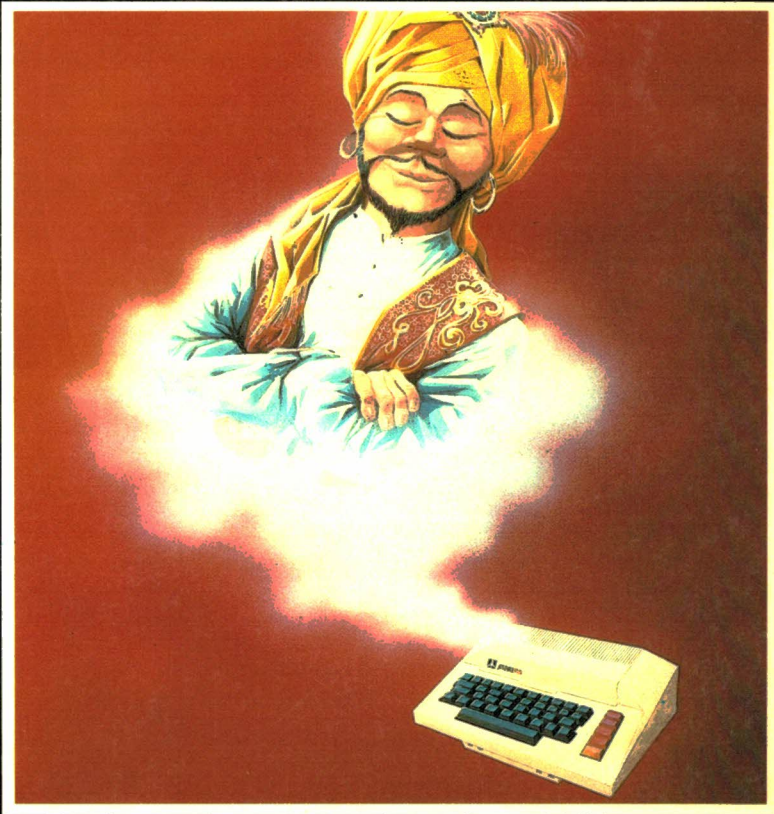


ATARI
MUSEUM.NL



Praktische gids
ATARI
600XL/800XL



Thomas Blackadar



Praktische Gids voor de ATARI 600XL/800XL



Praktische Gids voor de ATARI 600XL/800XL

Thomas Blackadar



DÜSSELDORF
BERKELEY · PARIS · LONDON



Originele uitgave in het engels

Titel van de engelse uitgave: The Easy Guide to Your ATARI 600 XL/800 XL

Origineel copyright © 1983, SYBEX INC., Berkeley, Calif., USA

Zetsel: tgr – typo-grafik-repro gmbh., remscheid

Druk: Drukkerij Salland BV, Deventer

Nederlands vertaling door **Emile Gemmeke**

Alle moeite is gedaan om zo compleet en accuraat mogelijke informatie te verstrekken. Sybex aanvaardt echter geen enkele verantwoording noch voor het gebruik ervan, noch voor een mogelijke inbreuk op patenten of andere rechten van derden welke eruit zouden voortvloeien. Fabrikanten houden zich het recht voor circuits te veranderen zonder verdere aankondiging. Technische specificaties en prijzen zijn aan snelle veranderingen onderhevig. Vergelijkingen en evaluaties zijn gegeven om hun onderwijskundige waarde. Voor exacte specificaties dient de lezer de gegevens verstrekt door de fabrikant te raadplegen.

ISBN 3-88745-109-0

1. uitgave 1985

Alle nederlandstalige rechten voorbehouden.

