

Bijlage: verklarende woordenlijst

Bijlage: ASCII

Bijlage: bruikbare adressen

Bijlage: bruikbare routines

Bijlage: file system

System calls versie 1.3

Voorbeelden

System calls versie 1.4

Z E M O N

Assembler-monitor ROM module voor de P2000

ZEMON is geschreven door Loek Hilgersom en Frits Jalvingh.

ZEMON

ZEMON is een assembler-monitor voor de Philips P2000 computer. ZEMON is zowel voor het T-model als voor het M-model geschikt. De verschillen tussen de computers worden door het programma ondervangen, zodat ZEMON op beide computers optimaal bruikbaar is.

Met ZEMON kunnen machinetaalprogramma's worden geschreven, opgeslagen, verandert, getest, enz. Daarnaast kunnen m.b.v. ZEMON twee computers met elkaar worden verbonden met een RS232 kabel, zodat programma's, zowel tekst- als objectfiles, kunnen worden overgezonden. Zo kan een tweede computer als geheugenbank dienen om snel grote programma's te assembleren, maar er kunnen ook programma's worden overgezonden naar andere computers dan de P2000. Kortom, ZEMON biedt onbeperkte mogelijkheden voor enthousiaste computerfreaks!

Reset toets

Bij de meeste modules, zoals Basic of een tekstverwerker wordt altijd bij het indrukken van de reset-toets het geheugen leeg gemaakt. In ZEMON werkt de reset-toets echter niet destructief, na het indrukken blijven programma's in het geheugen aanwezig. Omdat bij het aanzetten van de computer ook een reset-signaal wordt geproduceerd, wordt ook dan het geheugen niet gewist. Om zeker te zijn van een goede werking van ZEMON is het aan te raden bij het aanzetten van de computer eerst de "Clear RAM" opdracht ('C'-toets) te geven. Na bevestiging met 'Y(es)' wordt de inhoud van het ram-geheugen gewist. Ook als aan een geheel nieuw programma wordt begonnen kan het nuttig zijn eerst het geheugen leeg te maken (Zo krijgt men geen verwarring met gegevens van het andere programma).

Getallen invoer

Als er in de monitor of in de editor een getal moet worden ingevoerd, verschijnt er bij de meeste functies automatisch een hekje ('#') op de invoerregel. Dit staat daar omdat meestal hexadecimale getallen worden ingevoerd. Men kan ook decimale getallen, labels of hele berekeningen invoeren, zelfs in de monitor. Daarvoor moet eerst het hekje worden weggehaald met de Backspace toets ('<XI') of de regelwis toets ('IXI').

ZEMON's monitor

Als de computer wordt aangezet met de ZEMON rom module in de eerste sleuf, wordt op het beeldscherm een lijst met mogelijkheden zichtbaar. Met de ZEMON monitor kunnen machinetaalprogramma's worden getest, uitgevoerd, of verbeterd. U geeft de monitor een opdracht door de toets voor de gewenste opdracht in te drukken.

Toetsfuncties in ZEMON's monitor

Niet alle mogelijkheden van de monitor zijn in het lijstje op het beeldscherm opgenomen. Hieronder volgt een volledige lijst met bij iedere functie een uitleg.

A - Assembler

Entree in de 'full screen editor' van ZEMON's assembler, voor verdere informatie zie beschrijving van editor en assembler.

B - Breakpoints

Plaats op adres "at" een breakpoint (trap). Een breakpoint is slechts 1 byte lang! Tijdens het uitvoeren van een machinetaalprogramma zal bij het passeren van dit adres het programma stoppen en terugkeren naar ZEMON's monitor. Met "Registers" kunnen daarna de Z80 registers, zoals die stonden op het moment dat de breakpoint werd uitgevoerd, worden bekeken.

C - Clear RAM

Na bevestiging wordt het ram-geheugen vanaf adres #5000 tot en met #FFFF gewist. Tevens worden een aantal systeemvariabelen geïnitieerd (als bij de "Init"-opdracht). Omdat ZEMON bij RESET het geheugen nooit wist, is het aan te raden om bij het aanzetten van de computer altijd eerst de "Clear RAM" opdracht te geven.

D - Disassemble

Disassembleert een machinetaalprogramma vanaf adres "from". Als een label van een geassembleerd programma aanwezig is en de labels 'aan' zijn ("Labels on/off") dan worden alle 16-bits getallen waarvoor een label bestaat vervangen door dat label.

Toetsen in de disassembler:

- 'Enter' - Disassembleert 16 regels.
- 'Spatie' - Disassembleert 1 regel.
- 'STOP' - Keert terug naar de monitor.

E - Edit

Geeft een hex- en asciidump van het geheugen op adres "at", zowel in het hex- als in het ascii-vlak kunnen bytes worden veranderd (Memory-edit).

Toetsen in memory-edit:

- Cursorpijlen - Verplaatsen de cursor over het hele scherm.
- 'Backspace' - Werking gelijk aan cursorpijl naar links.
- 'TAB' - Verplaatst de cursor van het hex- naar het ascii-vlak, of andersom.
- 'Enter' - Geeft een volgende regel en plaatst de cursor vooraan het vlak waarin de cursor stond.
- 'DEF' - Alleen in ascii-vlak: Inverteert bit 7 van het teken waarop de cursor staat.
- Shift + 'IXI' - Wist het scherm en plaats de regel waarin de cursor staat bovenaan het scherm.
- 'STOP' - Keert terug naar de monitor.

F - Fill

Vult het geheugen vanaf adres "from" tot adres "until" (exclusief adres "until") met het getal "with". Als dit getal groter is dan 256 (#100) dan wordt het aangegeven stuk geheugen met een 16-bits getal gevuld (in laag-hoog bytevolgorde).

G - Go

Het machinetaalprogramma op adres "destination" wordt uitgevoerd. De registers van de Z80 krijgen voor het uitvoeren de waarden zoals die bij de "Registers"-opdracht zijn ingesteld. Het I-register wordt altijd op #60 gezet, SP krijgt een vaste waarde en PC wordt geladen met "destination". Na het uitvoeren van het machinetaalprogramma kan teruggekeerd worden met 'Ret', of, als de stackpointer veranderd is, met 'Rst #10' (Breakpoint).

H - Help

Geeft een verkort overzicht van de opdrachten, zoals na reset.

I - Init

Initialiseert een aantal systeemvariabelen, waaronder die van de printer en de labellengte (6). Dit kan nodig zijn na dat een programma de boel gesloopt heeft.

K - Kill breakpoint

Verwijdert de breakpoint waarvan het nummer wordt ingetypt. Hierbij wordt de oorspronkelijke inhoud van het adres hersteld. Mocht de breakpoint, om wat voor reden dan ook, al verdwenen zijn, dan wordt een waarschuwing gegeven en wordt de inhoud van het adres niet veranderd. Met shift+'2' (numeriek toetsenbord) worden alle breakpoints verwijdert.

L - Labels on/off

Schakelt de labels tijdens het disassembleren aan of uit.

M - Move

Verplaatst na bevestiging het geheugenblok tussen adres "from" en adres "until" (exclusief adres "until") naar adres "destination".

P - PGM save

Schrijft een file naar cassette als stand-alone machinetaalprogramma, met type aanduiding PGMP. Zo'n programma kan geladen worden zonder module in de eerste sleuf door een cassette in de recorder te plaatsen. Het programma wordt na het laden vanzelf uitgevoerd.

In te typen gegevens:

- "from" - Begin adres van de file in het geheugen.
- "until" - Eind adres van de file in het geheugen.
- "PGM load" - Stand-alone laadadres.
- "PGM run" - Startadres bij stand-alone laden.
- "name" - Naam van het programma. De naam mag 16 tekens lang zijn.

R - Registers

Geeft de inhoud van de Z80 registers zoals deze waren tijdens de laatste breakpoint, reset of NMI. Na een breakpoint of een NMI zijn de waarden van alle registers geldig, na een reset hebben AF en PC foutieve waarden. De inhoud van de registers kan ook worden veranderd. Als de "Go"-opdracht wordt gegeven krijgen de Z80 registers de waarden zoals die in "Registers" zichtbaar zijn (zie ook uitleg Go-opdracht).

De inhoud van de registers kan worden veranderd, waarvoor de volgende toetsen gebruikt worden:

- cursorpijlen - Naar links en rechts
- 'TAB' - Verplaatst de cursor naar het volgende registerpaar.
- 'I' - Schakelt de interrupts aan of uit.
- 'X' - Verwisselt de waarden van de registers met die van de 'alternate' registers (Exx en Ex AF,AF').
- Shift +'IXI' - Wist het scherm en plaatst de registers bovenaan.
- 'STOP' of 'Enter' - Keert terug naar de monitor.

S - Single step

Voert een machinetaalprogramma vanaf adres "from" stap voor stap uit. Voor iedere stap worden de instructie en de Z80 registers getoond. De Z80 registers kunnen voor iedere stap gewijzigd worden, zoals bij "Registers". Als de instructie op het beeld verschijnt is deze nog niet uitgevoerd. Wat er met de instructie gebeuren moet wordt met de volgende toetsen bepaald:

'Enter' - De instructie wordt uitgevoerd en keert direkt daarna terug naar single step.

'S' - (Skip) De instructie wordt niet uitgevoerd, single step gaat verder bij de volgende instructie.

':' - De instructie wordt op volle snelheid uitgevoerd. Bij veel instructies maakt dit geen verschil met 'Enter'. Bij een Call- of Rst-instructie wordt de subroutine op volle snelheid uitgevoerd. Als de processor terugkeert van de subroutine gaat het programma weer verder in single step. Er zijn routines waarbij dit absoluut NIET mag, bijv. de voorbeeld routine MESPR1.

'STOP' - Keert terug naar de monitor.

Als tegelijk met een van de toetsen de shift toets wordt ingedrukt, dan wordt de automatische herhaling van de toets versneld (fast auto-repeat).

Tijdens single step in de ZEMON rom kan het programma vastlopen doordat de data van de single step routine door een andere routine kunnen worden veranderd.

T - Symbol table

Leest een labeltabel met naam "name" in het geheugen vanaf cassette, disk of RS232. De labeltabel-pointers (SOL en EOL, zie bijlage) worden op de goede waarden gezet. Als men na het assembleren van een programma niet alleen de object, maar ook de labeltabel wegschrijft, kan deze later weer worden ingelezen met deze opdracht. Zo kan men altijd disassembleren met labels. De tabel komt altijd op de plaats waar de tabel stond toen hij werd weggeschreven (de tabel mag niet worden verplaatst). Door het inlezen van een labeltabel kan de tekst in het geheugen dus worden gesloopt!

V - V24 baudrate

Stelt de baudrate van de V24 interface (= RS232 = printerpoort) in op de waarde "baud". Baudrates tussen 75 en 19200 baud kunnen worden ingesteld, wordt een waarde buiten deze grenswaarden ingetypt dan volgt de foutmelding "Number too big". De baudrate wordt naar boven afgerond op de dichtsbijzijnde mogelijke waarde.

Z - Search

Zoekt vanaf adres "from" de ingetypte bytes "codes" in het geheugen. De ingetypte codes zijn 8-bits waarden, meerdere waarden kunnen worden ingetypt, gescheiden door komma's. De codes kunnen getallen zijn, maar ook ascii waarden. Als bijvoorbeeld de string "Hoi" moet worden opgezocht, moet bij "codes" worden getypt: "H","o","i"

Na ieder adres waarop de ingegeven codes gevonden worden wordt op een toets gewacht. Met 'STOP' wordt teruggekeerd naar de monitor, met een andere toets wordt verder gezocht. Als het einde van het geheugen is bereikt (adres \$FFFF) wordt de foutmelding "Not found" gegeven.

