

Copyright (C) 1983 E.S.G.G.

B E X T (8)

De Basic EXTension bestaat onder meer uit een reeks routines die aan de input- en outputdrivers worden toegevoegd en aldus een aantal lastige fouten in Standard Basic herstellen.

Als BEXT actief is kunnen alle CTRL en CURSOR besturingstoetsen worden gebruikt zonder syntax errors te veroorzaken. De reellengte wordt begrensd, waardoor het systeem niet meer vastloopt door een te lange invoerregel. De RUN/STOP of ESC toets werkt echt, dwz kort indrukken van deze toetsen stopt de listing of programmarun. Het indrukken van elke willekeurige andere toets zet de zaak weer in beweging. Voorts is een groot aantal hulproutines opgenomen.

Bext moet vanuit de monitor worden ingeschakeld door het commando: GO DFE0.

Tenzij anders vermeld geldt deze gebruiksaanwijzing voor alle BEXT versies, t/m BEXT 8

Deze handleiding is een compilatie van al hetgeen in de eerste jaargang van het ESGG periodiek over BEXT is geschreven. Alle inzenders en vragenstellers die zodoende indirect aan deze handleiding hebben meegewerkt worden bij deze bedankt voor hun activiteiten.

Samenstelling: Wim de Kreuk en Floor Vogelaar.

BEXT voorziet standaard in de volgende USR routines:

- 0: Basicode schrijf. Commando: I=USR(0): LIST
- 1: Basicode lees. De input verschijnt op regel 1 t/m 5.

Voor het gebruik van de Basicode routines is wel een Basicode interface nodig.

- 2: Kleine letters worden omgezet naar hoofdletters
- 3: Tape schrijfroutine. (TP\$ naar buffer)
- 4: Tape leesroutine (vult buffer als die leeg is; Kent de eerste cq de eerstvolgende bufferwaarde toe aan TP\$)
- 5: Tape open routine. (Geeft een naam (uit TP\$) aan de file)
- 6: Tape sluit routine. (schrijft de buffer weg naar cassette)

De routines 2 t/m 5 hebben gemeen dat ze hun activiteit botvieren op de momentele waarde van de eerstbenoemde (string) variabele. In veel programma's wordt deze TP\$ genoemd, zo ook in deze handleiding. Bij gebruik van deze routines moet u er dus voor zorgen dat uw programma begint met TP\$="" in de eerste programma regel.

- 7: Inkey. Na gebruik van deze functie geeft PEEK(-4041) de waarde van de ingedrukte toets. Deze is 0 als er geen toets werd ingedrukt.
- 8: Scroll. De bovenste 0 tot 15 regels blijven op het beeldscherm. Het aantal regels kan in adres F026 worden gezet. Waarde bij het aanzetten = 8.
- 9: Invers video. Deze routine zet automatisch de scroll-routine aan. Als dat niet gewenst is moet de waarde 0 in adres F026 worden gezet.
- 10: Trace on. De displaysnelheid wordt vertraagd en tijdens het indrukken van de RUN/STOP toets worden de uitgevoerde programmaregels getoond.
- 11: Trace off. Zie onder. Ook BEXT off.

USR(10) moet beslist worden gevolgd door USR(11). Als u dit vergeet is het mogelijk dat de machine zo crashed dat u uw programma verliest. Voor elk INPUT statement moet de TRACE routine worden uitgeschakeld. bijv:

```
100 X=USR(11): INPUT A$: X=USR(10)
```

ofwel: eerst TRACE uitzetten, dan input en daarna TRACE weer aanzetten.

Hoewel niet onmogelijk, moet het gebruik van TRACE in de direct mode worden afgeraden.

- 12: Printer aan. Alle output gaat via een gemodificeerde centronicsdriver naar de printer. Graphics worden als spatie afgedrukt.
- 13: Video driver aan. Sluit USR(8), (9) en (12) af. Alle output gaat rechtstreeks naar de Video driver.

Als de SCROLL routine (USR(8)) actief is kunt u bovendien de volgende functies gebruiken. Deze kunnen vanuit BASIC (PRINT CHR\$(X)) en vanaf het keyboard (CTRL 'X') worden geactiveerd.

CTRL G CHR\$(7) Geeft een toon aan bit 0 van de Par.poort van de Sorcerer. De frekwentie (F036 Hex) en tijdsduur (F038-F039) zijn instelbaar. De default frekwentie is 2400 Hz, de waarde die de Basicode interface nodig heeft. Let er op dat deze waarde intact is voor u USR(0) gebruikt.