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

Printed in the Netherlands

Copyright © 1985, SYBEX-Verlag GmbH, Düsseldorf

Verantwoording

Een boek als dit vereist teamwork van een heleboel mensen. Ik zou vooral graag Rudolph Langer en Paul Panish willen bedanken voor hun steun en adviezen met betrekking tot de ontwikkeling van dit project. Ook Joel Kriesman die waardevolle technische adviezen gaf en David Kolodney die een goede tekstbewerking heeft gedaan wil ik van harte bedanken. Ook mijn dank aan de mensen die zoveel gedaan hebben om van het manuscript een boek te maken: Valerie Robbins en Leslie Horston, tekstverwerking; Gloria Maya, Valerie Brewster en Cheryl Vega, lay-out; Janis Lund en Elizabeth Thomas, tekstcontrole en Mary Rose Orgen, ontwerp. Zij hebben waardevolle bijdragen geleverd.

Van Atari heb ik veel hulp gekregen van Margaret Lasecke en haar medewerkers. Van mijn oorspronkelijke contacten met Bruce Quill tot mijn uiteindelijke band met Trish Marino, was iedereen vriendelijk en behulpzaam.

Inhoud

Inleiding	11
DEEL 1 KENNISMAKING	15
1 Kennismaking met de Atari Computers	17
Angst	17
Wat is een computer?	18
Wat kan een computer?	19
De Atari computers	20
De installatie	23
Randapparatuur	27
2 Commerciële Software	31
Het kopen van programmatuur	32
Spelen	33
Serieuze programma's	36
Het gebruik van software	39
Wat er nog komt	40
3 Toetsenbord en Beeldscherm	43
Het toetsenbord	43
Cursorbesturing	47
RETURN en de foutmelding (ERROR)	51
Uw eerste instructie: PRINT	53
De ESC-toets	57
De speciale functietoetsen	60
Vrije oefeningen	63
Samenvatting	63

4 Grafieken op de Atari	65
Wat zijn grafieken?	65
Tekst: GRAPHICS 0	67
De POSITION opdracht	68
Vierkleuren tekening: GRAPHICS 7	72
Vrije oefeningen	79
Samenvatting	79
 DEEL 2 PROGRAMMEREN	 81
 5 Een programma schrijven	 83
Programma's opslaan	84
Programma: vierhoek tekenen	91
Programmeertips	92
Het opsporen van fouten (vlooienjacht!)	93
Opslaan op cassette	96
Vrije oefeningen	98
Samenvatting	99
 6 Het gebruik van variabelen	 101
Het opslaan van getallen	101
Variabelen in programma's	105
De READ en DATA opdrachten	111
INPUT: waarden toekennen vanaf het toetsenbord	113
Berekeningen maken	117
Strings: opslaan van letters	121
Vrije oefeningen	123
Samenvatting	124
 7 Het besturen van uw programma	 125
GOTO: springen naar een andere opdracht	125
FOR/NEXT lussen	130
IF instructies: beslissingen nemen	138
GOSUB: subprogramma's	142
Vrije oefeningen	145
Samenvatting	146

8 Geavanceerde grafische technieken	147
Het veranderen van kleur met SETCOLOR	148
Verwijderen van het tekstraam	154
De grafische standen	156
De tekststanden	161
Samenvatting	165
DEEL 3 HET OPSLAAN	167
9 Het opslaan op cassette	169
Het klaarzetten van de programma recorder	170
Het laden van kant en klare cassette programma's	171
Het opslaan van uw eigen programma's	173
Andere manieren om programma's op te slaan	175
Samenvatting	178
10 De disk drive	179
Het installeren van de disk drive	180
De omgang met diskettes	182
Het aanzetten van de disk drive	184
Het gebruik van kant en klare disketteprogramma's	187
De DOS disk	188
Bestanden	192
Het opslaan van eigen programma's	194
Diskettes kopiëren	196
Samenvatting	198
11 Nawoord	199
Andere uitbreidingsmogelijkheden	199
Andere wegen	201
Bijlagen	
A Verdere Literatuur	205
B BASIC woordenlijst	207
C Atari foutcodes	213
Oplossingen van de geselecteerde oefeningen	217
Trefwoordenlijst	220

Inleiding

Wanneer u net een Atari computer gekocht hebt, zult u wel verstoeld hebben gestaan toen u de doos opende. De computer zat er wel in, maar een instructieboekje? Alles wat u aantroef was een dunne *Owner's Guide* en een *Atari BASIC Reference Guide*, met de opmerking „for experienced programmers only” (alleen voor programmeurs met ervaring).