4 - 40 chars

8 - 80 chars

Stelt de computer in op 40 of 80 tekens per regel. Ook de tabulatie posities in de editor worden hierop aangepast.

Enter - Carriage return + line feed

Zet de cursor een regel omlaag.

Functies van het numerieke toetsenbord in de monitor

DEF - Calculate

Geeft de uitkomst van een ingetypte berekening in hexadecimaal, decimaal, binair en in asciiconstanten. Te gebruiken voor kleine berekeningen, omrekening van bijvoorbeeld decimaal naar hex of om de waarde van een label te tonen.

Printertoets (Shift+'00') - Printer on/off

Na het indrukken van deze toets worden alle regels die in ZEMON's monitor op het beeld verschijnen afgedrukt op de printer. Geeft de printer na enkele seconden nog geen ready signaal dan wordt automatisch de printer weer uitgeschakeld. Als de computer op een ready-signaal van de printer wacht kan dit altijd worden onderbroken door nogmaals de printertoets in te drukken.

Ook de foutmeldingen en geliste labels tijdens assembleren worden na indrukken van deze toets op de printer afgedrukt.

Index

Tape: Toont de inhoud van de cassette in de recorder. Hierbij wordt de naam, de type aanduiding, de lengte en het aantal blokken van de files zichtbaar. Als een cassette-fout wordt geconstateerd wordt een waarschuwing gegeven en verder gespoeld naar het volgende blok op de cassette.

Disk: Vraagt een drivenummer, waarna de inhoud van de disk in de gekozen drive wordt getoond. De eerste naam die verschijnt is de naam van de schijf, met daarachter het aantal gebruikte sectoren op de schijf (op 1 schijf in een 40-tracks drive kunnen 200 sectoren). Alle volgende namen zijn de files op de disk met de bijbehorende file-lengtes.

Bliksem toets (Shift +'2' - Format

Formaatteert de disk in de drive waarvan het nummer is opgegeven en geeft deze disk de opgegeven naam. Als na "name" wordt ingetypt: 2:Hallofans, dan wordt de disk in drive 2 geformatteerd en krijgt de disk de naam Hallofans (medium name).

INL - Load

Laad een file met naam "name" in het geheugen op adres "at". Als geen adres of adres 0 wordt opgegeven wordt de file ingelezen op het adres waar die file weggeschreven is (het adres dat in de header staat). Na lezen van de cassette wordt de naam van de gelezen file in de computer bewaard.

save

Schrijft een file uit het geheugen weg vanaf adres "from" tot adres "until" met naam "name". De file krijgt de type aanduiding BINB. (Bij disk is de extentie .BIN)

Cassette toets (Shift +'7') - Delete

Tape: Wist de file met naam "name", en alles wat daarachter op de cassette stond. Als geen naam wordt opgegeven wordt de hele cassette gewist. ('End of file mark' vooraan de cassette)

Disk: Wist de file met naam "name". Achter de naam moet ook de extentie (Bijv. .BIN of .ASM) worden getypt. Alle ruimte die op de disk vrijkomt wordt altijd automatisch weer benut als een nieuwe file op de disk wordt geschreven.

Disk toets (Shift +'8') - Toggle tape/disk

Schakeld om van tape naar disk, of andersom. Als er geen diskcontroller in de P2000 zit kan er niet naar disk worden omgeschakeld.

Shift +'5' - Rewind

Bekrachtigt een eventueel aanwezige terugspoelautomaat.

Wis scherm toets (Shift +'IXI') - Clear screen

Wist het scherm en zet de cursor bovenaan.

ZEMON's full screen editor

In ZEMON is een 'full screen editor' aanwezig om machinetaal programma's te schrijven. 'Full screen' (volledige scherm) houdt in dat men met de cursor over het hele scherm kan lopen en overal veranderingen in de tekst kan aanbrengen. Als het programma klaar is kan het worden geassembleerd met de 'START'-toets.

Vanuit de monitor komt men in de editor met de 'A'-toets ("Assembler"). Als de computer net is aangezet of als zojuist het geheugen is gewist, vraagt de computer "Repair (Y/N)?". Voordat men de editor binnengaat test ZEMON eerst de tekstpointers, dat zijn de adressen die aangeven waar de tekstfile begint en eindigt. Als het geheugen is gewist hebben al deze pointers de waarde nul. Nul is echter geen geldige waarde (Op adres 0 staat rom geheugen) en ZEMON vraagt daarom of de pointers hersteld mogen worden.

Het kan ook voorkomen dat een machinetaal programma dat u heeft laten lopen de pointers heeft gesloopt. Ook dan komt de vraag "Repair (Y/N)?". ZEMON probeert dan eerst of de aanwezige tekstfile nog te behouden is (die kan ook zijn gesloopt), zoniet, dan wordt de tekstfile leeg gemaakt.

Begin en End statements

Bij een aantal opdrachten in de editor kunnen de statements Begin en End worden gebruikt. Als deze statements in de tekst worden ingetypt dan geldt de opgegeven opdracht alleen nog voor het deel van de tekst tussen Begin en End. Alle Begin- en End-statements worden gewist zodra de opgegeven opdracht succesvol is afgerond. M.b.v. Begin en End kan van de tekst in het geheugen bijvoorbeeld een deel worden weggeschreven, gewist, verplaatst of afgedrukt op de printer.

Toets functies in ZEMON's editor

Cursorpijlen

Bewegen de cursor over het hele scherm.

Shift + cursor links

Plaatst de cursor aan het begin van de regel.

Shift + cursor omhoog

Bladert een pagina (24 regels) omhoog.

Shift + cursor omlaag

Bladert een pagina (24 regels) omlaag.

Shift + cursor rechts

Maakt ruimte voor een teken (insert).

Delete

Wist het teken voor de cursor en schuift de tekens achter de cursor een positie naar links. De cursor gaat een positie naar links.

Shift + delete

Wist het teken waarop de cursor staat en schuift de tekens achter de cursor een positie naar links.

TAB

Plaatst de cursor op de volgende tabulatie positie. De tabulaties zijn afhankelijk van het maximum aantal tekens per regel dat is ingesteld (40/80). De drie tabulaties zijn in te stellen op de eerste drie posities van TABTAB (zie bijlage). Op de vierde positie van TABTAB kan eventueel nog een extra tabulatie worden ingesteld.

Shift + TAB

Schakelt caps-lock aan of uit.

Enter

Voegt een regel toe onder de regel waarin de cursor staat en plaatst de cursor in de nieuwe regel op de positie van het mnemonic.

Shift + Enter

Deelt de regel waar de cursor in staat op de plaats van de cursor doormidden.

CODE

Biedt een aantal extra mogelijkheden:

I - Insert

Schakelt om naar insert mode. (tekens invoegen)

C - Correct

Schakelt om naar correct mode. (tekens overschrijven)

H - Hex

Laat getallen in hexadecimaal zien.

D - Decimal

Laat getallen in decimaal zien.

S - Search line

Plaats de cursor in de regel met regelnummer "nr.".

M - Move text

Verplaatst de tekstfile, eventueel alleen het deel tussen het 'Begin' en het 'End' statement, naar adres "destination" in het geheugen. Als de tekst op het opgegeven adres niet kan staan wordt de foutmelding 'bad dump address' gegeven.

Cursor omhoog - Top

Plaats de cursor bovenaan de tekstfile.

Cursor omlaag - Bottom

Plaats de cursor onderaan de tekstfile.

STOP

Keert terug naar de editor

Shift + CODE

Koppelt de regel waarin de cursor staat aan de regel eronder. Let erop dat de twee regels samen langer kunnen zijn dan de toegestane lengte (40 of 80 tekens) zodat een deel van de regel verloren kan gaan.

Toets functies numeriek toetsenbord in editor

DEF - Calculate

Zelfde functie als in monitor.

Printertoets

Na bevestiging wordt de tekstfile, eventueel alleen het deel tussen het 'Begin' en het 'End' statement, afgedrukt op de printer. In de tekstfile kunnen de statements 'Prout' en 'Coloms' worden gebruikt.

Het 'Prout' statement stuurt 8-bits expressies, gescheiden door komma's, naar de printer. Zo kunnen de speciale mogelijkheden van een printer worden benut. (Bijv. 'Emphasized' afdrukken op een Epson matrix printer (of soortgelijk type): Prout 27,"E")

Met het 'Coloms' statement kan een tekstfile in kolommen worden uitgeprint. Na 'Coloms' volgen altijd drie 8-bits expressies: Coloms <Kolommen>,<Regels>,<Tekens>

Hierin is <Kolommen> het aantal kolommen waarin moet worden afgedrukt, is <Regels> het aantal regels per pagina en is <Tekens> het aantal tekens per regel in een kolom (max.80). Tussen de kolommen worden altijd twee spaties geplaatst. Als het opgegeven aantal regels is afgedrukt wordt een form-feed naar de printer gestuurd, waardoor de printer het papier naar het volgende vel doordraait. (Voorbeeld: Coloms 2,66,39 drukt 2 kolommen van 66 regels naast elkaar af, met in iedere kolom 39 tekens. De listing wordt dus $39+2+39=80$ tekens breed.)

Het 'Prout' statement werkt ook bij het uitprinten van een listing tijdens assembleren, 'Coloms' heeft dan geen invloed.

Als tijdens het zenden naar de printer de printertoets wordt ingedrukt, dan wordt de actie gestopt.

STOP

Keert terug naar de monitor

ZOEK - Search

Zoekt een trefwoord in de tekstfile vanaf de regel onder de cursor. Het trefwoord wordt in de computer bewaard.

Bliksemttoets (Shift + '2') - Clear file

Na bevestiging wordt de tekstfile, eventueel alleen het deel tussen het 'Begin' en het 'End' statement, uit het geheugen verwijderd.

START - Assemble

Assembleert de tekstfile, zie uitleg assembler.

INL - Load text

Leest een tekstfile in vanaf cassette, disk of de RS232 poort. De ingelezen file wordt automatisch in de eventueel al in het geheugen aanwezige tekst tussengevoegd onder de regel waarin de cursor staat. De al aanwezige tekst wordt dus niet gewist!

Shift + '5' - Rewind

Bekrachtigt een eventueel aanwezige terugspoelautomaat

- Save text

Schrijft de tekstfile, eventueel alleen het deel tussen het 'Begin' en het 'End' statement, naar cassette, disk of naar de seriële interface met type aanduiding ASMT (.ASM bij disk) en naam "name". Als op de cassette of disk een file wordt gevonden met dezelfde naam (op cassette wordt alleen de eerste letter getest) dan wordt gevraagd of deze overschreven mag worden ("Delete old file? (Y/N)"). Alleen voor cassette: Als de nieuwe tekst langer is dan de file op de cassette dan wordt gevraagd of de verdere files op de cassette mogen worden gewist ("Delete rest? (Y/N) "). Dit wordt ook gevraagd als er niets achter staat (er wordt niet naar het einde van de file gespoeld om te kijken of er nog iets achter staat, omdat dit bij lange files erg lang duurt en men meestal wel weet of er nog iets op de cassette staat.)

Cassette toets (Shift + '7') - Delete

Zelfde functie als in monitor.

Shift + '8'

Als in een regel iets verandert is en de cursor is nog niet van de regel af geweest, dan kan door indrukken van deze toets de oude regel worden hersteld.

Memory toets 'M' (Shift + '9') - Status

Geeft informatie over de tekstfile e.d.:

Startadres tekstfile, lengte tekstfile, inhoud tekstbuffer.

Lengte labeltabel, vrij geheugen, adres RAMTOP.