CTRL I CHR\$(9) Horizontal TAB. De TAB toets plaatst de cursor in stappen van 8 naar rechts.

CTRL L CHR\$(12) Clear voor het onderste schermdeel

CTRL O CHR\$(17) Home voor het onderste schermdeel

CTRL R CHR\$(18) Clear voor het bovenste schermdeel

CTRL Z CHR\$(26) Cursor down. Negeert de scrollroutine waardoor tekst naar het bovenste schermdeel (protected field) kan worden verplaatst.

De volgende CTRL opdrachten kunnen vanaf het toetsenbord worden gegeven:

CTRL B Schrijft de laatst ingelezen cassettefile weg naar recorder 2. Als in adres F020 (-4064) de waarde '31'hex wordt geplaatst wordt naar recorder 1 geschreven.

CTRL E Schakelt de editor en de linker uit.

CTRL F Print de header van de laatst ingelezen cassettefile.

CTRL P =USR(12); printer aan.

CTRL T vertaalt tokens (enkeltoetsinstructies) naar normale tekst. Bijv: Graphic 5 wordt 'BYE'

CTRL V =USR(13); printer uit.

CTRL X BEXT off. = USR(11)

CTRL Y Copy; copieert files van recorder 1 naar recorder 2. Motorbesturing noodzakelijk.

De opgenomen EDITOR (daar begon alles mee) bevat de volgende mogelijkheden.

TAB eerste maal Editor aan (cursor gaat knippen)

TAB Delete character

SHIFT TAB Insert character

Keyboard U kunt wijzigen door over de bestaande tekst heen te typen.

RETURN De gewijzigde regel wordt in het BASIC programme geheugen opgenomen.

@ Search; Tekst die u na '@' in typt wordt in de BASIC programfile opgezocht. De regels waarin deze tekst voorkomt worden afgedrukt. De BASIC statements worden als graphics afgedrukt, tenzij TOKEN (CTRL T) actief is. In dat geval worden echter ook de 'echte' graphics vertaald.

CTRL N Herstelt programma's die verloren zijn gegaan na NEW of CLOAD.

CTRL L (was in BEXT 3 CTRL I) LINK; Drukt op het beeldscherm een adres af. Het te koppelen programmadeel moet nu, als u niet de Van Montfort Monitor 1.3(B) gebruikt, als eerste op de cassette staan. Laden met CLOAD NAAM 1 (of 2) ADRES. (het adres dat zojuist op het scherm werd afgedrukt). Na het laden opnieuw met TAB de editor in en dan:

CTRL I hetgeen de twee programmadelen koppelt. De programmadelen mogen geen gelijke regelnummers hebben aangezien die elkaar overschrijven.

De Taperoutines, USR(3) t/m USR(6), zijn bedoeld om de problemen met CLOADX en CSAVEX te omzeilen en om alfanumerieke data (strings) van en naar cassette te kunnen lezen en schrijven.

Deze routines maken gebruik van een hulpbuffer, die begint op adres FC00 hex en max. 1280 bytes groot mag zijn. (De buffer loopt dan van FC00 over FFFF naar 00FF). De oplottende lezer heeft gezien dat dit de zelfdefinieerbare graficsruimte en page 0 betreft. Als in uw toepassing een bufferlengte van 512 bytes voldoende is, en u gebruik maakt van de Sorcerer Standard Graphics, kunt u, door de opdracht PRINT CHR\$(12) uit te voeren als u klaar bent met uw tapeacties, deze grafics weer herstellen. De overige grafics worden niet overschreven en blijven dus steeds beschikbaar. De normaal voor BASIC benodigde of beschikbare geheugenruimte blijft, zelfs bij gebruik van de maximum buffergrootte (1280 bytes), onaangetaast. Door de buffergrootte in het BASIC programma te kiezen en op volraken daarvan te testen hebben we de mogelijkheid om bij elkaar behorende bestandselementen, zoals naam, adres en woonplaats in een adresbestand, samen weg te schrijven naar tape, waardoor inlezen en bewerken van bestandsgedeelten mogelijk wordt. De routines staan niet toe dat u met "lege" strings werkt. In uw programma moet daarop worden getest. Bijv: IF TP\$="" THEN TP\$=" " voor het wegschrijven en IF TP\$=" " THEN TPC="" na het inlezen.

LET OP !! De variabele die u voor uw taperoutines gebruikt, moet beslist als eerste in uw programma worden benoemd.