Het spijt mij te moeten zeggen dat dat alles is. Atari geeft eerlijk toe dat zij u niet meer leren dan de belangrijkste dingen. „Als u meer informatie wilt hebben dan moet u een boek kopen” zeggen zij.

Uw Atari computer is een geweldig ding: hij werkt goed en kan met uw eisen meegroeien. Ik heb dit boek geschreven als een soort gids, zodat u snel kunt leren wat u van uw computer moet weten.

Door het boek heen zult u de omgang en het gebruik met uw computer leren. U zult ontdekken waar u programma's kunt kopen zodat u uw computer meteen kunt gebruiken. Daarna leert u hoe u uw eigen programma's maakt en opslaat met behulp van een cassette recorder of disk drive.

Wat u van te voren moet weten? Niets. Dit is een *eenvoudige handleiding*, die ervan uitgaat dat u nooit een computer gebruikt hebt. U start helemaal aan het begin en het gaat langzaam genoeg om de begrrippen te leren zonder het computerjargon nodig te hebben.

U kunt dit boek ook gebruiken, als u al wat meer kennis hebt. U kunt dan de hoofdstukken overslaan die u al begrijpt en meer aandacht schenken aan gedeelten die u nog niet kent. Ik heb geprobeerd er een serieus en prettig leesbaar boek van te maken. Ik ga ervan uit dat u erg geïnteresseerd bent in uw Atari computer en dat u een duidelijke uitleg wilt, zonder veel poespas.

Ik heb ook geprobeerd me te concentreren op attractieve en leuke mogelijkheden van de Atari.

De hoofdstukken zijn gegroepeerd in drie delen. Op die manier kunt u uw studie splitsen in de drie fundamentele fases: kennismaking, programmeren en opslaan.

Het eerste deel, kennismaking, brengt u de basiskennis van uw computer bij. Voordat u iets anders doet, moet u leren hoe een computer werkt en hoe eenvoudige instructies ingetypt worden. Dit leert u spelenderwijs.

Het tweede deel gaat over programmeren. U leert eenvoudige instructies samen te stellen tot een geheel dat in staat is een complexe handeling te verrichten. Tijdens het leren van de principes van computerprogrammering, leert u tevens BASIC, de belangrijkste taal die door de Atari computers gebruikt wordt en een van de meest gebruikte computertalen.

Deel drie gaat over een onderwerp dat al eerder aan de orde kwam: het permanent opslaan van programma's op cassette of diskette. Hoofdstuk 9 beschrijft hoe u de Atari programma recorder met cassettebandjes moet gebruiken. Hoofdstuk 10 handelt over het gebruik van een disk drive voor snel en betrouwbaar opslaan. Op sommige plaatsen in dit boek zult u de neiging hebben om al naar deze hoofdstukken vooruit te kijken, vooral als u commerciële programma's gebruikt of uw eigen programma's wilt opslaan zodat u er ook naderhand weer mee kunt werken.

Om dit boek efficiënt te kunnen gebruiken moet u over een Atari computer beschikken. Voor het grootste gedeelte van dit boek hebt u de computer nodig om de voorbeelden op de voet te kunnen volgen.

Het maakt niet uit welke Atari computer u hebt. Dit boek is bedoeld voor de nieuwe typen 600XL™ en 800XL™, maar de beschrijving geldt ook voor de andere Atari modellen. De in ontwerp zijnde Atari 1400XL™ en de 1450XLD™ zijn in principe hetzelfde als de 800XL. Ze hebben alleen een paar eigenschappen meer. Alles in dit boek slaat ook op deze computers en de verschillen zijn in de tekst opgenomen.

