

REDAKTIE.

eindredakteur : Welmoed J. Jonker.
 ass. eindredakteur : Theo Huijgen.
 hardware-redakteur : Rob Borkent.
 software-redakteur : Kees van Duijvenbode.
 algemeen redakteur : Don Siahaya.

ABONNEREN.

U wordt abonnee op het ESGG-periodiek door het verschuldigde bedrag over te maken op postrekening 5368539 t.n.v. ESGG te Lopik, met vermelding 'abonnement periodiek'. Abonnementen gaan in per 1 juni van de lopende jaargang. Opzeggen: Uiterlijk vóór 1 maart van het lopende jaar!

ADMINISTRATIE ESGG-PERIODIEK.

Adreswijzigingen en klachten over de bezorging schriftelijk opgeven aan:
 Stichting ESGG
 Administratie ESGG periodiek
 Prins Hendrikstraat 3d
 3071 LG ROTTERDAM

KORRESPONDENTIE STICHTING ESGG.

de heer J.H.K.B. Nettelier
 sekretaris Stichting ESGG
 Prins Hendrikstraat 3d
 3071 LG ROTTERDAM

ADVERTENTIES.

Macro's: alleen voor bedrijven.
 acquireur: H. Herstel
 Mauritssingel 29
 3135 JM VLAARDINGEN.

Micro's: Alleen voor particulieren.

Formaat: een tekstregel is 66 tekens/spaties. per advertentie maximaal 6 regels.
 Prijs : elke twee regels tekst kost f. 3,00.
 Opgeven: per briefkaart aan de redactie; zet bij uw tekst het aantal regels van 66 tekens en uw postrekeningnummer.
 Betalen: gelijktijdig met het versturen van de briefkaart, door overmaking van het verschuldigde op postrekening 5368539 t.n.v. ESGG te Lopik, met vermelding 'micro's'.

Let op: Als de bijdrage van de betaling niet vóór de 25e dag van de even maand is ontvangen dan volgt geen plaatsing in dat nummer!

COPYRIGHT ESGG.

Het overnemen door abonnee's van in dit blad geplaatste artikelen, schema's of delen daarvan is toegestaan voor niet-commerciële doeleinden, mits met vermelding van de bron: ESGGetc.
 Het overnemen door derden (niet-abonnee's) is slechts toegestaan na verkregen schriftelijke toestemming van de ESGG-redactie. De redactie gaat ervan uit dat ingezonden kopij van de hand van de inzender is, tenzij uitdrukkelijk anders is vermeld.

SOFTWARE-VERZAMELAAR.

Stelt u door uzelf gemaakte, zg. public domain software, aan uw mede-leden beschikbaar? Zendt die dan aan:
 Voor opname op cassette: voor opname op disk:

Wim Warning Hermine Bakker
 Ganzebloem 16 Falklanddreef 18
 8265 MA KAMPEN 3363 AC UTRECHT

ESGG-SERVICE

De prijzen gelden vanwege de posttarieven uitsluitend in Nederland en België!

Bestellen: alleen per postgiro, op rekeningnummer 5368539 t.n.v. ESGG te Lopik; vermeldt: ESGG-service, en

vermeldt de naam en hoeveelheid van het gewenste artikel.

u ontvangt géén bevestiging van de order!

is het artikel niet (meer) leverbaar, dan ontvangt u WEL bericht!

Levering diskettes: Katalogus verkrijgbaar bij resp. CP/M-99 of ESGG-service.

Leverbare formaten 77 tracks hard- en softsectored, 40 en 30 tracks softsectored. De laatste twee formaten op resp. 2 en 3 schijven. Altijd levering van het genoemd aantal diskettes (eventueel onbeschreven).

Niet-ESGGleden en niet-abonnee's betalen per volume f. 10,- extra.

Garantie: De hardware van ESGG wordt gegarandeerd op juiste werking. Voor schade, ontstaan door onjuiste inbouw door anderen dan de technische medewerkers op Sorcerer Dagen, wordt geen aansprakelijkheid aanvaard!

Hieronder volgt een opgave van hetgeen thans verkrijgbaar is:

artikelnaam (prijzen per stuk!)	Sorcererdagprijs	per post
1. Software op cassette		
ESGG bibliotheek:		
nrs. 1 - 22	f. 7,50	f. 10,00
2. Software op diskette:		
CP/Mgg bibliotheek *) en		
ESGG diskettes 1 - 24 *)		
per volume:		
77 HS/SS		f. 25,00
40 SS		f. 30,00
30 SS		f. 40,00
*) zie: bestellen)		
3. Eprom Basic EXTension		
(versie 8) met beschrijving inbouw	f. 20,-	f. 25,00
4. Handleiding BEXT8	f.	f. 4,00
5. Invers video print		
(gebouwd)	f. 7,50	f. 12,50
6. Lichtpen ESGG	f. 35,00	f. 40,00
7. RANDISK uitbreiding	f. prijs 1986 ???	
8. Overzicht cass. software		
(delen: 1, p. deel)	f. 3,50	f. 4,50
9. Overzicht disk. software		
(delen: 1 - 2, p. deel)	f. 3,50	f. 4,50
10. Losse nummers van		
ESGG-periodiek	f. 3,50	f. 4,50
(zolang de voorraad strekt!)		
(jaargangen 1-3 alleen op diskette!)		

INPUT.

een rubriek voor het stellen van vragen en voor het geven van uw mening of commentaar.

Hebt u een probleem met systeem of programma, omschrijf dat zo volledig mogelijk en zendt het in een voldoende gefrankeerde omslag aan de redactie. Ons team zal trachten u een oplossing te geven. Wij behouden ons het recht voor probleem en oplossing in ons blad op te nemen.

MAATSTAF.

Wanneer de mens precies is begonnen vast te leggen welke methode en eenheid voor zaken als gewicht, lengte, inhoud en afstand etc. gebruikt zouden worden weet ik niet precies, maar al in de verre oudheid bestonden er vele systemen. Zo kende men o.m. maten als een el, een duim, een schepel. Deze waren min of meer volgens een vaste methode bepaald, maar een preciese eenheid is eigenlijk eerst later vastgesteld.

Thans bestaan er zeer nauwkeurige maatstaven voor de meter (in Parijs), voor de tijdseenheid 'seconde'. Maatstaven zijn vastgesteld om o.a. in het handelsverkeer voor uniformiteit te zorgen. Niet alleen in de handel, doch ook in de industrie en de wetenschap heeft men behoefte aan maatstaven. Toch blijven er nog wel zaken die zich niet onder een bestaande 'noemer' van één van de maatstaven laten vangen.

Met de voortschrijdende technieken is het mogelijk maatstaven steeds nauwkeuriger vast, of verder bij te stellen. De daarbij gehanteerde methodieken worden vervolmaakt; alles in een tijd die vroeger voor onmogelijk werd gehouden. De komst van microprocessoren maakte zulks mogelijk.

Een vreemde zaak is dan dat, waar handel en wetenschap veelal volgens gestandaardiseerde methoden en eenheden werken, dit in de wereld van de micro's niet of nauwelijks voorkomt. Binnen het CP/M gebeuren hanteren de merken hun eigen standaard voor het schijf-beschrijvingsformaat, terwijl ook de diameter van de schijf bij één en hetzelfde systeem kan variëren. Ook in de taal wil nog wel eens het een en ander verschillen. Daardoor is overdraagbaarheid bij CP/M systemen onderling in feite alleen mogelijk als van 8 inch (wel een universele standaard), of van een modem wordt gebruik gemaakt. Met de overdracht van programma's is men er echter nog niet. Vaak moet eerst het programma nog worden geconverteerd: het aanpassen van de instructies (bij Basic naar het dialect van uw systeem, of naar de juiste adressen van het ontvangende systeem) kan beginnen.

Bij onze getrouwe Exidy zijn de problemen van de uitwisselbaarheid van de schijven voor een flink aantal merken opgelost! Rob Borkent heeft namelijk een programma gemaakt dat de overdraagbaarheid mogelijk maakt van andere formaten naar dat van onze Exidy. Als bijzonder handigheidje: het kan allemaal binnen één (Exidy) systeem, vooropgesteld dat de drive die voor de overdracht gebruikt wordt, geschikt is voor 40 tracks (5").

Een volgend probleem is, zoals gezegd, de uniformiteit in de taal! Voor veel systemen is Basicode beschikbaar, maar deze 'middentaal' heeft in het algemeen niet die uitgebreide hoeveelheid instructies, die de Basic-versie van het eigen systeem heeft. Verder kent Basicode (versie 2) geen instructies voor het gebruik van diskdrives.