V24 baudrate, huidige regelnr, totaal aantal regels.

'-' (numeriek) - Clear buffer

Na bevestiging wordt de inhoud van de tekstbuffer gewist.

+' (numeriek) - Add buffer

De regel waarin de cursor staat wordt aan de tekstbuffer toegevoegd en de cursor wordt een regel omlaag verplaatst. De positie van de cursor op de regel wordt eveneens opgeslagen in de buffer.

'x' (numeriek) - Dump buffer

De totale inhoud van de tekstbuffer wordt onder de regel waarin de cursor staat aan de tekstfile toegevoegd. Hierbij wordt de horizontale positie van de cursor gelijk aan de positie die de cursor had toen de laatste regel aan de buffer werd toegevoegd. Dit is erg handig bij het opzetten van bijvoorbeeld tabellen.

'IXI' - Erase line

De regel waarin de cursor staat wordt gewist en de cursor wordt geplaatst op de positie van het mnemonic. (Dit is nog te herstellen met Shift + '5')

Shift + 'IXI' - Delete line

De regel waarin de cursor staat wordt volledig uit de tekstfile verwijderd. (Dit is niet meer te herstellen met Shift + '5')

ZEMON's assembler

De assembler vertaalt de tekstfile ('sourcefile') in een machinetaal programma ('object- of output-file'). ZEMON's assembler is een snelle 2-pass assembler. (ongeveer 6000 regels per minuut!)

De assembler wordt gestart met de 'START' toets vanuit de editor waarna de in het geheugen aanwezige tekstfile wordt geassembleerd. M.b.v. het 'Include' statement kunnen ook files vanaf de cassette of van de seriele interface (van een andere computer met ZEMON) worden geassembleerd.

ZEMON's assembler kent alle standaard Z80 mnemonics zoals die door Zilog zijn voorgeschreven. Ook zijn enkele niet-standaard instructies opgenomen, zoals de 'Sl' instructie, die Zilog niet opgeeft omdat de instructie in de Z80 microprocessor niet feilloos werkt (In bit 0 wordt een 1 geschoven in plaats van een 0). De instructie schuift de bits in de operand naar links. De operands bij deze instructie zijn gelijk aan die van 'Srl'. Verder kunnen de registerhelften van IX en IY afzonderlijk worden gebruikt als IXL, IXH, IYL, en IYH. (Zilog standaard voor de Z800 processor.)

De assembler kent naast de standaard operands ook enkele 'dialecten'. De toegestane afwijkingen zijn:

Standaard:	Afwijking:
Ex AF,AF'	Ex AF,AF
	of: Ex AF
In A, (#20)	In #20
Out (#10),A	Out #10
Add A,B	Add B
Sbc A,B	Sbc B

Hierdoor is het mogelijk ook files van de assembler van R. Eijnthoven (na omzetting) te assembleren. Het enige niet ondervangen verschil zijn de opdrachten 'Otdr' en 'Otir', die moeten in de andere assembler als 'OUTDR' en 'OUTIR' (niet standaard) worden ingetypt. Deze instructies worden echter op de P2000 nauwelijks gebruikt.

Het intypen van tekst regels

De mnemonics en operands mogen zowel in grote als in kleine letters worden ingetypt. Alleen bij labels worden grote en kleine letters afzonderlijk herkent (Het label "HALLO" is dus niet hetzelfde als "hallo", het register "HL" is wel hetzelfde als "hl".)

Het is niet van belang op welke positie op de regel de mnemonics en operands ingetypt worden, de editor zet ze, indien het mnemonic geldig is, automatisch op de goede positie. Deze posities zijn te veranderen door de tabulatie posities te veranderen (zie 'TAB' toets in editor).

Commentaar mag op alle posities op een regel worden geplaatst. Voor commentaar moet altijd een puntkomma ";" staan, die aangeeft dat de rest van de regel commentaar is. Als de puntkomma van het commentaar exact op de laatste tabulatie positie wordt getypt heeft dat twee voordelen: De opslag van de regel in het geheugen kost dan 1 byte minder en de plaats van het commentaar op de regel kan verandert worden door eenvoudig de laatste tabulatie positie te veranderen (zie 'TAB' toets in editor).

Een label moet altijd op de eerste positie van de regel beginnen om als zodanig herkent te kunnen worden.

Labels

Labels moeten beginnen met een teken met ascii waarde tussen #41 ("A") en #7A ("z"), m.u.v. "#". Dat zijn alle letters, groot en klein, plus de tekens "[", "\", "]", "^", "-" (met shift).

Voor de rest van het label mogen dezelfde tekens gebruikt worden, plus alle cijfers en de tekens "\$", ".", "<", "=", ">". Labels mogen niet als operand worden herkend. Zo is bijvoorbeeld het label "HL" verboden, maar "HLA" is weer wel toegestaan.

Van een label worden normaal alleen de eerste zes tekens opgeslagen (M.b.v. een systeemvariabele (LABLEN, zie bijlage) kan dit worden ingesteld tussen 2 en 20 tekens). Dat houdt in dat de labels HELLOFANS en HELLOFISH beide worden opgeslagen als HELLOF en dus gelijk zijn aan elkaar. Als een label uit minder dan zes tekens (of een andere ingestelde lengte) bestaat, worden ook minder tekens in het geheugen opgeslagen, waardoor geheugen bespaard wordt.

ZEMON assembler pseudo-mnemonics

Pseudo-mnemonics zijn geen Z80 instructies, maar dienen om extra mogelijkheden in de assembler te kunnen gebruiken. In de tabel met ZEMON's pseudo-mnemonics worden de letters b en w gebruikt als afkorting voor byte (8-bits expressie) en word (16-bits expressie).

Defb b1,b2..bn

Byte b1,b2..bn

De 8-bits expressies worden na elkaar op het huidige assembleeradres neergezet, na ieder byte wordt de locatieteller (programcounter) 1 verhoogd. Met Defb of Byte kunnen data in een programma worden opgenomen.

Defw w1,w2..wn

Word w1,w2..wn

De 16-bits expressies worden op de standaard Zilog manier (eerst het lage-, dan het hoge orde byte) op het huidige assembleeradres neergezet, waarna de locatieteller met twee verhoogd wordt. Defw (of Word) kan worden gebruikt om bijvoorbeeld 'jump-tabellen' op te zetten.

Defm "<TEKST1>","<TEKST2>"...<TEKSTn>"

Asc "<TEKST1>".

De tekst tussen de aanhalingstekens wordt op het huidige assembleeradres neergezet als ascii waarden. Voor ieder teken wordt de locatieteller een opgehoogd. Als achter het laatste aanhalingsteken een ^ (Shift + '@') wordt gezet, dan wordt bit 7 van het laatste byte van <TEKST> hoog gemaakt. Met Defm kunnen eenvoudig berichten in een programma worden opgenomen.

Defs w

Defs verhoogt de locatieteller met de waarde van w, waardoor in het programma een lege ruimte ontstaat, die bijvoorbeeld kan dienen als tekstbuffer. Achter Defs mag niet voorwaarts gerefereerd worden (§).

Defr b,w

Het 8-bits getal b wordt vanaf het huidige assembleeradres w maal neergezet. Hiermee kunnen grote delen van het programma van een 8-bits waarde worden voorzien. In de expressie w mag niet voorwaarts gerefereerd worden (§).

= w

Equ w

Voor Equ (of =) MOET altijd een label staan. Dit label krijgt de waarde w. Achter Equ (of =) mag niet voorwaarts gerefereerd worden (§).

Org w

Maakt de locatieteller gelijk aan w. Hiermee weet de assembler op welk adres in het geheugen het programma moet kunnen werken. Als geen Org-statement wordt gebruikt begint de locatieteller op adres 0. Normaal komt het programma ook op adres w in het geheugen (zie Direct/Indirect). In een file mogen meerdere Org's voorkomen, zodat het programma door het hele geheugen verspreid kan worden. Achter Org mag niet voorwaarts gerefereerd worden (§).

List ON/OFF

Zet de listing naar het scherm aan of uit. De listing wordt pas in de tweede door-gang (pass) gegenereerd.

Printer ON/OFF

Zet de listing naar de printer aan of uit.

Key

Drukt op het scherm af "Press key..." en wacht tot een toets wordt ingedrukt.

Labels ON/OFF/USED

De labels die gedeclareerd zijn (een waarde hebben gekregen) tussen Labels On en Labels OFF worden na het assembleren op het scherm, en eventueel op de printer, afgedrukt. Als op een willekeurige plaats in de tekst Labels USED wordt getypt dan wordt na het assembleren een lijst van alle niet-gebruikte labels gegeven.

(§) Voorwaarts refereren betekent verwijzen naar een label dat pas later in de tekst een waarde krijgt toegewezen. In expressies die de lengte van de objectfile kunnen beïnvloeden is dit niet toegestaan.

Indirect

Indirect w

Met Indirect kan de object van een programma op een ander adres worden neergezet dan het adres waarvoor het geassembleerd wordt (waar het moet werken). Als er geen expressie achter Indirect staat, dan wordt de object op het eerste vrije adres achter de labeltabel gezet. Dit eindadres van de labeltabel wordt na de tweede doorgang van de assembler getoond: "Symbol table from xx to yy". Hierin is xx het begin, en yy het eind van de labeltabel. yy is tevens het adres waar dan de object begint.

Als er wel een expressie achter indirect staat wordt de object op dit adres neergezet. Voor het assembleren van een programma dat in een rom-module moet worden gestoken kan worden getypt: regel1, Org #1000; regel2, Indirect.

Direct

Na Indirect schakelt Direct weer terug naar het gebruik van de Org-waarde als adres waar de object geplaatst moet worden.

Include "<NAAM>"

Bij grote programma (groter dan 7 kbyte op een 40k machine) is het beschikbare geheugen niet toereikend om zowel tekst als de labeltabel en de objectcode te bevatten. Om nu toch grote programma's te kunnen schrijven moet de tekstfile opgedeelt worden in twee of meer delen, die afzonderlijk op cassette worden weggeschreven. Ieder deel van de tekstfile dat op een cassette is geschreven kan dan uit het geheugen worden verwijderd en vervangen worden door de regel Include <NAAM>. Hierin is <NAAM> de naam waaronder u dat deel van de tekst op cassette heeft gezet.

Als nu wordt geassembleerd, wordt bij het Include-statement de opgegeven file vanaf de cassette geassembleerd. Hiertoe wordt blok voor blok van de cassette geladen en geassembleerd. Omdat de assembler twee doorgangen door de tekst moet maken wordt de tekst twee keer geladen. Als er meerdere teksten vanaf verschillende cassettes geincludeerd moeten worden is het nuttig om tussen de Include-statements een Key-statement op te nemen, zodat rustig van cassette kan worden gewisseld.

Include kan ook gebruikt worden om veel-gebruikte subroutines bij het programma te assembleren.

Het include-statement mag niet in een includeerfile voorkomen.

Als "R:" voor <NAAM> staat, dan wordt een tekstfile van de seriele interface gelezen. Op de printerpoort moet dan een andere computer met ZEMON zijn aangesloten. De andere computer moet dan een tekstfile zenden met de opdracht "RS include" (bij CODEtoets).

Norew

Als voor Include een Norew-statement (No rewind) staat wordt de cassette niet teruggespoeld bij het zoeken naar de bij Include opgegeven tekstfile. Dit is handig als meerdere includeerfiles op een cassette staan.

Header "<TEKST>"

In de assembler listing worden een lege regel, een regel met <TEKST> en nog een lege regel toegevoegd. Het Header-statement mag niet in een includefile voorkomen. Header kan worden gebruikt om een deel van de listing d.a.v. een kopje te scheiden van een ander deel.