Als u een van de oudere modellen hebt (model 400™, 800™ of 1200XL™ die niet meer gemaakt worden) dan hebt u de Atari *BASIC Computing Language* cartridge nodig voor dit boek. Wanneer u deze cartridge in uw computer stopt zal hij zich als een Atari 600XL of 800XL gedragen en u kunt ieder programma laten lopen alsof u een van de nieuwere computers hebt. Er zijn overigens enkele kleine verschillen met betrekking tot enkele details van het toetsenbord en de grafische mogelijkheden. Lees uit hoofdstuk 1 het gedeelte over „de Atari computers” als u van plan bent dit boek te gebruiken voor een ander model dan de 600XL of 800XL.

Computers lijken nogal geheimzinnig zolang u niet weet wat u er precies mee kunt doen. Maar na de eerste pogingen kunt u er al mee overweg. Dit boek is slechts een hulp bij het nemen van de eerste stappen.

Deel 1

Kennismaking



Hoofdstuk 1

Kennismaking met de Atari Computers

U heeft net een Atari computer gekocht en staat waarschijnlijk te trappelen van ongeduld om hem te kunnen gaan gebruiken. U bent benieuwd wat hij kan en hoe u hem de baas kunt.

Dit hoofdstuk is een overzicht. Het geeft enkele tips die U helpen handig met uw Atari om te gaan en gedetailleerde instructies om hem te installeren. Aan het einde van dit hoofdstuk zult u met uw computer kunnen beginnen te werken.

Angst

Voordat het avontuur begint, moet u enkele angsten overwinnen. Als het de eerste kennismaking met een computer is, dan bent u er misschien een beetje van onder de indruk. Dat is niet nodig: een computer is een machine, meer niet. Zoals een televisie of een afwas-machine kan een computer u helpen als u leert hoe u hem moet gebruiken en hij bijt niet als u een fout maakt.

U hoeft niet bang te zijn dat u de computer kapot maakt door een verkeerde toets in te drukken. Uw computer is een elektronisch apparaat dat dezelfde zorgvuldige behandeling vereist als een radio of een rekenmachine, maar u hoeft er niet bang voor te zijn. Bij normaal gebruik kan er weinig gebeuren. Het ergste dat er kan gebeuren is dat de woorden en getallen die u net ingetypt hebt verloren gaan, maar de computer kunt u daarmee niet beschadigen.

Uw Atari is uitgerust met een soort paniekttoets. Dit is de RESET-toets, die zich aan de rechterzijde van het toetsenbord bevindt. Als er iets gebeurt waardoor u niet meer weet wat er aan de hand is, dan kunt u altijd nog deze toets indrukken. De computer zal dan terugke-

ren naar de stand waarin hij bij het aanzetten stond en is klaar om opnieuw te starten. En als dat niet helpt dan kunt u ook nog altijd de stroom uitschakelen en weer aanzetten.

Als u toch nog aarzelt, speel dan een spelletje. Er is geen betere manier om aan een computer te wennen dan er iets leuks mee te doen. In hoofdstuk 2 leert u welke Atari spelletjes u het beste kunt kopen, maar nu wilt u waarschijnlijk wel eens met het bekendste spel beginnen: Pac-Man. Atari's versie van dit spelletje is een uitstekend geneesmiddel tegen computerangst.

Wat is een computer?

Hoe werkt een computer? Dat hoeft u niet te weten. U kunt hem als een gesloten doos beschouwen die informatie opneemt, door elkaar gooit en de resultaten zichtbaar maakt. Tijdens dit proces vinden ingewikkelde handelingen plaats, maar daarover hoeft u niet na te denken. Kijk alleen maar naar wat u erin stopt en wat eruit komt.