Een alternatief kan zijn een vertaal-programma, waarin alle bekende Basic-instructies zijn opgenomen. Dat betekent eigenlijk de vervaardiging van een databank, met voor elk merk een index die verwijst naar de instructie die in ons systeem daarvoor bestaat. Het programma moet in staat zijn de niet-bekende instructies van het ontvangen programma te verwisselen met de vergelijkbare instructie die de Exidy kent. Een aardig hulpmiddel zou verder kunnen zijn dat het programma zelf in staat is een DEF FN te genereren voor die instructies, die de Exidy niet kent. Daarmee zou zo'n vertaalprogramma dan zelfstandig de conversie kunnen realiseren. Omdat de gegevens een enorme ruimte vergen, is het eigenlijk vanzelfsprekend dat daarbij gebruik wordt gemaakt van de RAMDISK. Analoot aan zo'n programma voor Basic zou een programma voor het vertalen van machine- of andere taal kunnen worden gemaakt.

Als u zoiets een aardig idee vindt, probeer het dan eens uit te werken. Misschien dat u slaagt, waar anderen nog geen weg gevonden hebben. Dan kan u er aan mee helpen de Exidy zijn plaats tussen de andere computer-merken weer eens te laten bevestigen.

Welmoed Jonker.

I N F O I N F O I N F O I N F

- * Zeer binnenkort vindt er weer een Sorcerer Dag plaats (27 september!). In verband hiermede is aan de redactie gevraagd, de kopij eerder aan te leveren. Het is daardoor, en door het feit dat nog niet alle benodigde informatie binnen is, niet mogelijk het beloofde artikel over SUBMIT in dit nummer op te nemen. Zodra mogelijk zal publikatie volgen.
- * Nog even als geheugen-steun: De Sorcerer Dag vindt plaats in het zalencomplex Kunstmin aan de Boelekade 69 te Gouda. Bestuur en leden van de ESGG staan u daar graag te woord en tonen hun werkstukken. Hebt u ingeschreven voor een van de activiteiten in de technische hoek? Vergeet dan niet de daarvoor benodigde zaken (b.v. uw computer) mee te nemen!
- * Hebt u zich al opgegeven voor hulp aan de organisatie van de Sorcerer Dag? Als u bereid bent ergens aan mee te werken (b.v. de opbouw en het afbreken, of op andere wijze), geef dat dan even op aan onze sekretaris (Charles Netteler, 010-4330493).
- * In dit nummer beginnen we met twee series artikelen. Het ene is vooral gericht op gebruikers van een Exidy Sorcerer, die hun systeem zeer recent hebben aangeschaft en eventueel modificaties willen aanbrengen. In deze serie, van redakteur Kees van Duijvenbode, worden de door hem in zijn systeem aangebracht 'extra's' beschreven. De tweede serie gaat in op een van de meest gebruikte toepassingen van de micro, namelijk 'bestanden'. Aan de hand van het verbeterde ESGG bestandenprogramma zal worden getoond wat de mogelijkheden van een zg. elektronische kaartenbak zijn en hoe u eruit kunt halen wat er in zit.
- * De toegezegde verbeterde versie van het ESGG-bestanden programma kan op de komende Sorcerer Dag nog niet in de definitieve versie worden geleverd. Dat ligt in het feit dat er nog wordt gewerkt aan de routines die het mogelijk moeten maken de wijze van bedrukken van etiketten en van de uitdraai van de gegevens zodanig te maken dat vanuit elk door u gemaakt bestand de nodige informatie over de wijze van drukken wordt gegeven. Maar het staat eraan te komen!
- * Doordat er zo'n korte tijd tussen de Sorcerer Dag, de CP/M dag (Utrecht op 4 oktober) en de HCC-dagen op 21 en 22 november (ook Utrecht, maar de ESGG is er alleen op 22 november!) ligt, heeft het bestuur besloten NIET naar de CP/M gebruikersdag in Utrecht te gaan. U kunt op de Sorcerer Dag toch ook in het bezit van de gewenste schijven komen!
- * Een prettig bericht voor modem-bezitters. Van HCC-zijde is vernomen dat de FIDO-node Arnhem weer bereikbaar is, zij het onder een ander nummer. Dat betekent dat u binnenkort ook informatie over de ESGG en haar activiteiten op de 600 Ohm-lijn kan ontvangen. Ben Hooijenga, die zich voor de ESGG met de invoer heeft belast is nu druk bezig met het inbrengen van gegevens. Kijkt u maar eens of we er al zijn! Het nummer luidt als volgt: 085-23 33 77.
- * Naast alle opbeurende informatie, ook een droeve mededeling: Zoals u wellicht bekend is, wordt ons periodiek ook in het Engels vertaald en in een kleine oplage naar buitenlandse abonnee's verzonden. Het aantal betalende abonnee's op dit Engelse periodiek is thans echter zo klein geworden dat het niet langer rendabel is om de uitgifte ervan voort te zetten. Maandelijks steken verschillende hobbyisten zoveel tijd en moeite in deze aparte oplage, dat zij nauwelijks tijd hebben voor de hobby. Daarom, sorry, Engelsprekenden/lezenden, maar vanaf 1 juni 1987 zult u het -als dat gaat- met de Nederlandse uitgave moeten doen!

- * Weet u, dat u de redactie enorm blij kunt maken met een door u geschreven artikel? Alhoewel het in een enkel geval vrij lang heeft geduurd voor een ingezonden artikel werd opgenomen, zijn we nu vrijwel aan de bodem van de voorraadbak.
De leden van de redactie zijn, net als u hobbyisten, die -naast hun gewone werk en hun hobby- behoorlijk wat tijd steken in de vervaardiging van uw periodiek. Die door hen geïnvesteerde tijd begint nu echter zodanig te worden, dat de hobby eronder gaat lijden! Gun dók hen de tijd voor de hobby en help mee aan het vullen van ONS periodiek! U verwerft daarmee ook de dank van uw mede-leden.
- * Er is nog iemand die u blij kunt maken: onze eigen Hermine Bakker. Om gezondheidsredenen vraagt zij dringend of iemand de zorg voor Basicode van haar schouders wil nemen. Gevraagd wordt een enthousiasteling die zijn Basics door en door kent en met machinetaal kan werken en verder ook de coördinatie met de medewerkers kan regelen.
U staat er niet namelijk niet alleen voor! We ondersteunen vier versies te weten, Standard Basic, Standard Basic met BEXTB, MBasic en Exbasic. Als u wilt dat dit projekt niet voor de Exidy Sorcerer verloren gaat, kunt u zich melden bij Hermine, telefoon: 030-612405.
Er is een goede samenwerking met de Stichting Basicode en het projekt Basicode-3 bevindt zich thans in de afrondings-fase.
Wie o wie ?
- * Wist u, argeloze computer-gebruiker, dat probleemloos werken met cassettes nog niet hoeft te betekenen dat het overstappen naar diskdrives ook vlekkeloos gaat? Een Amsterdams lid dat dacht de oorzaak van het niet oproepbaar zijn van de B:drive in de apparatuur te moeten zoeken, bleek zijn mening te moeten herzien. De drives werkten vlekkeloos bij anderen (met de eigen computer). Hij onderzoekt nu of de hem via het lichtnet geleverde energie de verdachte is.
- * Zoveel Exidy's, zoveel mogelijkheden: Waar de 50-aderige kabel tussen computer en diskdrive-controller bij de ene Exidy maar zo'n 45 cm kan zijn, is hij bij de andere configuratie maar liefst 85 cm (!). Licht dat nu aan de Exidy (en de elektronische opbouw) of aan andere omstandigheden? Misschien dat de technuten onder u ons daarover eens wat meer zouden kunnen vertellen. We zijn benieuwd naar hetgeen u ons daarover toestuurt.

B L A D E N U I T A N D E R E B

- * Databus nr. 7/8: Thema van dit nummer is expert-systemen (7 delen). De bespreking van een IC waarmee het mogelijk is, reeksen bytes met elkaar te vergelijken, ook al gaat het om verminkte informatie. Verder de vaste rubrieken, zoals boekbesprekingen etc.
- * PCM nr. 7/8: Ook hier is het onderwerp 'expert-systemen' zij het al deel 2 (een praktijk-voorbeeld). Een artikel over elektronische post en wat daar zoal voor komt kijken. Computer-talen is het onderwerp van een ander artikel. Adventures: het spel-element in de computer. Resten nog de tests en andere vaste rubrieken.
PCM nr. 9 : In dit nummer informatie over een aantal modems en enkele, daarbij bruikbare Viditel-programma's. Als een exclusiviteit biedt men de lezer een 'onkraakbare' beveiliging voor uw software. Nu dus maar wachten op de hacker die meedeelt zo'n beveiligd programma te hebben gekraakt! Als u wat heeft aan aanvullende gegevens over videodisplays (beeldschermen), in dit nummer treft u daarover uitgebreide informatie aan. Verder de gebruikelijke tests van soft- en hardware.