Eject

Heeft het zelfde resultaat als Header, alleen wordt dezelfde tekst, zoals bij het vorige Header-statement is ingetypt, nu nogmaals afgedrukt. Als nog geen Header is gespecificeerd wordt 1 lege regel in de listing toegevoegd.

Pagelen b1

Pagelen b1,b2

Pagelen b1,0,b3

Pagelen b1,b2,b3

b1 = Het aantal te printen regels per pagina. (b1>4, bijv. 66 voor 12 inch papier) Achter pagelen moet altijd het aantal regels per pagina worden opgegeven.

b2 = Het aantal lege regels dat moet worden geprint om op het volgende vel papier te komen. Als niets wordt opgegeven of nul wordt ingevuld stuurt de assembler geen lege regels naar de printer, maar een form feed code (12) om op een nieuwe pagina verder te printen.

b3 = Het pagina nummer van de eerste pagina die geprint wordt. Als geen pagina nummer wordt opgegeven begint de assembler met pagina nr. 1.

Als de listing naar het scherm of naar de printer aanstaat drukt Pagelen de regel af die ook verschijnt als geassembleerd wordt ("ZEMON Z80 Assembler.." enz.). Achteraan deze regel komt het pagina nummer, daarna komt 1 lege regel. Als met het Title-statement een titel is gedefinieerd wordt ook deze titel afgedrukt met daarbij nog een lege regel. Als het in b1 opgegeven aantal regels is afgedrukt wordt, afhankelijk van de waarde van b2, een form feed of een aantal lege regels naar de printer gestuurd. Als b1 kleiner is dan 5 dan wordt de automatische pagina indeling weer uitgeschakeld.

Page

Page b3

Wist het scherm en stuurt een formfeed code naar de printer (als de listing naar het scherm en de printer aan staat), daarna wordt op de volgende pagina een kopje met paginanummer afgedrukt en eventueel een titel, exact zoals beschreven bij Pagelen. Als in Pagelen b2 ongelijk was aan nul, dan wordt in plaats van een formfeed naar de printer te sturen het resterende aantal regels op de pagina doorgedraaid. Het Page-statement kan gebruikt om een nieuw deel van het programma op een nieuwe pagina te laten beginnen. Als een waarde voor b3 wordt ingevuld dan krijgt de nieuwe pagina het pagina nummer b3.

Title "<TEKST>"

De tekst die achter Title wordt gedefinieerd wordt bij ieder Page-statement op de nieuwe pagina afgedrukt, zie Pagelen. Het Title-statement mag niet in een include-file voorkomen.

Prout b1,b2..bn

Als de listing naar de printer aan staat worden de waarden van de 8-bits expressies naar de printer gezonden.

If w

Schakelt het assembleren aan of uit, afhankelijk van de waarde van w. Als w kleiner of gelijk is aan 0, dan wordt het assembleren uit geschakeld, is w groter of gelijk aan 1 dan wordt het assembleren weer aan geschakeld. Zo kan een deel van de tekst tijdelijk worden weggelaten uit de objectfile, zonder dat het uit de tekst verwijderd hoeft te worden. Door achter If een label te plaatsen kan vooraan de tekst worden bepaald of bepaalde delen wel of niet moeten worden geassembleerd.

Schakelt het assembleren aan als het uit was, en uit als het aan was.

Schakelt het assembleren weer aan.

Begin

End

Coloms b1,b2,b3

Hebben alle geen invloed tijdens assembleren, ze werken alleen vanuit de editor.

De regels waarin de statements List, Printer, Labels, Include, Key, Coloms, Prout, Header, Page, Eject, Pagelen, Title, If, Else, Endif en Norew worden tijdens het assembleren niet op de printer of op het scherm afgedrukt. Als een tekst vanuit de editor met de printertoets wordt afgedrukt worden deze regels, behalve die met Coloms of Prout, wel op de printer afgedrukt.

Expressies

Zowel in de monitor, de editor als in de assembler mag voor ieder getal wat moet worden ingetypt ook een expressie (berekening) worden ingetypt. Een expressie heeft altijd de volgende vorm:

<factor> <operator> <factor>

Een expressie begint en eindigt altijd met een factor, tussen de factoren staat altijd een operator. De operatoren + en - mogen ook in een factor worden opgenomen (bijv. -3 of +--#7 (=7)).

ZEMON kent de volgende operatoren:

- Optellen van twee factoren
- Aftrekken van twee factoren
- Vermenigvuldigen van twee factoren
- Delen van twee factoren (De deling gebeurt met gehele getallen, het resultaat wordt naar beneden afgerond)
- Modulus operatie, rest na deling (29?8=5)
- & Binaire AND-operatie
- Binaire OR-operatie

In ZEMON kunnen de volgende factoren worden gebruikt:

- Decimale getallen. Een decimaal getal kan zonder meer worden ingetypt, Bijv. 10, -27, 0 enz.
- 2. Hexadecimale getallen, voorafgegaan door een '#' ('hekje'), om het te onderscheiden van decimale getallen of labels, bijv. #10, -#AE00, #7AF, #3d5b enz.
- 3. Binaire getallen, voorafgegaan door een '%' ('procent'), bijv. %110 (=6), %11100101, %10110111101, %0 enz.
- 4. Machten van 2, voorafgegaan door een '^' ('absoluutstreep'). Het getal achter '^' moet liggen tussen 0 en 7, en mag ook een label of een expressie tussen haakjes zijn. Zo kan het byte worden berekend waarin een of meer bits hoog zijn, bijv. %3 (=8), %3, %2 (=4), %2+%3 (=12), %2+3 (=7), %(%2+3) (=15=32), %H01, %(%H01+2) enz.
- 5. Asciiconstanten, ingesloten door aanhalingstekens ''. De waarde van de asciiconstante is de waarde van het ingetypte ascii teken. Eventueel kan achter het sluitende aanhalingsteken een '^' (Up-arrow, ascii #5E, toets Shift + '@') worden getypt wat aangeeft dat het zevende bit van het byte hoog wordt. Voorbeelden: "A" (=41), "", "q", "Z"^ (=8DA), "A"%5 (=145) enz.
- Labels. De regels die hiervoor gelden worden bij de uitleg van de assembler vermeld.
- 7. \$ ('dollar-teken'). \$ is de waarde van de locatieteller (programcounter) bij het assembleren. Zo is "Jr \$" een eindeloze loop.

In expressies mogen ook haakjes worden gebruikt. Om verwarring met de haakjes om een operand in de assembler (bijv. (HL) of (IY+3)) te vermijden worden als haakjes de tekens { en } (ascii #7B en #7D) gebruikt. Op de P2000 zijn dat de tekens 1/4 en 3/4. Haakjes mogen binnen elkaar worden gebruikt.

Denk er bij berekeningen aan dat getallen groter dan 65536 (#FFFF) niet kunnen worden verwerkt. Daardoor is bijvoorbeeld het getal #FFF9 gelijk aan -7 (Twee complement).

Foutmeldingen

Out of memory	Geheugen vol. Als deze foutmelding optreedt tijdens assembleren betekent het dat de labeltabel voor het programma dat wordt geassembleerd niet in het geheugen past. Als de foutmelding in de editor verschijnt is de tekstfile samen met de tekstbuffer te groot.
Illegal token code	De tekstfile is om een of andere reden gesloopt, en bevat illegale tekens met een waarde boven #E2. Normaal komt deze fout nooit voor.
No labels	Er is geen labeltabel aanwezig, of de aanwezige labeltabel is gesloopt.
Illegal label	In een label zijn illegale tekens gebruikt, of een label wordt voorafgegaan door spaties. (zie hoofdstuk: Labels)
Label not found	Het ingetypte label in de editor of de monitor is niet in de labeltabel aanwezig. Deze foutmelding komt niet voor tijdens assembleren.
'}' missing	In de ingetypte expressie zijn meer openende dan sluitende haakjes gebruikt.
Number too big	Getal te groot. Het getal is groter dan de maximaal toegestane waarde.
Bad characterconstant	Het sluitende aanhalingsteken in een asciiconstante (ascii waarde) is vergeten.
Illegal expression	De expressie (berekening) bevat een teken dat geen geldige operatie vertegenwoordigt.
Bad number	Het getal bevat illegale tekens voor de gekozen numerieke basis.
Label already defined	Het label heeft al eerder een waarde gehad. Het label behoudt de oude waarde.
Label expected	Er is een Equ-statement (of een =) gebruikt maar er staat geen label voor.
Illegal forward	Er wordt gerefereerd naar een label dat nog geen waarde toegekend heeft gekregen. Bij sommige opdrachten moet de waarde van het label al in de 'first pass' (eerste doorgang) bekend zijn. (Bijv. bij Equ, Org, Defr en Defs,If)
Unknown operand	De gebruikte operand is niet geldig voor de instructie die ervoor staat.
'(' missing	Het sluitende haakje is vergeten in een (IX+ of (Y+ operand.
Jump out of range	Een opgegeven relatieve sprong (Jr of Djnz) kan niet worden gehaald, is buiten het sprongbereik. De opdracht die de fout veroorzaakt wordt door de assembler vervangen door twee Nop-instructies, zodat het geassembleerde programma eventueel toch kan worden gestart, de lengte van de objectfile is toch goed.
Bad dump address	<p>Er wordt geprobeerd de object te plaatsen in de tekstfile, de labeltabel of de systeemvariabelen van ZEMON. De plaats van de object van het programma dat geassembleerd wordt is fout. Als de object daar geplaatst zou worden zou de assembler vast kunnen lopen. Om de fout te voorkomen kunt u een van de volgende maatregelen treffen: De tekstfile verplaatsen (CODE + Move text), het Org-adres van het programma veranderen, het programma m.b.v. Indirect voorlopig op een andere plaats zetten of Include gebruiken om de tekstfile in het geheugen te verkleinen. Na deze fout kan wel verder worden geassembleerd, er wordt echter geen object meer geproduceerd.</p> <p>De fout kan ook voorkomen bij het verplaatsen van tekstfile met de "Move text" opdracht. De tekstfile kan dan niet op het opgegeven adres worden geplaatst. Als deze fout optreedt tijdens het omzetten van een file van de assembler van R. Eijnhoven, dan betekent dit dat er in die tekstfile iets fout is.</p>

Printer off	De printer is niet aangesloten of staat 'Off line'. Als de computer wacht op een ready signaal van de printer en de printertoets (Shift + '00') wordt ingedrukt, dan verschijnt ook deze foutmelding en wordt de printer uitgeschakeld, zowel in de monitor als de editor of de assembler.
Not in includefile	Deze foutmelding verschijnt als in een includefile een van de opdrachten Include, Header of Title voorkomt.
Not found	Niet gevonden. Bij de zoek opdrachten betekent dit dat het opgegeven trefwoord of de opgegeven codes niet zijn gevonden.

Cassette foutmeldingen

No tape	Geen cassette. De cassette zit niet in de recorder of is er tijdens het draaien van de recorder uitgehaald. (Tape error A)
Checksum error	Leesfout. (Tape error C, D, J of L)
End of tape	Einde cassette, cassette vol. (Tape error E of F)
Write protect	Cassette is beveiligd tegen schrijven, geen stopje. (Tape error G)
Tape too long	De band in de cassette is te lang of gebroken. (Tape error I)
Not found	Niet gevonden. Het gezochte programma staat niet op deze kant van de cassette. (Tape error M of N)

Disk foutmeldingen

Disk error	Er zit geen disk in de drive / het deurtje van de drive is open / de disk is niet met ZEMON geformatteerd / er heeft zich een ernstige fout op de schijf voor gedaan.
Disk full	Het programma past niet meer op deze kant van de schijf
Illegal name	De ingetypte filenaam is niet geldig.
Illegal function	Er worden disk opdrachten gegeven terwijl ZEMON in cassette-stand staat
Not found	De gezochte file staat niet op deze kant van de schijf.