U praat met de computer door middel van het *toetsenbord*. Atari computers hebben een volledige toetsenbezetting zoals een typemachine en dit geeft de mogelijkheid letters, getallen en speciale symbolen in te typen. U gebruikt al deze toetsen voor uw contact met de computer.

Om iets terug te zeggen heeft de computer een elektronisch circuit dat woorden en beelden op een televisiescherm kan zetten. Hierdoor laat de computer de letters die u intypt op het beeldscherm verschijnen. Ook wanneer het nodig is iets te beantwoorden dan laat hij dat zien. In de *grafische stand* is het mogelijk de computer tekeningen te laten maken op het scherm.

De computer heeft een *geheugen*, waarin hij informatie voor toekomstig gebruik kan opslaan. U kunt dit geheugen gebruiken om er programmaregels in op te slaan, of de getallen die u in een berekening wilt gebruiken, of de woorden van een tekst die u aan het schrijven bent.

De hoeveelheid die u kunt opslaan op welk moment dan ook hangt af van de grootte van het computergeheugen. Dit wordt gemeten in *bytes*. Een byte is ongeveer de ruimte die nodig is om een letter of een

getal dat uit drie cijfers bestaat op de slaan. Meestal wordt de geheugenhoeveelheid aangeven met de letter „K”, wat ongeveer een duizendtal bytes inhoudt (1024 om precies te zijn). Een K kan ongeveer een derde van deze pagina bevatten.

Voor de toepassingen in dit boek hoeft u zich geen zorgen te maken over de grootte van het geheugen van uw computer. De Atari 600XL heeft een 16 Kbytes ingebouwd geheugen - meer dan genoeg voor welke toepassing dan ook in dit boek. De andere Atari's hebben zelfs meer. In een later stadium kunt u het geheugen van de 600XL uitbreiden zodat u ook gecompliceerdere programma's kunt uitvoeren.

Een ander punt met betrekking tot het geheugen van uw computer: alles wat u erin opbergt wordt uitgewist als u het toestel uitschakelt. Om de informatie permanent op te slaan hebt u een *programma recorder* of een *disk drive* nodig.

Wat kan een computer?

Uw Atari computer kan snel en gemakkelijk ingewikkelde berekeningen uitvoeren diagrammen en kleurrijke beelden op uw televisiescherm tonen. Met een tekstverwerkingsprogramma kunt u een brief veranderen zonder hem opnieuw te moeten typen. En natuurlijk kunt u er spelletjes mee doen.

De grote kracht van de computer is zijn flexibiliteit. Hij kan nummerieke berekenen maken als u dat wenst of zich beperken tot woorden en letters. Hij kan op het toetsenbord reageren en het scherm opmaken zoals u dat zelf wenst. Met zijn geheugen kan hij op zeer nuttige wijze informatie opslaan en ordenen.

U bestuurt uw computer met een *programma*, een gedetailleerde lijst van instructies die hij als een soort recept, stap voor stap, volgt om zijn taak uit te voeren. Het programma kan zo eenvoudig of ingewikkeld zijn als u zelf wilt en het kan ervoor zorgen dat de computer beslissingen neemt. Men noemt programma's meestal *software*.

Een goede manier om te beginnen is programma's kopen die door iemand anders geschreven zijn. Wanneer u commerciële programma's gebruikt hoeft u niet te weten hoe deze werkt. U geeft alleen algemene instructies en beantwoordt enkele vragen, waarna het pro-

gramma de details uitvoert. Hoofdstuk 2 is een handleiding voor enige programmatuur die u kunt kopen. Spelletjes zijn erg populair en de Atari computers hebben de meest uitgebreide keus aan onderhoudende software van alle homecomputers. U kunt echter ook programma's kopen voor andere doeleinden, zoals onderwijs en eigen huishouding. Tekstverwerking is in toenemende mate populair bij degene die graag willen beschikken over een elektronische typemachine.