SORCERER MAATWERK (1).

Door Kees van Duijvenbode.

Naar aanleiding van vragen die uit de enquête naar voren zijn gekomen, geef ik in dit artikel graag een aanzet tot wat misschien een serie kan worden over wat er zoal door leden aan hun Sorcerer versleuteld is. In dit eerste (en naar ik hoop niet laatste) deel wil ik beschrijven wat ik aan mijn computer heb gedaan en op welke wijze. Ik ben me ervan bewust dat het voor velen van u oude koek (en dus misschien weinig interessant) is, maar voor nieuwe(re) leden kan het misschien weer een duwtje in de rug zijn. De, in deze serie te beschrijven uitbreidingen stammen alle uit mijn beginperiode (op één na) en betreffen hardware-uitbreidingen.

De eerste uitbreiding die ik realiseerde, stamt uit de tijd dat ik nog met cassettes werkte en betreft een wijziging in de Sorcerer waardoor ik met een schakelaartje kon kiezen welke van de twee aangesloten recorders, voor wat betreft de motorsturing, default (dus 'nummer 1') moest zijn. Dit gaf, vond ik, het voordeel dat bij het kopiëren minder typewerk verricht hoefde te worden.

Voordat we aan het 'las- en snijwerk' gaan beginnen eerst nog even dit: Bij het beschrijven van de verbindingen ben ik uitgegaan van het bij mij aanwezige schema. In dit schema verwijst pen 1 van 12E-A naar een aansluiting IC6 op blad 6 en pen 1 van 12E-B naar IC9 op datzelfde blad. Het leek allemaal in orde dus....

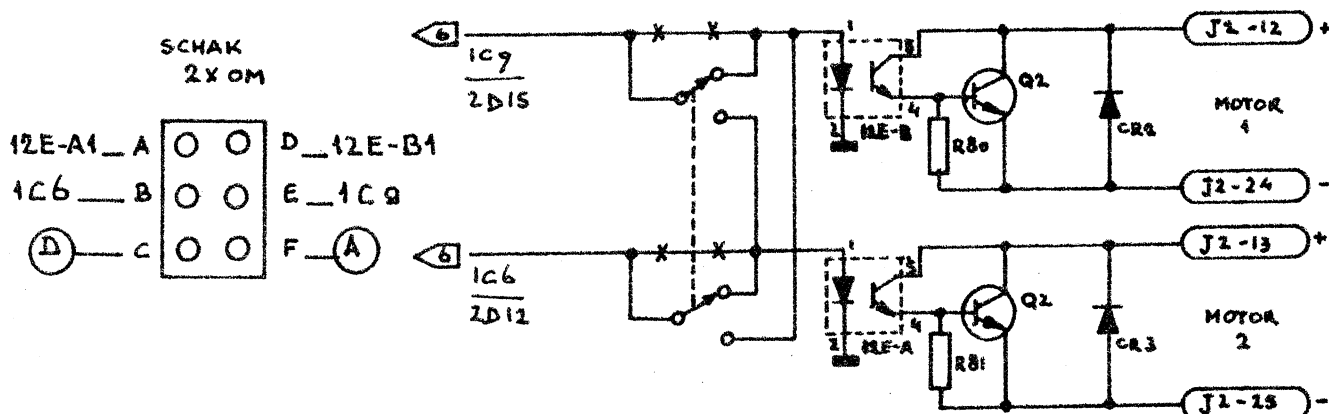
Ik keek dan ook vreemd op toen mij werd verteld dat bij het bekijken van het schema dat bij de redactie lag, deze pennen niet verwezen naar hetzelfde IC. Het IC is in beide gevallen hetzelfde, nl. 74LS374. In het ene schema heeft dat IC het nummer IC en in het andere 2D!

Om u houvast te bieden: 12E-A voert het signaal OOB5 en 12E-B het signaal OOB4. Voor de schema-kijkers: beide sets verwijzen wel naar het juiste blad, namelijk 6. Het is maar dat u het weet!

De wijziging geschiedt als volgt:

1. Knip met een fijn kniptangetje de verbinding van pen 1 van 12E-A met IC 1C pen 6 en van 12E-B die van pen 1 met IC9 door.
2. Verbindt de print-eilandjes van deze pennen met de middenkontakten van een dubbelpolige wisselschakelaar (tekening 2, punten B en E).
3. Verbindt pen 1 van 12E-A met één van de wisselkontakten (F) van de schakelaar en met het tegenoverliggende kontakt (A) van de andere helft van de schakelaar.

Als u het nog steeds kan volgen mag u pen 1 van 12E-B verbinden met de twee overgebleven kontakten (C en D) van de schakelaar.



Als u alles volgens plan hebt uitgevoerd, moet het mogelijk zijn, om met de schakelaar te kiezen van welke van de twee aangesloten recorders "default" de motor gaat lopen.

Uiteraard moeten de twee recorders wel via de seriële poort aangesloten worden, anders heeft u natuurlijk geen motorsturing.

Zo, dat was het voor deze keer. Een simpel begin, maar naar ik hoop toch voor een paar mensen interessant.

In volgende afleveringen hoop ik een hardwarematige video-inverse schakeling en een Basiccode-interface via de standaard cassette-interface omschakeling te behandelen.

Ik nodig hierbij andere knutselaars uit, ook eens wat over de door hen gepleegde (ver)bouwsels op papier te zetten. Denk niet te gauw dat dat voor anderen niet interessant is. Denk ook niet te gauw dat u niet kunt "schrijven". Zet maar gewoon uw uitgevoerde ideeën op papier dan maken wij indien nodig, er wel een leuk artikeltje van.

CP/M OPDRACHTEN.

De laatste tijd komen er erg veel tweede hands Exidy Sorcerer computers op de markt, welke afkomstig zijn van b.v. bedrijven die naar een moderner systeem overstappen. Uit vragen wordt afgeleid dat in veel gevallen geen handleidingen met de computer zijn geleverd. Voor beginnende computeraars is het ontbreken van een handleiding een handicap. Die wordt zwaarder als dan ook nog blijkt dat het moeilijk, zo niet onmogelijk is om aan die handleiding te komen. Daarom is het prettig dat onze redakteur Theo Huijgen in SCUA van mei '85 een artikel van G. Ohlenrott heeft gevonden, dat in die lacune kan voorzien.

Voor iedereen die nuttig en volledig van de CPM mogelijkheden gebruik wil maken, volgt hier een kort overzicht van de beschikbare opdrachten. Een deel van deze opdrachten leer je vrij gemakkelijk aan omdat je ze wel moet leren om de verschillende opstartprocedures te kunnen uitvoeren. De minder gebruikte opties kunnen gemakkelijk in het vergeetboek raken.

Er zijn drie types opdrachten te onderscheiden:

- de, als het ware, in het systeem ingebouwde.
- de slechts "tijdelijk" vanuit het geheugen oproepbare en
- de "Line Editing" opdrachten.

Wij zullen ze hierna alle drie beschrijven. Degenen die voor het eerst met CP/M gaan werken, moeten zich het volgende realiseren:

De eerste twee types zullen, na te zijn opgeroepen, slechts "werken" na het indrukken van de RETURN-toets <CR>, terwijl de Editor opdrachten al werken zodra ze zijn ingetoetst (indien ze juist gebruikt worden).

Voor de goede orde eerst nog even wat begrippen:

Bootstrap loader	de routine die zorgt voor het inladen van CP/M.
Default drive	de door het systeem geïntialiseerde drive.
Destination	de bestemming.
Logged(-in)	de actieve drive (schijf).
Source	de bron, b.v. een file.
Start-opdracht	er zijn enkele methoden, deze zijn afhankelijk van de componenten van uw systeem:
	- wanneer de zg. Van Montfort Monitor 1.3 is geïnstalleerd, start u altijd met CTRL-X.