RS 232 foutmeldingen

Checksum error	Het stopbit dat na iedere acht bits die over de seriële interface verzonden zijn hoort te volgen, is niet gedetecteerd.
Break	Tijdens zenden of ontvangen van de RS232 is op de printertoets (Shift + '00') gedrukt, waardoor de actie is onderbroken.

Na een foutmelding in de monitor wordt weer teruggekeerd naar het commandoniveau in de monitor. Na een foutmelding in de editor wordt op een willekeurige toetsdruk gewacht, waarna weer wordt teruggekeerd in de editor. Bij een foutmelding tijdens assembleren wordt de regel waaruit de fout voortkwam getoond en daarna op een toets gewacht. Wordt STOP ingedrukt, dan komt men direct weer in de editor, wordt een andere toets ingedrukt dan wordt verder geassembleerd. In veel gevallen is de geproduceerde objectfile nog uitvoerbaar.

In ZEMON is het woord 'Illegal' in de foutmeldingen afgekort tot 'Ill.', om geheugen te besparen

Inlezen en wegschrijven

Files wegschrijven of inlezen kan bij ZEMON niet alleen met op cassette, maar ook op disk of over de seriele interface (printerpoort, RS232 of V24). Als een file moet worden weggeschreven of ingelezen van de seriele interface, dan moeten voor de naam de tekens 'R:' worden getypt. Wil men bijvoorbeeld de tekstfile, of alleen het deel tussen het 'Begin' en het 'End' statement, wegschrijven naar de seriele interface, dan drukt men vanuit de editor op OPN en typt dan R:Naam + <Enter>. De baudrate kan worden ingesteld met de opdracht "V24 baudrate" in de monitor.

Om met de diskdrives te kunnen werken moet voor de naam het drivenummer en een ":" worden getypt. Wil men bijv. een tekstfile van kant 2 van drive 1 inlezen, dan typt men vanuit de editor INL + 3:Naam + <Enter>. Bij het lezen van de disk wordt op alle letters van de naam getest, ook de extentie.

Zo kunnen alle cassette operaties worden uitgevoerd met de disk of de seriele interface, door alleen iets voor de naam te typen. Omdat bij het lezen van de seriele interface niet op een naam wordt getest is het voor het lezen van een file voldoende om als naam in te typen R:.

Ook het 'Include' commando in de assembler kan met de seriele interface worden gebruikt. De computer die als geheugenbank dient, moet dan een tekstfile naar de seriele interface schrijven (2 keer, voor de beide doorgangen). In de andere computer moet dan in de tekstfile de volgende regel worden opgenomen:

Include "R:"

Tijdens het assembleren van de tekst wordt bij het passeren van het 'Include' statement de tekstfile van de andere computer mee geassembleerd.

Bij het zoeken van een file op de cassette of de seriele interface worden altijd de naam en de lengte van de laatst gevonden file op het scherm getoond. Na iedere cassette operatie wordt een eventueel aanwezige terugspoelautomaat bekrachtigd.

Disk drives

Met ZEMON kunnen 2 enkel- of dubbelzijdige 40-tracks drives worden gebruikt. De drives zijn als volgt genummerd: 1=Drive1, kant1 / 2=Drive2, kant1 / 3=Drive1, kant2 / 4=Drive2, kant2. ZEMON werkt met sectoren van 1K, waarvan er vijf op een track gaan. Hierdoor kan op een kant van een schijf (bij 40-tracks) 200K worden opgeslagen. Het is ook mogelijk om minder dan een hele sector te lezen. De eerste sector van de schijf is altijd in gebruik voor de sectormap, waarin staat welke sectoren gebruikt zijn, en door welke file ze zijn gebruikt. Op de tweede sector staat altijd het eerste deel van de 'directory' (inhoud), waarin van iedere file de naam, startadres en lengte staat.

Als een file op disk wordt geschreven wordt automatisch achter de naam een extentie gezet. In de monitor is dat ".BIN", in de editor ".ASM". Als men zelf al een extentie van drie letters achter de naam typt, dan zet ZEMON er geen extentie bij. Zo kan men ook een .BIN-file in de editor inlezen, door achter de naam de extentie in te typen. Typt men bij het inlezen geen extentie, dan zoekt ZEMON alleen naar files met extentie .ASM (of .BIN in de monitor).

Printers

Bij ZEMON kunnen vele printers worden gebruikt. Alle printers met een seriele aansluiting (=V24=RS232) kunnen zonder meer worden aangesloten. De snelheid waarmee gegevens naar de printerpoort worden gezonden (de baudrate) is instelbaar tussen 75 en 19200 baud met de opdracht "V24 baudrate".

Voor printers met een parallelaansluiting (Centronics) is een interface nodig om de printer aan te kunnen sluiten op de P2000. Om zo'n printer dan te kunnen gebruiken moet een eigen printerroutine (printerdriver) worden geïnstalleerd. Daarvoor moet u de pointer naar het einde van het geheugen (RAMTOP, zie bijlage) omlaag verplaatsen om een vrij stuk geheugen te creëren om de printerroutine in te plaatsen. Dan moet de pointer naar de normale printerroutine (PROVEC, zie bijlage) verplaatst worden naar de nieuwe routine. Als bij het terugkeren uit de printerroutine de carryflag geSET is, verschijnt de foutmelding "Printer off".

Na iedere regel die vanuit ZEMON wordt afgedrukt op de printer wordt een afsluitcode naar de printer gestuurd, normaal is dat een code #A. Voor sommige printers is het echter vereist dat de printer na iedere regel een #A en een #D krijgt toegestuurd. Daarvoor moet men op adres CRPROUT (zie bijlage) het byte #D (0D) zetten.

Deze veranderingen in de systeemvariabelen kan men maken met de Memory-edit opdracht ('E'-toets in monitor).

De seriele interface

De seriele interface (=RS232, =V24, =printerpoort) wordt in ZEMON niet alleen gebruikt voor de printer maar ook voor communicatie tussen twee computers. De baudrate is instelbaar op alle gangbare snelheden voor RS232 interfaces. Hierdoor is het mogelijk ook met andere computers dan de P2000 te communiceren.

Bij de ZEMON opdrachten voor het uitprinten van gegevens op een printer worden van de seriele interface de volgende aansluitingen gebruikt:

- Pen 3 (Uitgang) : Data out.
- Pen 20 (Ingang) : Printer ready signal.
- Pen 7 (Massa) : Signal ground.

Voor het zenden naar de printer kan dus een normale printerkabel gebruikt worden, zoals die bij alle andere P2000 programma's wordt gebruikt.

Bij het zenden en ontvangen over de seriele interface worden bij ZEMON slechts twee signaal leidingen en een massa aansluiting gebruikt. De twee signaal leidingen hebben beide een dubbele functie.

Functie tijdens zenden:

- Pen 2 (Ingang) : Request to send.
- Pen 3 (Uitgang) : Data out.
- Pen 7 (Massa) : Signal ground.

Functie tijdens ontvangen:

- Pen 2 (Ingang) : Data in.
- Pen 3 (Uitgang) : Request to send
- Pen 7 (Massa) : Signal ground.

"Request to send" wordt door de ontvangende computer laag gemaakt om aan te geven dat de ontvangende computer klaar is om data te ontvangen. De data wordt verzonden over de lijnen "Data in" en "Data out". De zendende computer wacht voor ieder byte dat gezonden moet worden tot de lijn "Request to send" van de andere computer laag wordt. Hierdoor blijft de computer die een file wegschrijft wachten tot de andere computer klaar is om iets te kunnen ontvangen.

RS232 file-operaties kunnen worden onderbroken met de STOP-toets, printer acties met de printertoets.

Om twee P2000 computers te verbinden is nu slechts een twee-aderig afgeschermd kabel nodig, waarin de twee signaal leidingen gekruist zijn aangesloten. Om te kunnen communiceren met andere computers moet men eerst weten welke aansluitingen deze heeft. Bij een computer met een UART of SIO o.i.d. erin worden meestal vier signaal lijnen gebruikt. Om deze op de P2000 aan te kunnen sluiten moet in de kabel een schakelaar worden aangebracht voor zenden/ontvangen.

Het formaat waarin bytes worden overgezonden is een standaard formaat. Normaal, als de lijnen niet gebruikt worden, zijn ze alle hoog, logisch "1", wat volgens de RS232 standaard inhoudt dat de spanning op de leidingen dan tussen -5V en -15V t.o.v. massa moet zijn. Zoals gezegd wacht de zendende computer eerst op een laag (logisch "0", spanning tussen +5V en +15V) wordende "Request to send"-lijn, waarna er 1 byte wordt overgezonden. Het zenden begint met het startbit, dat altijd laag is. Hieraan kan de ontvangende computer zien dat het zenden is begonnen. na dit startbit komen de acht data-bits, eerst bit 0, en als laatste bit 7. Na de data-bits volgt het stopbit, dat altijd hoog is. ZEMON werkt niet met pariteit-bits of een checksum.

Als een file over de RS232 interface wordt weggeschreven, wordt eerst een 'header' weggeschreven. Hierin staan de naam van de file, de type aanduiding, het startadres en de lengte. De header die ZEMON gebruikt ziet er als volgt uit: 32 bytes naam, 2 bytes startadres, 2 bytes lengte, 4 bytes leeg, 4 bytes type aanduiding, 4 bytes leeg. De header is altijd 48 (=30) bytes lang.

In Memory-edit zou de header er zo uit zien:

```
6100 44 69 74 20 69 73 20 64 Dit is d
6108 65 20 72 75 69 6D 74 65 e ruimte
6110 20 76 6F 6F 72 20 64 65 voor de
6118 20 6E 61 61 6D 2E 20 20 naam.
6120 00 61 30 00 00 00 00 a0
6128 41 53 4D 54 00 00 00 00 ASMT
```

ZEMON's memory-map

	User RAM	Vrij geheugen waarin alle files worden opgeslagen.

	ZEMON systeemvariabelen	Het voor ZEMON benodigde geheugen voor pointers e.d.

	Monitor ROM systeemvar.	Pointers e.d. voor de monitor ROM.

#5800	Video attributes M-model	Tweede video pagina, attributen, alleen in M-model.

	Video memory	Video geheugen voor 24 regels van 40 of 80 tekens
#5000	-----	
	ZEMON ROM	

	Monitor ROM	

Het vrije geheugen (User RAM) begint bij ZEMON op #6300. Hierdoor blijft er meer geheugen vrij dan bij de meeste andere modules. Dit vrije geheugen loopt door tot het einde van het geheugen, #9FFF bij een 16K P2000, #DFFF bij een 32K P2000 en #FFFF bij een 40K P2000.

Het vrije geheugen wordt gebruikt om programma's, de tekst en de labeltabel op te slaan. Als u een 'Clear RAM' opdracht heeft gegeven en daarna met de full screen editor gaat werken, staat uw tekst altijd op adres #6300. Wanneer u de tekst assembleert zal in het geheugen achter de tekst de labeltabel (symbol table) komen. Het adres waar de objectfile van uw programma komt bepaalt u zelf met een 'Org' of een 'Indirect' instructie. Deze object mag natuurlijk niet in de tekst, in de labeltabel, of in de ZEMON-systeemvariabelen worden geplaatst, gebeurt dit wel, dan volgt de foutmelding 'Bad dump address'.

