneer MEMMAN.BIN vanuit Basic of MEMMAN.COM vanuit MSXDOS geladen wordt, wordt ook de Tracer geladen. Dit kan met behulp van het MemMan configuratieprogramma CFGMMAN.COM vanuit DOS gewijzigd worden. Voor meer informatie verwijzen we naar de MemMan handleiding, die in het bestand MM23INTR.TXT op de disk te vinden is.

Wanneer MemMan al geladen is kunt U Tracer natuurlijk gewoon inladen met het daarvoor bestemde commando. Vanuit Basic is dat:

```
CMD TL ("TRACER",T)
```

De toevoeging 'T' mag eventueel weggelaten worden, maar dan zal de welkomstboodschap niet verschijnen.

Vanuit DOS kan de Tracer ook geladen worden, en wel met behulp van het programma TL.COM. Het recept is eenvoudig, gewoon een kwestie van achter de prompt intikken:

```
TL TRACER
```



Index

| | | | |
|----------------------|--------|-----------------|--------|
| automode | 12 | karacterset | 14 |
| Breakpoint | 13 | Key | 15 |
| breakpoints | 14 | kleuren | 15 |
| Brk | 13 | laden | 22 |
| CFGMMAN | 21 | Lin | 13 |
| CFGMMAN.COM | 22 | line | 13 |
| cIR | 15, 16 | maxfiles | 20 |
| CMD | 8 | Mem | 17 |
| CMD CLEAR TRACER | 9 | MemMan | 22 |
| CMD TK | 8 | memory | 17 |
| CMD TL | 22 | Off | 12 |
| CMD TRACER | 8 | oplossingen | 21 |
| CMD TROFF | 8 | palletregisters | 15 |
| CMD TRON | 8 | problemen | 21 |
| cNt | 15 | return | 16, 20 |
| commando's | 14 | sEc | 15 |
| counter | 15 | seT | 14 |
| Ctrl-Stop | 10 | SET PAGE | 15 |
| DATA line | 20 | Shift-Home | 11 |
| DOT line | 20 | spatiebalk | 10 |
| Dsp | 16 | stappen | 16 |
| editor | 11 | statement | 12 |
| error | 20 | Stm | 12 |
| event traps | 18 | stP | 16 |
| eVn | 18 | teller | 15 |
| eXp | 18 | TL | 22 |
| expression evaluator | 13, 18 | TRON | 5 |
| Full | 12 | trUe | 13 |
| full window | 12 | tYp | 17 |
| GOSUB | 20 | verdwijnen | 15 |
| Home | 11 | versie | 16 |
| hoofdscherm | 14 | wissen | 16 |
| Hot | 12 | | |
| hotkey | 12, 15 | | |
| Inf | 20 | | |
| INPUT | 12 | | |