	- wanneer u een hard-sectored systeem heeft, met het opdracht GO BC00 (CP/M 47K!), of GO DC00 (CP/M 55K).
	- heeft u een ongemodificeerd softsectored videodisk systeem, dan start u met GO BE00.
	- De overige softsectored systemen (40 of 77 tracks) starten allemaal met GO BF00 (48K) of GO DF00 (56K).
Sign-on message	de boodschap die na het starten van CP/M op het scherm verschijnt.
System disk	de schijf waarop uw 'systeem' staat, de 'moeder'-schijf, zoals die door de leverancier van de computer wordt geleverd.
TPA	Transient Program Area, het gebied dat door programma's kan worden gebruikt.
.FLN	Door de schijver gebruikt voor de aanduiding 'programma-naam, ook wel file-naam genoemd.
.TYP	Het type van de betreffende file, ook wel EXTEND genoemd.
:	het teken achter de drive-letter (moet!!).
=	vergelijkingsteken, ook gebruikt bij kopiëren.
>	de prompt; na dit teken kunt u CP/M opdrachten geven.
[]	optionele toevoeging.
*	wordt gebruikt om een zg. al dan niet gehele wildcard te geven. B.v. *.COM in een DIR-opdracht geeft een overzicht van alle COM-files.
?	wordt gebruikt om bij het zoeken naar files met bijna gelijke naam, alle aan de voorwaarde voldoende files te vinden, b.v. TEKST?.WPF vindt zowel TEKST1.WPF, als TEKST2.WPF.
<CR>	hiermee wordt bedoeld de RETURN-toets.

Line editing opdrachten.

CTRL-C	Warme start.
CTRL-E	Vervolg op de volgende regel. Wordt gebruikt als lange opdrachten worden geschreven.
CTRL/H of SHIFT/RUB	Wist het karakter waarop de cursor staat.
CTRL/J	Einde invoerregel (werkt zoals <CR>).
CTRL/M	Einde opdracht-invoer.
CTRL/P	Aktiveert/stopt de uitvoer naar de printer.
CRTL/R	Herhaalt de regel die wordt geschreven.
CRTL/S	Onderbreekt de uitvoering. Vervolgt na drukken op een willekeurige toets.
CRTL/U	Wist de regel die geschreven is, (voordat op <CR> is gedrukt!).
CRTL/X	Idem; bij Van Montfort Monitor 1.3 ook koude start.
SHIFT/RUB	Wist een karakter.

Ingebouwde opdrachten.

De opdrachten die hierna zijn beschreven, zijn zg. ingebouwde opdrachten, d.w.z. zij maken deel uit van CP/M, in tegenstelling tot opdrachten zoals PIP, STAT, SYSGEN, die op de betreffende schijf moeten staan om gebruikt te kunnen worden.

DIR [x:]	Toont op het scherm de directory-inhoud. X: betekent dat de directory van drive x: moet worden gelezen. Indien de drive-aanduiding ontbreekt dan wordt de directory van de actieve drive getoond.
----------	---

DIR [x:]FLN.TYP	Toont het indexbestand van drive x: waarvan de namen corresponderen met FLN.TYP.
DIR N*.BAS	Toont alle .BAS(IC) files waarvan de naam met 'N' begint.
ERA [x:]FLN.TYP	Verwijdert de naam van de opgegeven file FLN.TYP van drive x:. FLN en/of TYP kunnen dubbelzinnig zijn.
ERA [x:]*.*	Verwijdert alle files van de schijf in drive x: CP/M vraagt in dit geval expliciet om bevestiging van de gegeven wis-opdracht.
REN NFLN.TYP=OFLN.TYP	Zoekt in de directory de O(UD)FLN.TYP en verandert de naam in N(IEUW)FLN.TYP.
SAVE nn [x:]FLN.TYP	Schrijft een deel van de TPA van het geheugen (vanaf 100H gerekend) in de file FLN.TYP op drive x:. Hierbij is nn het decimale aantal 256 bytes grote pagina's in het geheugen.
TYPE [x:]FLN.TYP	Toont de inhoud van de file FLN.TYP op drive x: op de monitor, of na CTRL-P op de printer.
USER n	Zet het gebruikersnummer (USER NUMBER) op n. "n" is een geheel getal tussen 1 en 15.

stelsel generatie kommando's.

MOVCPM is een programma dat gebruikt wordt voor het aanmaken van een CP/M systeem in een gewenste geheugen-grootte. Met SYSGEN wordt de door MOVCPM vervaardigde versie op de systeem-sporen van de aangewezen schijf geplaatst.

MOVCPM	Maakt een nieuwe kopie van CP/M die gebruik maakt van het beschikbare geheugen, geeft de besturing over aan deze versie van CP/M, maar maakt geen file op schijf.
MOVCPM nn	Maakt een nieuwe kopie van CP/M die gebruik maakt van nn Kbytes, geeft de besturing over aan deze versie van CP/M, maar schrijft deze niet weg.
MOVCPM * *	Maakt een kopie van van CP/M die gebruik maakt van het totale beschikbare geheugen en bewaard deze kopie in het geheugen om desgevraagd naar de schijf te worden weggeschreven, of te worden gebruikt door SYSGEN voor het plaatsen op de systeem-sporen.
MOVCPM nn *	Idem ,doch met gebruik van nnK bytes. nn is een geheel decimaal getal vanaf 20 tot 64 voor CP/M versie 2.0 of latere versies.
SYSGEN	Laadt het "SYSGEN" programma om CP/M vanuit het geheugen op de systeem-sporen te plaatsen of om een kopie van de systeem-sporen van de aangegeven drive op die van een andere te plaatsen.

Assembleer opdrachten.

ASM.COM assembleert files in assembler (mnemonics) geschreven (ASCCI) files naar een formaat dat met het LOAD kommando kan worden verwerkt. ASM produceert een .HEX en een .PRN file. De laatste kan vrij omvangrijk

zijn, zodat het zinvol is een lege of vrijwel lege schijf te gaan gebruiken.

ASM FLN Assembleert de file FLN.ASM op de diskette die actief is (logged drive) voor de file.

ASM FLN.OPT Assembleert de files FLN.ASM op drive O: (A,B,C...P) Schrijft HEX file naar drive P: (A,B,C...P) of springt een drive verder als P:=X: Schrijft een PRN file naar drive T, zendt deze naar de terminal indien P:=X: of springt een drive verder als P:=Z:

DUMP [x:]FLN.TYP Toont de Hexcodes van de file FLN.TYP in drive x:. Indien FLN niet benaamd, is dan wordt de eerste file getoond die overeenkomt met FLN.TYP.

LOAD [x:]FLN Leest de file FLN.HEX en vormt deze om tot een file FLN.COM en schrijft die naar drive x:.

Programma-ontwikkeling met ED.COM.

ED is een soort tekstverwerker voor het vervaardigen van b.v. door SUBMIT.COM uit te voeren batch-files. In ED kunnen ook source-files worden vervaardigd, die met ASM.COM kunnen worden omgezet naar .COM files. Dit vergt van de gebruiker kennis van het programmeren in ASSEMBLER. Voor dit doel worden ook veelvuldig tekstverwerkers gebruikt.

ED [x:]FLN.TYP ED start en haalt daarna de opgegeven file van de aangegeven drive. Vindt ED geen file, dan wordt aangenomen dat de gebruiker een nieuwe file wil vervaardigen. In de hier gegeven structuur wordt op drive x: een file in de directory geschreven met de naam FLN.TYP. Als de file niet bestaat geeft ED als antwoord 'NEW FILE'.

ED opdrachten.

I Start aanmaak nieuwe file.

^Z Beëindigt het aanmaken van de nieuwe file.

nA Bij bestaande files wordt het aantal opgegeven regels ingelezen uit de in de opdracht achter ED gegeven file. Alleen A betekent dat maar één regel wordt ingelezen.

+/-nB Verplaatst de CP (character pointer) naar het begin van de buffer met +B, of het eind met -B.

+/-nC Verplaatst de CP vooruit (+) over n characters, of terug (-) in een regel. Voor <CR> rekent de CP twee tekens (RETURN en LINEFEED).

+/-nD Verwijdert vanaf hier (+) n tekens (m.i.v. CP) of terugtellend vanaf CP (-). Geen getal betekent 1 teken.

E Sluit de ED bewerkingen af en schrijft de bewerkte file naar de opgegeven schijf (automatisch) nadat van de oorspronkelijke file een .BAK file is gemaakt.

nFstring Zoekt n maal naar de opgegeven string. Als de string wordt afgesloten met <CR> dan kunnen geen andere ED kommando's worden toegevoegd. Moet dat, dan CTRL Z gebruiken.

H Werkt als E, maar komt daarna terug in ED met de file die net is weggeschreven.

+/-nK	Verwijdert n regels vooruit (+) m.i.v. CP-regel, of terug (-) vanaf de CP.
+/-nL	Verplaatst de CP naar het begin van regel n (vooruit of terug). Als n=0 dan naar het begin van de aktuele regel.
Q	Verlaat ED zonder iets aan de ingelezen file te doen.
+/-nT	Type de tekens vanaf de CP tot het eind van de regel. Als n=0 dan de huidige regel tot aan de CP. (+) is van af en (-) is terug vanaf. B#T toont alle regels in de buffer.
+/-U	Vertaalt alle ASCII input naar UPPER (+) of schakelt de vertaling af (-).
n:	Verplaatst de CP naar het begin van regel n.