Wilt u de object van uw programma laag in het geheugen plaatsen, bijvoorbeeld op adres #6300, dan moet eerst de tekst omhoog worden verplaatst. Dat kan met de opdracht 'Move text' (bij CODE-toets). U moet dan wel zorgen dat er achter de tekst nog voldoende ruimte is voor de labeltabel van uw programma, anders kan het programma niet worden geassembleerd. (De labeltabel komt altijd achter de tekst.)

In ZEMON is een 'pointer' aanwezig (RAMTOP, zie systeemvariabelen) die aangeeft waar het einde van het geheugen zich bevindt. De tekst en de labeltabel van de assembler gaan nooit voorbij deze pointer. Na reset is deze pointer gelijk aan het hoogste geheugenadres, maar u kunt deze pointer ook verplaatsen, zodat het gebied boven de pointer beschermd is tegen per ongeluk overschrijven. Let er wel op dat een objectfile er nog wel in kan worden geplaatst!

Verklarende woordenlijst

Deze lijst is een poging om een aantal termen die in ZEMON en in deze gebruiksaanwijzing voorkomen nader te verklaren. In hoeverre dat gelukt is mag u zelf uitmaken....

ASCII	American Standard Code for Information Interchange, zie pagina ASCII.
Baud	Eenheid van snelheid bij data communicatie. 1 baud is 1 bit per seconde.
Baudrate	Snelheid van data communicatie. (gebruik: "De baudrate is 1200 baud.")
Binaire	Tweetallig stelsel, met als cijfers 0 en 1.
Computerfreak	Persoon met een ernstige psychische afwijking die zelfs door Freud is onderkend. De belangrijkste fysieke symptomen zijn tintelende vingers en allergie voor slapen.
Control code	Code met een speciale besturingsfunctie.
Data	Gegevens (meervoud).
Decimaal	Normale tientallige stelsel, met als cijfers 0123456789.
Editor	Programma dat het mogelijk maakt gegevens te veranderen.
End-of-file-mark	Speciaal blok dat op de minicassettes wordt geschreven om aan te geven dat achter dat blok geen programma's meer staan.
Error	Fout.
Expressie	Berekening.
File	Blok gegevens, bijv. tekst, object enz.
Formfeed	Opdracht die het papier van de printer 1 vel verder draait. (of het vel papier uit de printer draait)
Full screen editor	Editor waarbij men over het hele scherm kan bewegen en overal veranderingen kan aanbrengen.
Hex, hexadecimaal	Zestientallig stelsel, met als cijfers 0123456789ABCDEF.
Memory-map	Overzicht van de geheugen indeling.
Objectfile	Outputfile. Het machinetaal programma dat door een assembler o.i.d. is geproduceerd.
Operator	Rekenkundige functie (bijv. +, / enz.).
Pass	Doorgang van de assembler door de tekst.
Printerdriver	Programma dat de tekens die er naar toe worden gestuurd naar de printer doorstuurt.
Programcounter	Locatieteller. In de Z80 microprocessor is dit het register dat aangeeft waar de volgende instructie uit het geheugen moet worden gehaald. In de assembler is het de pointer die aangeeft waar de het volgende byte object zou moeten staan (het Org-adres).
Sector	Deel van een track op een schijf. De grootte van een sector kan softwarematig worden ingesteld. Bij ZEMON is een sector 1 kbyte (1024 bytes). Op een track passen 5 sectoren van 1K, zodat er met ZEMON 5K op een track kan worden geschreven.
Sourcefile	Tekstfile. De tekst die in de full screen editor wordt ingetypt. (De tekst waarop een bewerking wordt verricht)
Statement	Instructie.
Symbol table	Label tabel. De tabel in het geheugen waarin alle symbolische referenties (labels) met de bijbehorende waarden worden opgeslagen.
Systeemvariabelen	RAM-geheugen waarin gegevens van een programma worden bijgehouden.
Track	Spoor op een floppydisk. (35, 40 of 80 per schijf, afhankelijk van de diskdrive)

ASCII

ASCII is een internationaal gebruikte standaard voor het uitwisselen van gegevens tussen computers, waarbij ieder teken (letter, cijfer, leesteken enz.) een eigen code heeft. Bijna iedere zichzelf respecterende computer werkt met deze codes, maar Philips vond dat de P2000 hier wel van af mocht wijken en gebruikte de Teletekst standaard (Nouja, standaard...). Onder deze teletekst tekens bevinden zich een aantal vreemde tekens die geen mens ooit nuttig kan gebruiken, waardoor een aantal tekens die in de ASCII standaard zaten verdwenen zijn, en die tekens waren wel nuttig! Hieronder staan de verschillen tussen ASCII en Teletekst, met daarachter het teken dat ZEMON print als de hex code naar de printroutine wordt gestuurd.

Hex-code	ASCII	Teletekst	ZEMON
#23	#	Pondteken	#
#5B	[<-	In het M-model: [, in T-model: <- (T-model kent geen blokhaken).
#5C	\	1/2	1/2 (P2000 kent geen \).
#5D]	->	In het M-model:] , in T-model: -> (T-model kent geen blokhaken).
#5F	-	#	- (In het T-model is deze '-' langer dan het gewone min-teken).
#60	-	-	als #5F.
#7B	{	1/4	1/4 (P2000 kent geen accoladen).
#7C			Het T-model heeft geen normale absoluutstreep, alleen een dubbele.
#7D	}	3/4	3/4 (P2000 kent geen accoladen).
#7E	~	Deelteken	Deelteken (:-)

Bruikbare adressen in ZEMON's systeemvariabelen

ZEMON's systeemvariabelen beginnen op adres #6070 en eindigen op #6300. In dit gebied houdt ZEMON al zijn gegevens bij. Een aantal van deze gegevens mag ook worden veranderd. Veranderingen aanbrengen in de systeemvariabelen moet wel voorzichtig gebeuren want een verkeerde verandering kan fataal zijn. Voor de adressen van de systeemvariabelen wordt een apart vel toegevoegd, omdat deze in een eventuele volgende versie kunnen veranderen.

Naam	Functie
------	---------

LNAD	Line address. Adres van eerste positie op een regel in van scherm-printroutine (PR).
TABTAB	Tabulatie tabel. Hierin staan de drie of vier tab-posities van de editor.
RAMTOP	Adres van de hoogste vrije geheugen positie. (zie pagina Memory-map)
SOT	Start of text. Begin adres van de tekstfile in het geheugen.
SOB	Start of buffer. Begin adres tekstbuffer = eind adres tekstfile.
EOB	End of buffer. Eind adres tekstbuffer.
SOL	Start of labels. Begin adres labeltabel.
EOL	End of labels. Eind adres labeltabel.
X.POS	Horizontale positie op de regel in de scherm-printroutine (PR).
X.MAX	Maximale horizontale positie op een regel. (39 of 79)
LABLEN	Label lengte. De waarde van dit adres bepaalt het aantal tekens dat van de labels wordt opgeslagen (instelbaar van 2 tot 20 tekens).
PROVEC	Printer-out vector. Adres van printerdriver.
INITBUF	Printer initialize buffer. Alle tekens uit deze buffer, maximaal 10, worden bij het initialiseren van de printer naar de printerdriver gestuurd. Als er minder dan 10 tekens in de buffer staan, moet als laatste byte een #FF worden toegevoegd. Initialiseren van de printer gebeurt bij het indrukken van de printertoets ('Printer on'), of bij de eerste 'Printer ON'-instructie tijdens assembleren.
CRPROUT	Als in dit adres het byte #D (0D) staat dan wordt na iedere regel in plaatst van alleen een #A, ook een #D naar de printerdriver gestuurd.
BUFFER	In BUFFER moeten de namen van de files worden geplaatst die worden weggeschreven of ingelezen met de routine FILE. Voordat er iets in BUFFER wordt gezet kan men het beste de buffer eerst leeg maken door de routine CLRBUF aan te roepen.
USERSP	Bij de opdrachten 'Go' en 'Single step' staat op dit adres de stack.
FREE	Begin van vrije geheugen ruimte.

Bruikbare routines in ZEMON

Functie

- RST10** Keert onder alle omstandigheden terug in de monitor. De Z80 registers kunnen met de 'Registers'-opdracht worden bekeken.
- NMI** Als RST10. Als de NMI aansluiting van de Z80 laag wordt, voert de Z80 een Call #66 uit, waarmee in ieder programma terugkeert in ZEMON's monitor. Als op de NMI ingang een schakelaar wordt aangesloten heeft men een altijd werkende machinetaalbreaktoets. Als deze schakelaar elektronisch ontdenderd is, heeft PC in 'Registers' de waarde van de programcounter op het moment dat het programma werd onderbroken.
- Drukt het teken in het A-register af op het scherm op de huidige printpositie (zie LNAD en X.POS in de systeem variabelen). De printroutine kent enkele speciale besturingscodes:
A=8 : Verplaatst de printpositie 1 positie naar links, en gaat eventueel terug naar de vorige regel.
A=#C: CLS. Wist het scherm en plaatst de printpositie links-boven.
A=#D: CRLF. Plaatst de printpositie vooraan de volgende regel.
- GET** Toets routine. Wacht op een toetsaanslag en keert dan terug met de ASCII waarde van de toets in het A-register.
- CKEY** Als GET, plaatst echter een cursor op de huidige printpositie zolang er op een toets wordt gewacht.
- ERR\$HL** Drukt de foutmelding waarvan in het A-register de foutcode zit af op het scherm.
- Drukt "? (Y/N) " af op het scherm en wacht daarna op een toets. De routine keert pas terug als de toets 'Y' ("Yes"), 'N' ("No") of 'STOP' ("No") is. Als "Yes" is geantwoord keert de routine terug met de carryflag geSET (C), anders met de carryflag geRESET (NC).
- PROUT** Printer driver. Stuurt het teken in het A-register naar de printer. Een teken #D wordt als een #A, of als #A en #D naar de printer gestuurd (zie CRPROUT in systeem variabelen).
- Stuurt een regel van het scherm vanaf adres HL naar de printerdriver.
- Print het getal in HL in hexadecimaal op het scherm.
- Print het getal in HL in decimaal op het scherm.
- Raad eens?
- MSG** Drukt de string achter Call MSG af op het scherm:
- ```

Call MSG
Defb "Hallo fans,"
Defb #D
Defb "ik ben ZEMON."^

```
- Dit drukt op het scherm af: Hallo fans,  
ik ben ZEMON.
- Het laatste teken van de string moet bit 7 geSET hebben (Zoals hier gedaan met '^'). De uitvoering van het programma wordt voortgezet achter de string. In de string mogen daarom geen tekens voorkomen waarvan bit 7 geSET is. Het 11de gebod: Vergeet Nooit Het Bit 7 Van Het Laatste Tekens, Of Er Gebeuren Vreselijke Dingen!
- DSKTOGGLE** Schakelt om van disk naar tape of andersom.

## ZEMON's file system

ZEMON's file system (de routines om files te schrijven of te lezen en de daar aan aanverwante taken) is zeer geavanceerd, menig operating system kan er jaloers op zijn! Met 1 routine kan ZEMON alle functies van de cassette, disk, of de seriele interface verrichten! Dit heeft als voordeel dat iemand die nog nooit met diskdrives heeft gewerkt, toch programma's kan schrijven die met diskdrives werken. Een programma dat deze routine gebruikt kan zonder moeite de cassette, de disk, en de seriele interface gebruiken!

## Functie

Voert een operatie uit op disk, tape of RS232. De functie-code in het A-register bepaalt welke operatie uitgevoerd wordt. De verschillende operaties hebben ook verschillende parameters nodig. De registers die de parameters bevatten zijn echter steeds dezelfde.