Op een gegeven moment wilt u uw eigen programma's schrijven, zodat u uw computer de baas kunt. Commerciële programma's zijn duur en soms onhandig. Als u van plan bent uw computer optimaal te gebruiken, dan moet u hem zelf direct kunnen besturen. In de hoofdstukken die nog komen leert u uw eigen programma's schrijven. Op deze manier kunt u uw computer precies zeggen wat hij moet doen. Tegelijkertijd kunt u de verdere mogelijkheden van de computer testen zodat u hem de baas wordt.

De Atari computers

Atari heeft veel verschillende computers verkocht sinds ze op de markt kwamen. Om verwarring te voorkomen, zal ik de ontwikkeling en de kleine verschillen ervan uitleggen. Figuur 1.1 geeft een overzicht van de modellen. Ze kunnen in twee groepen worden verdeeld: de nieuwe en de oude.

De nieuwere, die voor het eerst eind 1983 werden verkocht hebben allemaal een XL aanduiding in het modelnummer en een wit-bruin frame. Van deze nieuwe computers waren bij het ter perse gaan van dit boek de goedkope versies 600XL en 800XL verkrijgbaar. De andere twee, 1400XL en 1450XLD, zijn nog steeds in ontwikkeling. Men zegt dat de ontwikkeling voor een of waarschijnlijk beide wordt stopgezet en dat ze nooit op de markt zullen komen. In dit boek komen zij volledigheidshalve wel voor, maar worden niet in detail behandeld.

Voordat Atari de nieuwe computers introduceerde, verkochten ze de modellen Atari 400 en 800. Ondertussen was er een Atari 1200XL, een model dat slechts enkele maanden in de handel is geweest.

kenmerken	600XL	800XL	1400XL	1450XLD	400	800	1200XL
	(verkrijgbaar)		(gepland)		(uit de productie)		
omvang geheugen	16K	64K	64K	64K	16K	48K	64K
Typemachine toetsenbord	•	•	•	•	*	•	•
ontwerp XL-style	•	•	•	•			•
Programmeerbare functietoetsen			•	•			
Ingebouwde BASIC	•	•	•	•			
Monitoraansluiting		•	•	•		•	•
Telefoonaansluiting			•	•			
Spraaksynthesizer			•	•			
Ingebouwde diskdrive				•			
Grafische instellingen	11	11	11	11	6-9	6-9	11
Tekstinstellingen	5	5	5	5	3	3	5
					* platte toetsen		

Figuur 1.1: De kenmerken van de Atari computers

Gelukkig zijn al deze modellen slechts variaties van dezelfde computer. De oudere modellen missen een paar eigenschappen en daardoor is het noodzakelijk dat u een aparte BASIC taal cartridge gebruikt. Enkele van de nieuwere XL computers kregen ingebouwde uitbreidingen, maar over het algemeen zijn deze computers hetzelfde en kunnen dezelfde programma's uitvoeren.

Dit boek is in eerste instantie bedoeld voor Atari 600XL, afgebeeld op figuur 1.2, en 800XL, een grotere versie met een groter geheugen. U kunt het echter ook voor de andere oudere of nieuwere modellen gebruiken. Alle beschreven programma's werken op de nieuwe computers en wat het uiterlijk betreft zijn er slechts kleine verschillen. Als u een van de oudere computers hebt (400, 800 of 1200XL), dan hoeft u alleen maar een paar dingen aan te passen.

Waarom al deze modellen? Atari probeert een lijn van computers te ontwerpen waaruit iedereen iets van zijn gading kan vinden. Model 600XL is bedoeld voor degene die vooral gebruik willen maken van

spelletjes en eenvoudige programma's. Voor degene die een computer met meer mogelijkheden wil hebben zijn er twee modellen die iets duurder zijn. Elk model heeft een paar mogelijkheden meer dan het vorige. De computers lijken echter zoveel op elkaar dat u de 600XL kunt uitbreiden tot een van de andere modellen door extra apparaten bij te kopen.