Programma aanpassing met DDT.

Het Dynamic Debugging Tool (DDT) is een krachtig programma voor het aanbrengen van wijzigingen en aanvullingen in bestaande programma's. Van dit soort programma zijn er vele. Een van de nieuwere is SID.COM, dat een aantrekkelijker lay-out op het beeldscherm heeft.

DDT	Laadt het programma DDT en wacht op verdere (DDT) instructies.
DDT B:FLN.TYP	Verplaatst het DDT programma hoger in het geheugen, laadt dan de FLN.TYP van drive B: vanaf adres 100H.
DDT opdrachten.	
ASsss	Laadt de assemblercode instructies beginnend bij adres ssssH.
D	Toont op het scherm de inhoud van de volgende 192 geheugenadressen.
Dsss,ffff	Toont op het scherm de inhoud van de geheugenadressen ssssH tot ffffH.
Fsss,ffff,cc	Schrijft cch (in de 8-bit Code) naar de adressen ssssH tot ffffH.
G	Begint met uitvoeren vanaf het adres dat in de programmateller staat.
Gsss,bbbb	Plaats een breakpoint bij bbbbH en gaat met de uitvoering verder vanaf ssssH.
Gsss,bbbb,cccc	Plaats breakpoints bij bbbbH en ccccH en gaat met de uitvoering verder bij ssssH.
Gbbbb	Plaats een breakpoint bij bbbbH en gaat met de uitvoering verder op het adres bepaald door de inhoud van de programmateller.
L	Disassembleert de volgende 11 regels van het Hex programma.
Lsss	Disassembleert de volgende 11 regels van het programma vanaf adres ssssH.
Lsss,ffff	Disassembleert vanaf ffffH tot ssssH.

Mssss,ffff,dddd	Verplaatst de inhoud van geheugenblok ffffH tot sssH naar een ander geheugengedeelte te beginnen bij adres ddddH.
IFLN.TYP	Aktiveert het control-blok "Default File" met de naam FLN.TYP.
R	Leest een file in vanaf de diskette (eerst I kommando gebruiken).
Rnnnn	Leest een file van de schijf in het geheugen met beginadres nnnnH hoger dan normaal voorzien (100H +nnnnH).
Sssss	Toont de inhoud van adres sssH op het scherm en laat de mogelijkheid om deze inhoud te wijzigen.
Tnnnn	Doorloopt de uitvoering van (Hex) nnnn programma-instructies.
Unnnn	Voert nnnnH programma-instructies uit. Onderbreekt de uitvoering en toont de inhoud van CPU-registers.
X	Toont de inhoud van de CPU registers.
Xr	Toont de inhoud van het CPU register -R- en wijzigt dit, indien gewenst.

Periferie omwisseling.

PIP (peripheral interchange program) is een programma voor overzetting van files of informatie. Dit kan zowel tussen drive en computer, als tussen logische en/of fysieke toestellen. Hierbij moet niet alleen worden gedacht aan drives, maar ook aan beeldschermen, printers en eventueel andere apparaten worden gedacht. Het verkeer kan zowel van, als naar een dergelijk toestel geschieden. Met behulp van STAT.COM kunnen referenties aan periferie-toestellen worden toegekend.

Omdat in het vorige nummer al uitgebreid is ingegaan op de kopieer-mogelijkheden tussen drives van PIP.COM, wordt dit hier niet verder behandeld.

PIP opdrachten.

[x:]FLN.TYP=DEV:[p] Kopieert DATA van apparaat DEV: naar de file FLN.TYP in drive x:
 DEV:=[x:]FLN.TYP[p] Kopieert DATA van de file FLN.TYP van drive x: naar apparaat DEV:
 DST:=SRC:[p] Kopieert DATA van apparaat SRC naar apparaat DST:.

PIP parameters.

B Bepaalt BLOCK MODE overdracht.
 Dn Wist alle tekens na de n-de kolom op een regel.
 E Toont wat gekopieerd wordt op het scherm.
 F Wist de in een bestand aangebrachte pagina-indeling.
 Gn Haalt file van het USER-gebied n.

H	Kontroleert de juiste Intel Hex-formaten, verwijdert overbodige tekens en drukt eventuele fouten op de console af.
I	Houdt geen rekening met ":00" records bij het overschrijven van Intel HEX files.
L	Zet hoofdletters in kleine letters om.
N	Voegt regelnummer toe aan iedere gekopieerde regel.
O	Kopieert Object-files (houdt geen rekening met EOF markeringen).
Pn	Eindigt na iedere n regels met een opdracht voor een nieuwe pagina.
R	Laat het kopiëren van systeem-files toe.
Ss Z	Specificeert de start om te kopiëren na het ontmoeten van de string "s".
Tn	Tabuleert elke n-de kolom tijdens de overdracht.
U	Zet kleine letters om in hoofdletters.
V	Vergelijkt bron en kopie op juistheid.
W	Overschrijft Read/Only files, als de te kopiëren file dezelfde naam heeft.
Z	Zet de parity bit van ASCII karakters op 0.

PIP bestemming's benamingen.

CON:	PRN:	LST:	Kommando's naar een logisch apparaat TTY:
PTP:	LPT:		
CRT:	UP1:	UL1:	Kommando's naar fysieke apparaten
UC1:	UP2:		
	OUT:	PRN:	Speciale PIP apparaten.

PIP bron benamingen.

CON:	RDR:		Ontvangt logische signalen.
TTY:	PRT:		
CRT:	UR1:		Ontvangt uit fysieke apparatuur UC1:
UR2:			
NUL:	EOF:	INP:	Ontvangt uit speciale PIP apparaten.

Statistiek.

STAT	Toont nog vrijblijvende geheugenruimten op elke aangesloten drives, na de laatste koude of warme start.
STAT [x:]	Toont de nog vrije geheugenruimte op drive x:
STAT [x:]FLN.TYP	Toont alle kenmerken van de FLN.TYP, die zich op drive x: bevindt. De file aanduiding mag algemeen zijn bv. *.TYP of FLN.* of *.*.
STAT DEV:	Definieert de fysieke apparaten.
STAT VAL:	Toont de mogelijke toekenningen van de apparatuur en een overzicht van alle stat opdrachten.
STAT log:=fys:	Toewijzing veranderen van de koppeling logische namen met fysieke apparatuur.
STAT USR:	Toont de aktuele USR-nummer alsmede alle files met het nummer van de USR voor wie ze toegankelijk zijn.
STAT [x:]dsk:	Toont alle kenmerken van diskette in drive x:
STAT [x:]R/O	Kent een tijdelijke bescherming tegen overschrijving toe aan drive x:. Met de toevoeging van de file-naam kan een enkele file beschermd worden. De opdracht wordt dan: STAT (x:) FLN \$R/O. Deze opdracht is ook visa versa te gebruiken: STAT (x:) FLN \$R/W.

Batch procedures.

Het programma SUBMIT.COM heeft bij CP/M een zelfde functie als de BA(tch) opdracht in Exidy's monitor. Het stelt de gebruiker in staat een aantal opeenvolgende handelingen automatisch te laten uitvoeren.

SUBMIT FLN par

Maakt een file \$\$\$SUB met de opdrachten die zijn opgenomen in FLN.SUB. Desgewenst kunnen delen van de opdrachtregels in FLN.SUB worden vervangen door parameters ("par") tijdens het opbouwen van \$\$\$SUB. CP/M haalt de opdrachten bij voorkeur uit deze file en niet van het toetsenbord.

In een van de komende nummers zal nader op de mogelijkheden van SUBMIT worden ingegaan.

Diversen.

x:

De zg. drive-specificatie. Verandert de actieve disk naar x:. X kan een letter zijn van A tot P. Bij ontbreken van x: in een opdracht wordt als drive de reeds actieve drive verondersteld.

verboden tekens

Er zijn een aantal tekens die, vanwege hun bijzondere functie in CP/M, niet in namen van files mogen voorkomen. Dit zijn:

< > . , : = ; * ? []

Ook mag in de naam geen spatie voorkomen.

file-typen

Er zijn vaste afspraken gemaakt over de geldige file-typen die in CP/M mogen worden gebruikt. Deze zijn:

COM: files van dit type zijn COMMAND files. Deze worden opgeroepen door het intikken van de naam (zonder het type!).

ASM: files in zg. source, kunnen door ASM.COM worden verwerkt tot .HEX en .PRN files.