Bekrachtigd een eventueel aanwezige terugspoelautomaat als er met cassette is gewerkt en schakeld de motor van de diskdrives uit als er met disk is gewerkt. Moet worden aangeroepen ter afsluiting van een of meer file-operaties.

## Functiecodes bij FILE:

- FS.LOAD** Leest een file met lengte BC in het geheugen op adres HL.  
Entry: BC = lengte file  
HL = laadadres
- FS.SAVE** Schrijft een file weg met naam in BUFFER en lengte BC vanaf adres HL. DE wijst naar een vierletterige extentie (type aanduiding, bijv. ASMT).  
Entry: BC = lengte file  
DE = pointer naar extentie  
HL = startadres  
Naam van maximaal 34 tekens in BUFFER
- FS.SEP** (Search program) Zoekt een programma met naam in BUFFER en de extentie waar DE naar wijst op disk, cassette of RS232. Deze routine moet altijd worden aangeroepen voordat een file wordt gelezen met FS.LOAD of FS.BLOK.  
Entry: DE = pointer naar extentie  
Naam van maximaal 34 tekens in BUFFER  
Exit: BC = lengte van de gevonden file  
HL = laadadres van de gevonden file
- FS.BLOK** Leest een blok van maximaal 1K (#400=1024 bytes). Alleen het laatste blok dat van een file wordt gelezen mag korter zijn dan #400 bytes.  
Entry: BC = lengte van blok dat wordt gelezen, maximaal #400 bytes.  
HL = laadadres
- FS.DEL** Wist de file met naam in BUFFER (alleen voor disk en cassette). Op cassette worden ook alle files die achter de genoemde file staan gewist.  
Entry: Naam van maximaal 34 tekens in BUFFER. Bij disk gebruik moet ook de extentie erbij worden getypt.
- FS.FOR** Formatteert een schijf (alleen disk). De schijf krijgt de naam in BUFFER.  
Entry: Naam van maximaal 34 tekens in BUFFER.
- FS.DIR** Toont de directory (inhoud) van een schijf of cassette.

Alle operaties geven in het A-register de foutcode terug. Is dit nul, dan is de operatie succesvol verlopen. De foutmeldingen kunnen op het scherm worden afgedrukt met de routine ERR\$HL.

Bij alle operaties moet vooraan de naam (op adres BUFFER en BUFFER+1) staan op welk medium het betrekking heeft. Staat er R: dan wordt de operatie uitgevoerd op de RS232, staat er 1:, 2:, 3: of 4: dan wordt de operatie op de drive met dat nummer uitgevoerd.

# ;ZEMON system calls and variables version 1.3

|           |           |                                           |
|-----------|-----------|-------------------------------------------|
| RST10     | Equ #0010 | ;Breakpoint                               |
| NMI       | Equ #0066 | ;NMI vector                               |
| MSG       | Equ #1008 | ;Print message                            |
| PRINT     | Equ #104A | ;Print routine                            |
| GET       | Equ #104D | ;Key routine                              |
| FILE      | Equ #1050 | ;File system (Disk/Tape/Serial-interface) |
| FILEND    | Equ #1053 | ;Terminate file-action                    |
| ERR\$HL   | Equ #1108 | ;Print error A                            |
| CLRBUF    | Equ #1224 | ;Clear buffer (80 chars)                  |
| CKEY      | Equ #151E | ;Cursor-key-routine                       |
| YESNO     | Equ #183D | ;Ask Yes/No                               |
| LPR.LN    | Equ #1F17 | ;1 line to printer                        |
| DSKTOGGLE | Equ #2CF3 | ;Toggle disk/tape                         |
| PROUT     | Equ #3023 | ;Send A to printer                        |
| PBIN      | Equ #3A5B | ;Print number HL binair                   |
| PDEC      | Equ #3A6B | ; decimal                                 |
| PHEX      | Equ #3A70 | ; hexadecimal                             |

## ;Functioncodes for FILES and FIL.ERR

|          |           |                               |
|----------|-----------|-------------------------------|
| FS.LOAD  | Equ 0     | ;Load file (HL,BC)            |
| FS.SAVE  | Equ 1     | ;Save file (HL,DE,BC,<name>)  |
| FS.SEP   | Equ 2     | ;Search prog. (DE,<name>)     |
| FS.BLOCK | Equ 3     | ;Load block 1K (HL,BC#)       |
| FS.DEL   | Equ 4     | ;Delete file (DE,<name>)      |
| FS.FOR   | Equ 5     | ;Format medium (<name>)       |
| FS.DIR   | Equ 6     | ;Directory (<medium>)         |
| FS.BACK  | Equ 7     | ;Block back, only tape        |
| LNAD     | Equ #60B0 | ;Line address printroutine    |
| TABTAB   | Equ #60B4 | ;Tabulation table             |
| RAMTOP   | Equ #60BA | ;Top of free RAM              |
| SOT      | Equ #60EC | ;Start of text                |
| SOB      | Equ #60F0 | ;Start of textbuffer          |
| EOB      | Equ #60F2 | ;End of textbuffer            |
| SOL      | Equ #60F4 | ;Start of labels              |
| EOL      | Equ #60F6 | ;End of labels                |
| X.POS    | Equ #60FA | ;X-position printroutine      |
| X.MAX    | Equ #60FF | ;Maximum X value (39 or 79)   |
| LABLEN   | Equ #6109 | ;Significant label length     |
| PROVEC   | Equ #616E | ;Jumpvector printroutine      |
| INITBUF  | Equ #6170 | ;Printerinitialisation buffer |
| CRPROUT  | Equ #617A | ;Newline: LF or LF+CR         |
| BUFFER   | Equ #6200 | ;Buffer for names etc.        |
| USERSP   | Equ #62B0 | ;User stackpointer            |
| FREE     | Equ #6300 | ;Start of usermemory          |

Voorbeelden voor ZEMON's assembler versie 1.2, 15-9-'85

```

0000 6 ;#####
0000 7 ;#
0000 8 ;# VOORBEELDEN voor ZEMON's assembler versie 1.2
0000 9 ;#
0000 10 ;# Assembleren met <START> op het numerieke toetsenbord, daarna met een
0000 11 ;# toets (niet 'A') terug naar monitor.
0000 12 ;#
0000 13 ;# Dan kan het programma worden uitgevoerd met:
0000 14 ;# >Go destination: #9000 ('6'-toets)
0000 15 ;# of met:
0000 16 ;# >Go destination: START
0000 17 ;#
0000 18 ;# De routine laat een decimaal getal intypen, daarna wordt dit getal in een
0000 19 ;# willekeurige basis afgedrukt.
0000 20 ;#
0000 21 ;# File started 15-7-'85, last update 15-9-'85.
0000 22 ;#
0000 23 ;#####
0000 24
0000 25 Org #9000 ;Adres waar het programma wordt geplaatst.
0000 26
104A ==== 27 PRINT Equ #104A ;Adressen uit bijlage
151E ==== 28 CKEY Equ #151E ;
0000 29
0000 211000 30 START Ld HL,16 ;Getallen afdrukken in hexadecimaal
0003 221E91 31 Ld (BASIS),HL
0006 32
0006 33 ; Voor een andere basis moet de Ld HL,16 hierboven vervangen worden door
0006 34 ; Ld HL,<n> waarbij <n> de nieuwe basis is. (n=2 is binair)
0006 35
0006 CDE190 36 LOOP Call MESPRT
0009 0D 37 Defb #D
000A 47656566 38 Defm "Geef een getal"
001B CDA690 39 Call LINEDIT
001B CD5A90 40 Call EVAL.DEC
001E CD2390 41 Call PR.NUM
0021 18E3 42 Jr LOOP
0023 43
0023 44 ;PR.NUM: Drukt het nummer in HL af op het scherm in de numerieke basis
0023 45 ; (hexadecimaal, decimaal, octaal enz.) die in BASIS staat.
0023 46
0023 ED5B1E91 47 PR.NUM Ld DE,(BASIS) ;Numerieke basis in DE
0027 CD3D90 48 Call DIVIDE ;Deel het getal door de basis,
002A C5 49 Push BC ;Bewaar rest, dit is een digit.
002B 7C 50 Ld A,H ;Is quotient al nul?
002C 85 51 Or L
002D C42390 52 Call NZ,PR.NUM ;Nee, deel nog eens (recursie).
0030 C1 53 Pop BC ;Haal rest van stack.
0031 79 54 Ld A,C ;A is een digit
0032 55
0032 56 ;DIGIT vertaalt een getal in A naar zijn ASCII waarde, 0 wordt bijv. #30,
0032 57 ; 7 wordt #37, 15 (#F) wordt #46 ("F"). Daarna wordt het digit geprint.
0032 58
0032 C630 59 DIGIT Add "0"
0034 FE3A 60 Cp #3A
0036 3802 61 Jr C,NADD07
0038 C607 62 Add 7
003A C34A10 63 NADD07 Jp PRINT
003D 64
003D 65 ;DIVIDE deelt het getal in HL door dat in DE. Het quotient komt in HL, de rest

```

## Voorbeelden voor ZEMON's assembler versie 1.2, 15-9-'85

```

903D 66 ; in BC. Er wordt niet getest op delen door nul e.d.
903D 67
903D 3E10 68 DIVIDE Ld A,16 ;Het aantal bits van de getallen.
903F 010000 69 Ld BC,0 ;Rest wordt nul.
9042 70
9042 E5 71 UDIVLP Push HL ;Bewaar het deeltal.
9043 29 72 Add HL,HL ;Schuif bovenste bit uit deeltal.
9044 CB11 73 Rl C ;Schuif het bit in de rest.
9046 CB10 74 Rl B ;
9048 69 75 Ld L,C ;Stop de rest in HL voor de SBC.
9049 60 76 Ld H,B ;
904A A7 77 And A ;Zorg voor NC voor de SBC instructie.
904B ED52 78 Sbc HL,DE ;Is de rest groter dan de deler?
904D 3001 79 Jr NC,ISGOED ;Ja, dan is alles goed.
904F 19 80 Add HL,DE ;Nee, herstel rest.
9050 44 81 ISGOED Ld B,H ;Zet de rest terug in BC.
9051 4D 82 Ld C,L ;
9052 3F 83 Ccf ;Als NC dan kon er gedeeld worden.
9053 E1 84 Pop HL ;Herstel deeltal.
9054 ED6A 85 Adc HL,HL ;Schuif bit in het antwoord.
9056 3D 86 Dec A ;Alle bits gehad?
9057 20E9 87 Jr NZ,UDIVLP ;Nee, nogeens.
9059 C9 88 Ret ;
905A 89
905A 90 ;EVAL.DEC evalueert het decimale getal in de buffer, tot dat een onbekend teken
905A 91 ; wordt gevonden. Het resulterende nummer staat dan in het HL-register.
905A 92
905A 01F590 93 EVAL.DEC Ld BC,BUFFER ;Hier staat het getal.
905D 210000 94 Ld HL,0
9060 CD9F90 95 Call CHAR ;Haal 1 teken op in A, sla spaties over.
9063 FE30 96 Cp "0"
9065 3B1B 97 Jr C,BADNMBR
9067 FE3A 98 Cp "9"+1 ;Tekens tussen "0" en "9"?
9069 3017 99 Jr NC,BADNMBR ;Nee, verkeerd getal!
906B D630 100 EVD.LOOP Sub "0" ;Getal nu 0..9
906D 5D 101 Ld E,L
906E 54 102 Ld D,H ;Getal in HL nu in DE.
906F 29 103 Add HL,HL ;Getal * 2
9070 29 104 Add HL,HL ; * 4
9071 19 105 Add HL,DE ; * 5
9072 29 106 Add HL,HL ; * 10
9073 5F 107 Ld E,A
9074 1600 108 Ld D,0 ;DE wordt A.
9076 19 109 Add HL,DE ;Tel nieuwe digit op bij HL.
9077 CD9F90 110 Call CHAR ;Haal volgende teken.
907A FE30 111 Cp "0"
907C D8 112 Ret C
907D FE3A 113 Cp "9"+1
907F D0 114 Ret NC ;Laatste digit?
9080 18E9 115 Jr EVD.LOOP ;Nee, volgende.
9082 116
9082 CDE190 117 BADNMBR Call MESPRT ;Druk af: Verkeerd getal ingetypt!
9085 5665726B 118 Defb "Verkeerd getal ingetypt!"
909D BD 119 Defb #BD
909E D7 120 Rst #10 ;Breakpoint, keert terug in monitor.
909F 121
909F 122 ;CHAR haalt een teken uit (BC) en slaat spaties over.
909F 123
909F 0A 124 CHAR Ld A,(BC) ;Haal teken in A.
90A0 03 125 Inc BC ;Verhoog pointer.