Als u een uitbreiding van het geheugen in een 600XL aanbrengt, wordt het in feite een 800XL. De toekomstige 1400XL en 1450XLD zijn hetzelfde als de 600XL en 800XL, met enkele extra mogelijkheden. Zij hebben een ingebouwd *telefoon modem*, waarmee het mogelijk is programma's en informatie via de telefoon naar andere computers te zenden. Zij hebben ook een ingebouwde *spraak synthesizer* die het mogelijk maakt de computer Engelse woorden te laten spreken. De bijzonderheid bij model 1450XLD is de ingebouwde disk drive en de ruimte voor een tweede. Als u een van deze computers hebt, kunt u beginnen alsof u het een 600XL is. Daarna kunt u dan proberen de overige mogelijkheden onder de knie te krijgen.

Als u een van de Atari modellen 400, 800 of 1200XL hebt die voor het einde van 1983 verkocht werden kunt u dit boek ook gebruiken, maar dan moet u opletten. Om te beginnen is het uiterlijk van deze computers anders dan van de nieuwe modellen en sommige toetsen zitten



Figuur 1.2: Dit boek is geschreven voor de Atari 600XL. De 800XL is in wezen hetzelfde, maar met een groter geheugen.

op een andere plaats. Als u hoofdstuk 3 leest zult u enkele details met betrekking tot het toetsenbord moeten veranderen om ze bij uw computer te kunnen gebruiken.

Een ander verschil heeft betrekking op de grafische mogelijkheden. Het grafische systeem is in principe hetzelfde zowel bij de oude als bij de nieuwe Atari's, maar de XL modellen hebben een paar mogelijkheden meer. Enkele staan beschreven in hoofdstuk 8. Als u een van de oude computers hebt dan kunt u geen gebruik maken van deze geavanceerde grafische mogelijkheden, maar met de rest van het boek zult u geen problemen hebben.

In tegenstelling tot de nieuwere modellen missen Atari 400, 800 en 1200XL die niet meer gebouwd worden, de ingebouwde BASIC programmeertaal. Indien u iets anders wilt gebruiken dan alleen commerciële programmatuur moet u de Atari BASIC Computing Language cartridge kopen. *Als u dit boek bij een van de oudere modellen gebruikt moet deze BASIC cartridge in uw computer steken.* Figuur 1.3 geeft een overzicht van de Atari computers en beschrijft waarop u eventueel tijdens het lezen moet letten.

De installatie

Het enige dat u naast de computer nodig hebt is een televisietoestel. U kunt een gewoon type gebruiken, maar u krijgt de beste resultaten met een nieuwer model. Een televisie die ouder is dan tien jaar geeft een trillend beeld.

Een kleurentelevisie is erg aan te bevelen. De Atari computers hebben een zeer bijzonder grafisch kleurensysteem dat 256 verschillende schakeringen kan vertonen. Veel programma's uit dit boek maken hiervan gebruik. Als u dit op een zwart-wit televisie moet bekijken mist u een heleboel.

Als u een aparte televisie voor uw computer kiest zult u er waarschijnlijk een met een klein of middelgroot beeldscherm nemen. Een beeldscherm van 25 tot 38 cm is voor de leesbaarheid ruimschoots voldoende. Grotere schermen zijn niet nodig. Vaak zijn ze zelfs slechter te lezen.