PRN: Zie bij ASM

HEX: Zie bij ASM; deze files kunnen door LOAD.COM naar .COM files worden vertaald.

BAS: Files in Basic vervaardigd (M- of EXBASIC).

INT: Intermediate file, meestal in gebruik bij in BASIC geschreven bestand-programma's. Het bestand zelf (de data) staat dan vaak in een .DAT file.

BAK: Een zg. back-up van een door ED vervaardigde file. Ook tekstverwerkers produceren vaak (al dan niet automatisch) een .BAK file.

\$\$\$: Tijdelijk aangemaakte file, door PIP of ED. Worden weer gewist als de feitelijke file is weggeschreven.

SUB: Hulp-file voor het SUBMIT.COM programma. Vanuit deze file wordt een werklijst \$\$\$SUB vervaardigd en uitgevoerd.

WPF: Toevoeging voor files van de tekstverwerker (wordprocessor).

AUTOMATISEREN.

door Welmoed Jonker.

Automatiseren is een veelgehanteerde kreet in het bedrijfsleven. Over de wijze waarop, de middelen waarmee en bovenal waar dat moet gebeuren, willen de meningen nog wel eens uiteen lopen.

Dat het zonder inzicht binnenhalen van een computer 'omdat het naburige bedrijf dat ook doet' veel vaker leidt tot een chaos in de orde, dan het omgekeerde, is een bekend gegeven.

Hobbyisten met computers kennen vaak maar al te goed de problemen die ontstaan als de computer zijn intrede doet. Hebben ze u bij de aankoop van uw Exidy ook zo prettig voorgelicht, dat u dacht van de hel in de hemel te komen?

Gelukkig zien steeds meer bedrijven die zich met automatiseren bezighouden in, dat het niet alleen een kwestie is de klant een compleet systeem 'aan te smeren', maar dat het voor de toekomst beter is ook aandacht te besteden aan de nazorg.

Een computer is nu eenmaal zó gekocht en wanneer de gebruiker, zonder enige kennis van zaken, een duur gekocht programma wil gaan gebruiken, dan is het toch te gek dat het programma dat in de winkel zo fraai oogde, nu niet hetzelfde doet! Zo iets kan de gebruiker-zonder-achtergrondkennis diep in het verderf storten, of zoals een kennis van mij eens zei: '1000 hartaanvallen in een keer' bezorgen.

Waarom bedrijven automatiseren? Daarvoor zijn erg veel redenen aan te voeren, waarvan de eerder genoemde buurman er één kan zijn. Tegenwoordig zal dat geen steekhoudend argument meer zijn, maar de tijd is nog niet zover geleden dat zulks niet tot de onmogelijkheden behoorde!

Tegenwoordig zijn het vooral economische en organisatorische redenen die ervoor zorgen dat het elektronische bureau steeds verder voortschrijdt. U moet dit zeker niet zien in het perspectief 'vandaag nog met pen en inkt en mogen alle papier de deur uit'. Er zijn natuurlijk een aantal computer-giganten die in hun marketing dit soort kreten op de klanten afjaagt. Net als in de oorlog is dit een onderdeel van de strategie om de klant computer-rijp te maken. Houdt hem voor ogen dat hij straks het enige bedrijf heeft dat nog met antieke werk-methoden aan de gang is. Dat daarin niet alleen veel energie (van het personeel), maar vooral veel (door de baas betaalde uren) moet worden gestoken en voilà de eerste twijfel is gezaaid!

Met de juiste demonstraties (en dat wordt de verkoper aangeleerd) wordt vervolgens een aanval gedaan op de volgende verdedigings-linie. Valt ook die, dan bestaan er steeds minder bezwaren tegen automatisering. Op dat punt wordt het volgende psychologische bombardement op de argeloze en wijfelende klant losgelaten. Men toont hem hoe bepaalde programma's hem in staat stellen het bedrijf weer beheersbaar te maken. Als dat gebeurt op een moment dat men toch al heeft ervaren dat de greep op de bedrijfsvoering minder is geworden, dan is er weer een slag gewonnen.

Uiteindelijk zal de klant 'vallen' en kan het computer-bedrijf de buit gaan binnenhalen. Dat dit zo cru wordt gesteld, moet u zeker niet zien als een maatstaf voor de wijze van handelen door alle bedrijven. U zult echter ook wel hebben meegemaakt dat de verkoper, bij een vraag om informatie een ware stortvloed over u laat gaan. Als u na zijn betoog wat op adem komt en u gaat na wat er verteld is, dan zult u in veel gevallen tot de ontdekking komen dat u met die informatie nauwelijks wat kunt aanvangen. U hebt vaak niet het inzicht die gegevens op de juiste waarde te schatten en verder is de informatie zo doorspekt met vak-jargon dat zeker de adspirant-computer gebruiker daar niets van heeft gesnapt. Hij zal echter niet voor dom willen doorgaan en dus doen alsof hem alles duidelijk is.

Een voorbeeld van zo'n houding wordt gedemonstreerd in het VARA-radio programma van Spijkerman (dinsdag's rond 13.30 uur). Spijkerman heeft de gewoonte ergens heen te bellen en een verhaal af te steken dat is doorspekt met allerlei -zelfgemaakte- dure woorden. De persoon aan de andere kant van de lijn blijkt in veel gevallen nauwelijks te weten waarover Spijkerman het heeft, maar doet veelvuldig alsof hij snapt waar het om gaat en begint zichzelf dan, in zijn antwoord, de das om te doen! Hiermee wil ik alleen illustreren dat u onbekende begrippen steeds moet laten uitleggen, zodat het u inderdaad duidelijk is waarover wordt gesproken. Alleen dan bent u in staat een enigszins onderbouwd oordeel te vellen.

Terug naar het betoog.

Als je aan automatiseren denkt, moet je dan alles daaronder laten vallen, of moet dat niet?

Wanneer u terugdenkt aan uw kinderjaren, dan zult u zich kunnen herinneren dat u een vanzelfsprekende handeling als lopen ook niet in een keer heeft geleerd. Uw eerste pogingen gingen niet verder dan wat eenvoudige opdrakoefeningen. Daarna begon u voorzichtig u op te trekken totdat u wist hoe u moest staan. Vervolgens heeft u, zwaar leunend op de salontafel, uw eerste schreden gezet. Nu loopt u zonder daarbij na te denken, alhoewel het automatisme een enkele keer faalt: u ligt dan op uw gezicht!

Iets dergelijks zal zich ook bij automatiseren voordoen. Als u verstandig begint gaat u in de eerste tijd, na de komst van de treiteraar, pogen uw kennis van de bediening van de computer op een akseptabel niveau te brengen. Pas dan bent u in staat met programma's te leren omgaan! Waarom? Wel, gewoon omdat er in veel van de verkrijgbare programma's voor computers van wordt uitgegaan dat u bekend bent met de basis-begrippen en dus klaar bent voor het 'voortgezet onderwijs'.

We beginnen dus bij het begin en gaan ons afvragen wat in het bedrijfsproces zich leent voor automatisering.

In principe is dat vrijwel alles, maar omdat we voorzichtig zijn stellen we ons voor, alleen die werkzaamheden voor automatiseren voor te dragen, die vallen onder de categorieën 'tijdrovend', 'geestdodend' en 'repetent'. Daarbij kunt u zich elke combinatie van de genoemde categorieën als automatiseerbaar indenken. Automobielfabrieken zijn een duidelijk voorbeeld van vergaande automatisering: las-robots, onbestuurde (menselijk gesproken!) transportwagentjes, computer-gestuurde verfspuiten en lak-kontroles.

In kantoren zijn vele voorbeelden van automatiseerbare werkzaamheden te vinden. Tot die werkzaamheden behoren ook de zaken, waardoor b.v. tekstverwerkers en bestanden-programma's zo'n grote vlucht hebben genomen! Sekretarissen die steeds weer dezelfde, of vrijwel dezelfde brieven moesten tikken, kregen (volgens de fabrikanten) weer zin in het werk en de chef van het magazijn kon nu eindelijk eens op tijd en met minder moeite de voorraad-overzichten afleveren. Ook de boekhouding profiteerde van de automatisering. Was het in het verleden nogal tijdrovend om de debiteurenboekhouding bij te houden, door de koppeling van de magazijn-administratie aan de debiteuren-administratie was het faktureren en rappelleren een fluitje van een cent geworden.

Een welkome toevoeging is het calculatie-programma, in het Engels spreadsheet genoemd. Met dat programma werd het maken van een kostprijs-berekening gereduceerd tot het intikken van een moeder-model, waarvan elke keer weer gebruik kon worden gemaakt en waarbij vele prijs-varianten in een oogwenk konden worden geproduceerd.