```

## Voorbeelden voor ZENON's assembler versie 1.2, 15-9-'85

|             |     |                                                                                  |                                           |
|-------------|-----|----------------------------------------------------------------------------------|-------------------------------------------|
| 90A1 FE20   | 126 | Cp " "                                                                           | ;Is het teken een spatie?                 |
| 90A3 28FA   | 127 | Jr Z,CHAR                                                                        | ;Ja, dan volgende teken ophalen.          |
| 90A5 C9     | 128 | Ret                                                                              |                                           |
| 90A6        | 129 |                                                                                  |                                           |
| 90A6        | 130 | ;LINEDIT laat een regel van maximaal BUFLen tekens intypen en zet deze op BUFFER |                                           |
| 90A6        | 131 | ; De regel wordt afgesloten met <RETURN>. Het byte #D geeft het einde van        |                                           |
| 90A6        | 132 | de string in de buffer aan.                                                      |                                           |
| 90A6        | 133 |                                                                                  |                                           |
| 90A6 21F590 | 134 | Ld HL,BUFFER                                                                     | ;Op dit adres komt de string te staan.    |
| 90A9 112800 | 135 | Ld DE,BUFLen                                                                     | ;E=maximale lengte, D=0.                  |
| 90AC CDE190 | 136 | Call MESPRT                                                                      |                                           |
| 90AF 3AA0   | 137 | Defb " : ^                                                                       | ;Druk : af.                               |
| 90B1        | 138 |                                                                                  |                                           |
| 90B1 CD1E15 | 139 | Call CKEY                                                                        | ;Wacht op een toets en plaats cursor.     |
| 90B4 FE0D   | 140 | Cp #D                                                                            | ;Is het de <RETURN> toets?                |
| 90B6 2004   | 141 | Jr NZ,LN.NRET                                                                    |                                           |
| 90B8 77     | 142 | Ld (HL),A                                                                        | ;Ja, sluit de string af met #D.           |
| 90B9 C34A10 | 143 | Jp PRINT                                                                         | ;Print CRLF en keer terug uit LINEDIT     |
| 90BC        | 144 |                                                                                  |                                           |
| 90BC 01B190 | 145 | Ld BC,LNEDLOOP                                                                   | ;Laad het loopadres in BC.                |
| 90BF C5     | 146 | Push BC                                                                          | ;Zet het op de stack als return adres     |
| 90C0 FE0B   | 147 | Cp B                                                                             | ;Delete toets ingedrukt?                  |
| 90C2 200C   | 148 | Jr NZ,LN.NWIST                                                                   | ;Nee, spring                              |
| 90C4 7A     | 149 | Ld A,D                                                                           | ;Ja, wis 1 teken:                         |
| 90C5 A7     | 150 | And A                                                                            | ;Als D nul is kan er niets weg.           |
| 90C6 CB     | 151 | Ret Z                                                                            |                                           |
| 90C7 2B     | 152 | Dec HL                                                                           | ;Zet pointer 1 teken terug.               |
| 90C8 15     | 153 | Dec D                                                                            | ;Zet teller 1 teken terug.                |
| 90C9 CDE190 | 154 | Call MESPRT                                                                      |                                           |
| 90CC 0B20B8 | 155 | Defb B," ",B+17                                                                  | ;Haal het teken van het scherm weg.       |
| 90CF C9     | 156 | Ret                                                                              |                                           |
| 90D0        | 157 |                                                                                  |                                           |
| 90D0 FE20   | 158 | Cp " "                                                                           | ;Is de waarde van het teken boven spatie? |
| 90D2 DB     | 159 | Ret C                                                                            | ;Nee, volgende toets.                     |
| 90D3 FE80   | 160 | Cp #80                                                                           | ;Is de waarde onder #80?                  |
| 90D5 D0     | 161 | Ret NC                                                                           | ;Nee, volgende toets.                     |
| 90D6 4F     | 162 | Ld C,A                                                                           | ;Bewaar het teken in C register.          |
| 90D7 7A     | 163 | Ld A,D                                                                           | ;Aantal ingetypte tekens in A.            |
| 90D8 BB     | 164 | Cp E                                                                             | ;Mogen er nog tekens bij?                 |
| 90D9 D0     | 165 | Ret NC                                                                           | ;Nee, volgende toets.                     |
| 90DA 71     | 166 | Ld (HL),C                                                                        | ;Ja, plaats teken in buffer,              |
| 90DB 23     | 167 | Inc HL                                                                           | ;verhoog pointer voor volgende teken,     |
| 90DC 14     | 168 | Inc D                                                                            | ;verhoog teller van aantal tekens,        |
| 90DD 79     | 169 | Ld A,C                                                                           | ;teken terug in A,                        |
| 90DE C34A10 | 170 | Jp PRINT                                                                         | ;druk het teken af op het scherm.         |
| 90E1        | 171 |                                                                                  |                                           |
| 90E1        | 172 | ;MESPRT print de string achter de Call-instructie door het return adres te       |                                           |
| 90E1        | 173 | misbruiken. MESPRT keert terug nadat van een teken bit 7 geSET was, en           |                                           |
| 90E1        | 174 | dan het programma meteen achter dat teken. Deze routine is ongeveer              |                                           |
| 90E1        | 175 | hetzelfde als de routine MSG (zie bijlage) in de ZENON ROM-module.               |                                           |
| 90E1        | 176 |                                                                                  |                                           |
| 90E1 E3     | 177 | Ex (SP),HL                                                                       | ;Bewaar HL en haal return adres in HL,    |
| 90E2 F5     | 178 | Push AF                                                                          | ;bewaar AF.                               |
| 90E3 CDE990 | 179 | Call PR\$NUL                                                                     | ;Druk string vanaf HL af tot een 7e bit.  |
| 90E6 F1     | 180 | Pop AF                                                                           | ;Herstel AF,                              |
| 90E7 E3     | 181 | Ex (SP),HL                                                                       | ;Zet nieuw returnadres en herstel HL.     |
| 90EB C9     | 182 | Ret                                                                              |                                           |
| 90E9        | 183 |                                                                                  |                                           |
| 90E9 7E     | 184 | Ld A,(HL)                                                                        | ;Haal teken op in A.                      |
| 90EA 23     | 185 | Inc HL                                                                           | ;Verhoog string-pointer.                  |

Voorbeelden voor ZEMON's assembler versie 1.2, 15-9-'85

|             |            |              |                                            |
|-------------|------------|--------------|--------------------------------------------|
| 90EB CB7F   | 186        | Bit 7,A      | ;Laatste teken?                            |
| 90ED CBBF   | 187        | Res 7,A      | ;RESET bit 7 om laatste teken te printen.  |
| 90EF CD4A10 | 188        | Call PRINT   | ;Druk het teken af.                        |
| 90F2 2BF5   | 189        | Jr Z,PR\$NUL | ;Nog niet laatste? dan volgende teken.     |
| 90F4 C9     | 190        | Ret          |                                            |
| 90F5        | 191        |              |                                            |
| 002B ====   | 192 BUFLN  | Equ 40       | ;Maximaal aantal tekens in buffer.         |
| 90F5        | 193 BUFFER | Defs BUFLN+1 | ;Ruimte voor het aantal tekens + <RETURN>. |
| 911E 0000   | 194 BASIS  | Defw 0       | ;Adres voor opslag van getalbasis.         |
| 9120        | 195        |              |                                            |

# ;ZEMON system calls and variables version 1.4

|           |           |                                           |
|-----------|-----------|-------------------------------------------|
| RST10     | Equ #0010 | ;Breakpoint                               |
| NMI       | Equ #0066 | ;NMI vector                               |
| MSG       | Equ #1008 | ;Print message                            |
| PRINT     | Equ #104A | ;Print routine                            |
| GET       | Equ #104D | ;Key routine                              |
| FILE      | Equ #1050 | ;File system (Disk/Tape/Serial-interface) |
| FILEND    | Equ #1053 | ;Terminate file-action                    |
| ERR\$HL   | Equ #1108 | ;Print error A                            |
| CLRBUF    | Equ #1224 | ;Clear buffer (80 chars)                  |
| CKEY      | Equ #151E | ;Cursor-key-routine                       |
| YESNO     | Equ #183D | ;Ask Yes/No                               |
| LPR.LN    | Equ #1F17 | ;1 line to printer                        |
| DSKTOGGLE | Equ #2CF6 | ;Toggle disk/tape                         |
| PROUT     | Equ #302A | ;Send A to printer                        |
| PBIN      | Equ #3A62 | ;Print number HL binair                   |
| PDEC      | Equ #3A72 | ; decimal                                 |
| PHEX      | Equ #3A77 | ; hexadecimal                             |

## ;Functioncodes for FILES and FIL.ERR

|         |           |                                 |
|---------|-----------|---------------------------------|
| FS.LOAD | Equ 0     | ;Load file (HL,BC)              |
| FS.SAVE | Equ 1     | ;Save file (HL,DE,BC,<name>)    |
| FS.SEP  | Equ 2     | ;Search prog. (DE,<name>)       |
| FS.BLOK | Equ 3     | ;Load block 1K (HL,BC*)         |
| FS.DEL  | Equ 4     | ;Delete file (DE,<name>)        |
| FS.FOR  | Equ 5     | ;Format medium (<name>)         |
| FS.DIR  | Equ 6     | ;Directory (<medium>)           |
| FS.BACK | Equ 7     | ;Block back, only tape          |
| LNAD    | Equ #60B0 | ;Line address printroutine      |
| TABTAB  | Equ #60B4 | X ;Tabulation table             |
| RANTOP  | Equ #60BA | X ;Top of free RAM              |
| SOT     | Equ #60EC | ;Start of text                  |
| SQB     | Equ #60F0 | ;Start of textbuffer            |
| EOB     | Equ #60F2 | ;End of textbuffer              |
| SOL     | Equ #60F4 | ;Start of labels                |
| EOL     | Equ #60F6 | ;End of labels                  |
| X.POS   | Equ #60FA | ;X-position printroutine        |
| X.MAX   | Equ #60FF | ;Maximum X value (39 or 79)     |
| LABLEN  | Equ #6109 | X ;Significant label length     |
| PROVEC  | Equ #616E | ;Jumpvector printeroutine       |
| INITBUF | Equ #6170 | X ;Printerinitialisation buffer |
| CRPROUT | Equ #617A | ;Newline: LF or LF+CR           |
| BUFFER  | Equ #6200 | ;Buffer for names etc.          |
| USERSP  | Equ #62B0 | ;User stackpointer              |
| FREE    | Equ #6300 | ;Start of usermemory            |

BA4D = #6100