Oudere versies van BEXT konden worden gestart door een zogenaamde 'warme start'. Dit was te realiseren door de beide RESET toetsen kortstondig in te drukken terwijl u de ESC toets blijft indrukken tot u het woord 'READY' op het beeldscherm ziet verschijnen of door vanuit de monitor middels het commando PP terug te gaan naar BASIC.

Hoewel STANDARD BASIC geheel in de oorspronkelijke staat in uw pack is blijven zitten, BEXT is slechts een aanvulling, kan de bovenstaande opstartmethode toch roet in het eten gooien. Wat gebeurt er namelijk?

- 1: de waarden op de USR vector (260 en 261) worden op het beginadres van de BEXT USR selectieroutine gezet en
- 2: De inputvector wordt op DF37 hex gezet (als bij SE I=DF37) waardoor alle BASIC input de BEXT correctieroutine gaat doorlopen.

Ad 1: Sommige programma's (waaronder KISS2) starten met een machinetaalroutine die, evenals BEXT, een eigen USR vector plaatsent. Vervolgens gaan deze programma's via een warme start naar BASIC waardoor BEXT wordt aangezet en de USR vector wederom wordt gewijzigd.

Ad 2: Andere programma's, waaronder Toolkit en System3, gebruiken een eigen inputroutine, aangestuurd door een van Basic afwijkende inputvector. Deze inputvector wordt door het programma geset, waarna de besturing, via een warme start aan BASIC wordt overgedragen. Die warme start zet BEXT aan en dus wordt de zojuist gesette inputvector weer veranderd.

Verderop worden voor een aantal van deze programma's oplossingen gegeven maar wij zijn tot de conclusie gekomen dat het beter is om van het opstarten van BEXT dmv een warme start af te zien. De opvolgende handelingen gevormd door het intoetsen van BYE (CR) en GO DFE0 (CR) zijn nu ook weer niet zo omvangrijk dat van een grote achteruitgang in het bedieningscomfort kan worden gesproken. Sinds enige tijd wordt BEXT 8 dan ook geleverd zonder de mogelijkheid van het opstarten dmv een warme start.

U kunt nu dus volledig zelf bepalen of u van BEXT gebruik wilt maken of niet, afhankelijk van het programma dat u gebruikt.

Als we het bovenstaande iets anders benaderen kan ook het volgende worden gesteld:

- 1: Als u tevoren de opdrachten POKE 260,88: POKE 261,223 laat uitvoeren, heeft u de USR functies 0 t/m 13, onafhankelijk van het al dan niet aanstaan van BEXT, tot uw beschikking. U kunt deze combineren met eigen USR programma's, genummerd van 14 tot 255. Het adres van deze programma's moet dan op de adressen 101 hex en 102 hex worden gezet. (POKE 257, lower byte: POKE 258, higher byte). Bij de aanroep van deze extended USR programma's bevat het E register reeds de USR variabele.
- 2: Onafhankelijk van het al dan niet aanstaan van BEXT kunt u dmv het Monitorcommando SE I=DF37 gebruik maken van de Basic correctieroutines.

Als laatste programma op ESGG verzamelband 7 staat DEMT1 een DEMO, van de hand van A.W.v.d.Ven, Roer 85 te Groningen waarin de werking van de taperoutines, USR(3) t/m USR(6), duidelijk wordt gemaakt. Warm aanbevolen.

Memorymap (geheugenindeling) van BEXT:

TAB editor met Linker, Search en Retrieval	DAD6-DC0E
TRACE routine	DC0F-DC1B
VIDEO routines met Scroll, Tabulator en Geluid	DC1C-DCB3
BASICODE leesroutine	DCBC-DD73
BASICODE schrijf routine	DD74-DE12
COPY van recorder 1 naar recorder 2	DE13-DE5F
Deel van het routinekeuzeprogramma	DE60-DE7C
Onderkast-->Kapitaal	DE7D-DE91
Tape schrijfroutine	DE92-DEAC
Tape open routine	DEAD-DEC1
Tape leesroutine	DEC2-DF04
INKEY routine	DF05-DF0F
INVERS VIDEO routine	DF10-DF24
Deel van het routinekeuze programma	DF25-DF33
Routinekeuze hoofdprogramma	DF34-DFDF
Opstart routine	DFE0-DFF9
Warme start vector	DFFA-DFFC
Koude start vector	DFFD-DFFF

Programma variabelen:

functie:	adres:	default:
Recorder pointer	F020	38
Scroll pointer	F026	58
Checksum (basicode)	F030	
Karakterteller (basicode)	F031	00
Adrespointer (intern)	F032-3	
Bufferpointer (intern)	F034-5	
Frekwentie (geluid+basicode)	F036	1C
Inkey character	F037	
Tijdsduur (geluid+basicode)	F038-9	03 01

LET WEL Door de vergaande comprimering van BEXT wordt door de afzonderlijke routines dikwijls gebruik gemaakt van subroutines uit andere delen. Let daar vooral op als u delen van BEXT buiten de EPROM wilt gebruiken.