Onze Exidy is een computer die thuishoort in de groep Personal Computers. Met die aanduiding bedoelde men de groep computers voor klein-zakelijk gebruik. U moet natuurlijk dan niet uitgaan van ons systeem in zijn kleinste samenstelling (computer, video-display en cassette-recorder(s)), maar in

de samenstelling (configuratie) voor bedrijven. Dat is minimaal computer, beeldscherm, twee diskdrives (5 of 8 inch) en een printer. De zakelijke gebruiker gaat ook niet uit (althans in vrijwel alle gevallen) van eigengemaakte programma's, maar van kant-en-klare, gekochte programma's. Tot de laatstgenoemde programma's behoren voor het Exidy systeem de Word-processor in pack-vorm, Spellbinder (ook een tekstverwerker), het calculatie-programma SuperCalc, bestanden-programma's als dBASE II. Door software-huizen zijn voor de Exidy ook faktureer-, voorraad- en boekhoud-programma's vervaardigd, waarvan vele zich ook met de modernere programma's kunnen meten.

Voor eigen gebruik waren genoemde programma's veelal niet interessant door de hoge aanschafprijs. De ware hobbyist zoekt in zulke gevallen naar een alternatief. Is dat er niet, of voldoet dat niet aan zijn wensen dan maakt hij zo'n programma. De CP/M- zowel als de ESGG-software bibliotheek bevatten vele voorbeelden van zulke programmatuur, vaak zelfs van behoorlijk hoge kwaliteit. Omdat, in tegenstelling tot de zakelijke programma's de gebruiker kan zien wat er in het programma staat (niet beschermd of gecompileerd) is hij ook in staat die programma's zondig naar believen aan te passen.

Hobbyisten kunnen eenlingen zijn, die slechts bezig zijn met hun hobby en deze niet dienstig maken aan een of ander doel, zij kunnen ook van de computer gebruikmaken om de werkzaamheden in werk of hobby te vereenvoudigen. Een programma als op schijf 2 (ESGG-bestanden programma), schijf 8 (cijfer verwerking LEAO) of schijf 12 (leden-administratie) zijn daarvan goede voorbeelden.

Juist voor deze groep gebruikers is het misschien nuttig een inzicht te geven in het hoe en waarom van de ontwikkeling van een bestanden-programma. We zijn van plan onderop te beginnen, met het waarom. Vervolgens geven we aan langs welke weg we denken tot een oplossing te kunnen komen en ten slotte geven we u aan wat onze oplossing is.

Als leidraad maken we gebruik van het ESGG-bestanden programma, om de simpele reden dat u dat waarschijnlijk al in huis heeft en omdat de schrijver dezes zich in dit programma heeft verdiept om daarna door middel van verbeteringen, een gebruiksvriendelijke versie te kunnen leveren. Een eerste poging vindt u op schijf 18, met voorbeeld-programma.

Programma-ontwikkeling.

Er zijn beduidende verschillen tussen de methoden van de professionele, en van de hobby-programmeur. De professionele programmeur -dat wil zeggen, degene die zich beroepshalve met programmeren bezig houdt- heeft een opleiding gehad die hem in staat stelt in de meeste talen die door computers worden gebruikt, programma's te kunnen vervaardigen. Hij is dan in het algemeen ook in staat in machinetaal te werken.

De hobby-programmeur doet in veel gevallen niet voor hem onder! Ook daar treffen we meesters in de verschillende talen, alleen komt het in die groep minder vaak voor dat de programmeur in vrijwel alle bekende talen kan werken. De meest gebruikte taal bij de niet-beroepsmatige programmeurs is Basic.

In tegenstelling tot de beroeps-programmeur, die 'van huis uit' is aangeleerd gestructureerd te programmeren, maakt de hobby-programmeur vaak zijn programma 'voor de vuist weg'. Het vooronderzoek, het opzetten van de stroom-schema's wordt in deze groep vaak als overbodig en tijdvretend ervaren. Veel van de programma's die u onder ogen krijgt, kunnen zijn vervaardigd zonder vooronderzoek, direkt achter het toetsbord ontwikkeld. Dat wil niet zeggen dat hobby-programmeurs per definitie slecht programmeren! Zeker niet!

Toch kunnen we er niet omheen, te leren omgaan met een gestructureerde wijze van programmeren, waarbij eerst alle wensen worden gespecificeerd,

de algoritmen worden ontwikkeld en pas in de laatste fase, het programma zelf wordt gemaakt.

Als u op dit moment denkt dat het genoemde ESGG bestanden-programma dan wel met gebruikmaking van die methode zal zijn ontwikkeld, moet ik u teleurstellen. Het programma zelf is 'even vlug ertussendoor' door Charles Nettelner vervaardigd, omdat hij een programma nodig had waarmee hij de a-bonnee's van het periodiek kon registreren. Door zijn ervaring als voormalig programmeur kon dat ook wel zo.

Het programma was verder alleen bedoeld voor eigen gebruik en behoefde van hem dan ook niet gebruikersvriendelijk te zijn.

Toch is hij voor de aandrang bezweken en heeft de set programma's aan de ESGG bibliotheek beschikbaar gesteld. Het programma had een KISS2-achtige structuur, maar was geheel berekend en toegesneden op schijf-systemen. Om een redelijke snelheid bij lezen en schrijven te bereiken had Charles gekozen voor de RANDOM-structuur, waarbij een record ergens in de data-file eerder is te vinden dan wanneer van een sequentiële structuur gebruik gemaakt wordt. Als voormalig cassette-systeem gebruiker (en wie is dat niet geweest?) kende ik KISS2, waardoor ik het werken met het ESGG-pakket vrij snel onder de knie had.

Toch zinde mij de opzet niet. Niet zozeer vanwege de opzet, maar veeleer vanwege de wijze waarop de gebruiker (u en ik) niet met zachte hand werden gekorrigeerd wanneer een fout werd gemaakt. Ook de presentatie op het beeldscherm liet te wensen over. Let wel, dit is geen kritiek op Charles! Hij had het programma tenslotte niet voor ons bedoeld!

Wanneer je je dan eenmaal in de structuur van een programma verdiept, dan zie je steeds meer punten die om aandacht en verbetering vragen. Zo komt het dan, dat het aanpassen en verbeteren niet 'even' is te doen, maar nu al bijna twee jaar gaande is. Het einde is er nog niet, maar begint nu wel in zicht te komen!

Terugkomend op de methoden die in het professionele vlak worden gehanteerd zien we dat daar volgens een vast patroon wordt gewerkt. Een van die methoden is van Nederlandse origine en heeft de naam 'SDM' ofwel System Development Methodology meegekregen. Deze methode is natuurlijk niet de enigszins makende, maar ook hobby-programmeurs kunnen hun voordeel doen met kennis van de wijze waarop hun vakbroeders plagen te werken. In SDM legt men de nadruk op een gefaseerd verloop van de vervaardiging van een programma. Deze fasen bestaan in het algemeen uit: onderzoek, ontwikkeling, test, terugkoppeling en bijstelling, voordat het produkt gereed is voor gebruik. Door steeds een dergelijke lijn te volgen, zal het beoogde doel waarschijnlijk niet zoveel eerder, maar wel met een beter resultaat worden bereikt. Systematisch werken levert een beter doordacht produkt dat, door het voorwerk, meestal minder fouten zal bevatten en beter aan de 'klantspecificaties' zal voldoen.

Wat beoogt men feitelijk met SDM. Kort gezegd: structuur in de ontwikkeling, zodat niet alleen de ontwikkelaar, maar later ook andere programmeurs met het produkt uit de voeten kunnen. De SDM kan zowel voor kleinschalige als grote projecten worden gebruikt, terwijl ook nieuwbouw c.q. vernieuwbouw van informatiesystemen tot het dekkingsveld worden gerekend. In het informatica-onderwijs is SDM een 'verplicht nummer' bij de lessen over systeem-ontwikkeling. SDM is geen diktator, meer een gids. Het vertelt de ontwikkelaar wat hij moet doen, en op welk moment dat dient te gebeuren. Ook kan hij erin vinden hoe dat moet gebeuren, maar SDM biedt geen technieken, zoals gestructureerd programmeren, gestructureerd ontwerpen of iets dergelijks. Daardoor is SDM uitstekend te combineren met zelf-ontwikkelde, of voor de eigen omstandigheden beter toepasbare technieken.

Een van de 'ontwikkelaars' van SDM, Pandata, heeft voor de meest voorkomende situaties een aantal technieken ontwikkeld. Daaronder vallen TIA

(Technieken voor Informatie Analyse), GOS (Gestructureerd Ontwerpen van Systemen), GOP (Gestructureerd Ontwerpen van Programma's) en TOT (Technieken Om te Testen). Het scheiden van de technieken van de System Development Methodology heeft in belangrijke mate bijgedragen aan het succes ervan. De steeds veranderende inzichten omtrent programmeren en ontwerpen hebben dan ook nauwelijks invloed gehad op de wijze waarop SDM is opgebouwd. Dit maakt deze methodiek dan ook tot een konstante faktor in vele organisaties. Het bevordert daarmee de continuïteit van de wijze waarop systemen worden ontwikkeld, hetgeen een belangrijk voordeel is voor het uitvoeren, organiseren en managen van systeem-ontwikkelings-activiteiten in bedrijven.

SDM kent de volgende fasen:

0. de voorstudie;
1. de definitie-studie;
2. het functioneel ontwerp;
3. het technisch ontwerp;
4. het programmeren en toekennen van taken;
5. het testen;
6. de konversie en invoering;
7. het systeemgebruik en -beheer.

Het gaat natuurlijk uiteindelijk om de ontwikkeling van een systeem. Nauwkeuriger uitgedrukt: om de ontwikkeling van een informatie-systeem, waarbij het niet noodzakelijk is om direkt al aan computers te denken! Er zijn immers legio informatie-systemen die puur handmatig zijn. Een voorbeeld van zulk een handmatig systeem is b.v. de leden-administratie van een kleine voetbalclub.

Bij SDM gaat het natuurlijk wel in de eerste plaats om het werken naar een geautomatiseerd systeem!

Om met zo'n systeem te kunnen werken is apparatuur en programmatuur nodig. De apparatuur is in het algemeen niet specifiek voor de toepassing, terwijl de programmatuur -de software- juist wel specifiek voor een bepaalde toepassing is. Wanneer we dit kort, uit de losse pols formuleren, kunnen we stellen dat de programmatuur de toepassing is, het informatie-systeem.

Laten we samen eens proberen na te gaan, hoe een methode om een systeem te ontwikkelen eruit zou kunnen zien, uitgaand van het programma. Volgens de regels van SDM wordt programmatuur geschreven in fase 4 en in de daarop volgende fase getest. Het -naar iedereen hoopt- inmiddels foutloze programma kan vervolgens worden ingezet (fase 6), maar het werkzame leven begint pas in de daarop volgende fase.

Voordat een programma geschreven kan worden, moet er behoorlijk wat werk worden verzet. Allereerst worden de doelstellingen van het te ontwerpen informatie-systeem vastgelegd (fase 1), waarna deze moeten worden vertaald in een systeem-ontwerp. Dat gebeurt dan in de fasen 2 en 3. Tijdens de eerste fase van het systeem-ontwerp, het zg. functioneel, of basis-ontwerp wordt het systeem in termen van de gebruikers-eisen gespecificeerd, waarna tijdens de tweede ontwerp-fase (3), het technisch of detail-ontwerp, het systeem zover wordt uitgewerkt dat de programmeur aan de slag kan. Hij zal de taken gaan beschrijven. Dit laatste is zeker geen overbodige toevoeging omdat een informatie-systeem in het algemeen bestaat uit programma's die machinale activiteiten beschrijven en uit procedures die handmatige-acties beschrijven. Juist voor die procedures moeten de taak-beschrijvingen worden gemaakt. De volgende fase (4) van SDM is dan ook niet compleet als het toekennen van de taken er niet aan is toegevoegd.

De 7 fasen, zoals eerder genoemd, worden het SDM-trajekt genoemd. In veel gevallen is het zinvol om, alvorens het SDM-trajekt in te gaan, te onderzoeken of dat nu wel zinnig is om te doen. Dit zogenaamde voor-onderzoek

behoort niet tot de SDM en om dat tot uitdrukking te brengen, spreekt men van fase 0. Ziedaar.

Al eerder is gesteld dat SDM voorschrijft wat er gedaan moet worden, zonder aan te geven op welke wijze. Een systeem-ontwerper heeft dus eigenlijk niet zo veel aan SDM, tenminste niet inhoudelijk. Hij wil weten hoe tewerk moet worden gegaan, met andere woorden, hij moet kennis hebben van de technieken. Voor de manager ligt dat anders: hij hoeft niet precies te weten hoe iets gedaan wordt, maar wat hij wel wil weten is hoever men is gevorderd met het werk. Hij zal SDM dan ook een prachtige kapstok vinden.

Een methode zou geen methode zijn, als de fasen ervan ook niet weer waren opgedeeld in deeltaken, de activiteiten. Fase 1 in SDM kent 11 activiteiten en fase 2 zelfs 14!! SDM schrijft per activiteit ook de produkten voor. Activiteit 2.11 produceert b.v.:

- de benodigde apparatuur;
- de leverings-schema's;
- de bestelling c.q. reservering van de apparatuur.

Voor de hobby-programmeur geen probleem: hij heeft de apparatuur al en de volgende stap kan door hem worden gedaan. Als SDM u een omslachtige methode lijkt, dan moet dat -vanuit de hobby-sfeer bekeken- worden beaamt. Ongenschijnlijk gaat u heel wat tijd steken in het op papier zetten van uw wensen en eisen. Toch is het een zinvolle bezigheid. Het dwingt u namelijk goed na te denken over de wijze waarop straks uw programma moet werken. Is het niet beter vooraf alles goed te overzien, dan, wanneer u denkt dat het programma klaar is, er achter te moeten komen dat u iets over het hoofd heeft gezien? Bot gezegd: als u een bestanden-programma gaat maken mag het toch niet voorkomen dat er bijvoorbeeld een uitvoer-routine (om afdrucken op papier te maken) ontbreekt?

Verder mag een programma zich toch niet ophangen alleen maar omdat op een vraag een niet-toegelaten antwoord werd gegeven (een alfa-numerieke invoer in een numerieke variabele?).

Daarom, besteedt er wat tijd aan, het zal u zeker goed van pas komen.

Een ander aspekt van het werken met SDM, en dan moet u even niet als computer-enthousiast denken, is dat in een vroeg stadium al wordt vooruitgelopen op het moment van invoering. Beter daar te vroeg, dan te laat aan denken! Vooral in bedrijven speelt dat een rol omdat de meeste werknemers nogal angstig het moment tegemoet zien dat ze als het ware 'voor de leeuwen' worden geworpen! Je hebt nu eenmaal niets aan een prachtig informatie systeem, waarmee niemand wil werken.

De rode draad door SDM is dan ook de konsekvente aandacht voor de menselijke taken in het uiteindelijke systeem, met het motto dat het gemakkelijker is de computer-verwerking aan te passen aan de mens, dan andersom. De menselijke taken staan primair in deze methode van ontwikkeling.

Verder is SDM sterk zelf-dokumenterend. Elk produkt is een bouwsteen van de dokumentatie. Het eind-rapport van elke fase (denk eraan, dat dit een bedrijfsmatig proces is!) kan dan ook voor het overgrote deel bestaan uit de produkten van voorgaande activiteiten van die fase. In dit verband wordt daarom wel eens de kreet: mijlpaal-dokumentatie geslaakt. De eind-rapporten sluiten fasen af van een proces, waarvan tot in detail bekend is wat er is gebeurd en wat er nog moet worden gedaan.

In een volgend deel wordt ingegaan op ons onderwerp, het ESGG bestanden-programma, waarbij zal worden getracht een en ander aan de praktijk te toetsen.

(wordt vervolgd)

NAGEKOMEN MEDEDELING

Toekomstige RAMDISK gebruikers, let op!

Rob Borkent heeft opgemerkt dat de huidige RAMDISK software nog niet te gebruiken is, wanneer u 8" drives (en bijbehorende CP/M) heeft. De thans gebruikte software is niet in staat de BIOS goed aan te passen.

Aan een speciale versie voor 8" wordt gewerkt. Wilt u meer weten? Spreek dan Rob Borkent aan op de Sorcerer Dag (in de Technische Hoek).

I C R O ' S M I C R O ' S M I C

Te koop gevraagd: 1 diskdrive hardsectored 77 tracks, liefst Micropolis. R.B.J.Th. Vernimmen, tel. 01608-14078.

te koop: Exidy Sorcerer 52 k intern geheugen, met 2 ingebouwde diskdrives, 640 kb opslagcapaciteit, CPM 2.2. Apparatuur nog in bedrijf. prijs n.o.t.k. G.A.P.Sanders, Kennedylaan 25, 5768 VL MEIJEL (L)04766-2226.

A.S. zaterdag
27-9-86
SORCERER Dag!!
Komt U ook??