Nu wat tips:

Proefondervindelijk is vastgesteld dat de inhoud van adres DCBC (nu 24hex) 20 moet zijn. We hebben op onze oproep in ESGG 5 een twintigtal reacties gehad waaruit blijkt dat deze waarde de minste problemen gaf bij het BASICODE lees programma. Laat dit door een programmereigenaar even veranderen. Kent u niemand in uw omgeving die dat voor u kan doen, neem dan even contact op met ons secretariaat voor een oplossing.

BEXT 3 werkt niet goed samen met de monitor 1.3. In BEXT 8 is een routine van de gebr. Van MONTFORT opgenomen die dat probleem oplost. BEXT 3 bezitters kunnen tegen inlevering van hun oude EPROM en een geldbedrag van f 10,-- BEXT 8 krijgen.

Ook mensen die van hun "warme start" af willen kunnen hun EPROM laten omprogrammeren. Voorwaarden als in voorgaande alinea.

Raadpleeg voor informatie hierover het ESGG periodiek of vraag inlichtingen bij onze stichtings secretaris, Charles Nettelers, Pr. Hendrikstraat 3d, 3071 LG Rotterdam, tel. 010-330493.

Wat voorbeelden:

```
INKEY 10 I=USR(7): I=PEEK(-4041): IF I=0 THEN 10
        20 PRINT CHR$(I)
```

Eerst de INKEY opdracht USR(7): De ASCII waarde van de ingedrukte toets staat op adres F037 (PEEK -4041). Als er geen toets was ingedrukt bevat dat adres de waarde 0. Regel 10 wordt net zolang uitgevoerd tot er wel een toets werd ingedrukt. Regel 20 drukt daarna het daarmee overeenkomende character af. Natuurlijk had de toetswaarde ook aan een lege string kunnen worden toegekend of aan een bestaande string worden toegevoegd.

```
20 A$=CHR$(I) of
20 A$=A$+CHR$(I).
```

Kleine letters --> hoofdletters:

```
10 INPUT TP$
20 I=USR(2)
30 PRINT TP$
```

TP\$ is de eerst benoemde variabele. !!!

In regel 10 wordt om INPUT gevraagd. Regel 20 test if in die INPUT kleine letters voorkomen en verandert die in hoofdletters. Regel 30 laat het resultaat zien.

Als toepassing wordt getipt de bestandsnaaminvoer voorafgaande aan de TAPE Open routine.

Tenslotte voor diegenen die geen afstand willen doen van hun Warme Start mogelijkheid een aantal wijzigingstips:

TOOLKIT: Laadt Toolkit met LO (niet LOG) en wijzig als volgt:

```
EN #89E (CR)          6B C0 / (CR)
EN #A6E (CR)          6B C0 / (CR)
EN #A93 (CR)          6B C0 / (CR)
EN #C94 (CR)          6B C0 / (CR)

SE F=54 (CR)         SE X=#880 (CR)
SA TLKIT #600 #DFF
```

(CR)=RETURN toets.

#=0 voor het relocatable programma

#=3 voor de 16k versie

#=7 voor de 32k versie

#=B voor de 48k versie

SYSTEM 3 Laadt SYS3 met LO (niet LOG) en wijzig als volgt:

```
EN 1C98 (CR)          6B C0 (CR)
EN 1D23 (CR)          6B C0 (CR)
SE F=30 (CR)          SE X=0F00 (CR)
SA SYS3A 0F00 1F4F
```

(CR)=RETURN toets.

KISS2 Neem in het Basic programma de volgende regel op:

```
5 POKE 260,51: POKE 261,188
```

De waarde 188 geldt voor een 48k Sorcerer. Voor een 32k of 16k machine moet dit respectievelijk 124 en 60 zijn.

Gebruik voor vragen en opmerkingen over BEXT de rubriek INPUT in ons ESGG periodiek.

Nog geen abonnee ?

Maak f 18,00 over op giro 5368539 t.n.v. ESGG te Lopik.

En u bent het.

Het abonnementsjaar loopt van juni van het lopende jaar tot en met mei van het daaropvolgende jaar. Wilt u eerdere nummers ontvangen, maak dan per nummer f 3,50 extra over.

Copyright (C) 1983 E.S.G.G.