Atari 600XL	Dit boek is speciaal voor dit model geschreven. Er zijn geen aanpassingen nodig.
Atari 800 XL	In feite is dit model hetzelfde als de Atari 600XL met een groter geheugen. U kunt met dit boek werken alsof U een 600XL heeft.
Atari 1400XL in ontwikkeling	Als dit model op de markt komt is het te vergelijken met een Atari 800XL met een uitgebreider toetsenbord en enkele extra mogelijkheden. U kunt er dit boek voor gebruiken, maar de in deze uitvoering ingebouwde telefoonmodem en spraaksynthesizer worden hier niet beschreven.
Atari 1450XLD in ontwikkeling	Evenals model 1400XL, is dit model nog in ontwikkeling. Als het op de markt komt heeft het dezelfde mogelijkheden als model 1400XL met als extra een disk drive. Voor wat betreft de basisfuncties kunt u dit boek gebruiken.
Atari 400 wordt niet meer verkocht	Deze al oudere computer heeft dezelfde binnenstructuur als model 600XL. Door het platte toetsenbord is hij moeilijker te gebruiken dan de andere Atari computers. De beschrijvingen van zijn uiterlijk in hoofdstuk 3 en de geavanceerde grafische mogelijkheden in hoofdstuk 8 zijn iets anders. U MOET ER EEN ATARI BASIC CARTRIDGE BIJ GEBRUIKEN.
Atari 800 wordt niet meer verkocht	Deze oudere computer heeft dezelfde binnenstructuur als model 800 XL. De beschrijvingen van zijn uiterlijk in hoofdstuk 3 en de geavanceerde grafische mogelijkheden in hoofdstuk 8 zijn iets anders. U MOET ER EEN ATARI BASIC CARTRIDGE BIJ GEBRUIKEN.
Atari 1200XL wordt niet meer verkocht	Deze computer ziet en uit als model 1400XL, maar beschikt niet over het telefoonmodem, de spraaksynthesizer en de ingebouwde BASIC. Om hem te kunnen gebruiken, MOET U ER EEN ATARI BASIC CARTRIDGE BIJ HEBBEN. Overigens lijkt hij op de computers die in dit boek beschreven zijn.

Figuur 1.3: Hoe dit boek bij de diverse modellen gebruikt wordt.

Alle Atari's met uitzondering van de 600XL en het uit de productie genomen model 400 hebben de mogelijkheid om een *kleurenmonitor* in plaats van een televisietoestel aan te sluiten. Een monitor ziet eruit als een televisie maar heeft geen ontvangedeelte en kan daarom niet als gewone t.v. worden gebruikt. Een monitor geeft gewoonlijk een beter beeld dan een gewone televisie. Sommige monitors werken met een fijn raster (een fijnere detailweergave) zodat u met wat extra apparatuur meer dan 40 tekens op een regel kunt krijgen.

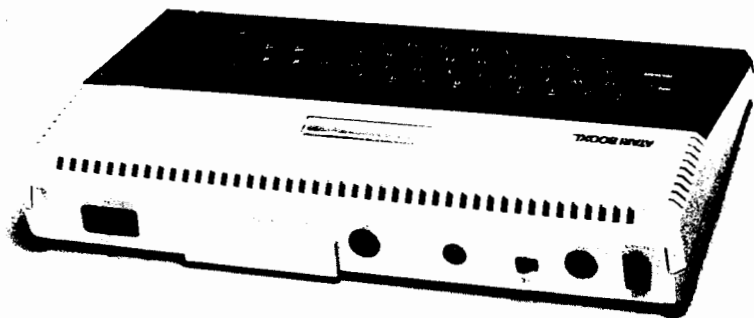
In de doos bevinden zich o. a. de gebruiksaanwijzing en diverse losse onderdelen. Haal de computer voorzichtig uit de doos en let erop dat u niets verliest. Drie belangrijke dingen moeten er in de doos zitten.

Ten eerste de computer zelf. Zet hem nadat u hem uit de doos hebt gehaald zo neer dat u de achterkant kunt zien zoals op figuur 1.4. Modellen die anders zijn dan model 600XL zien er bijna hetzelfde uit.

Ten tweede, de *adaptor*, die de 220 V omzet in de stroom die de computer gebruikt. Dit apparaatje heeft een voedingsdraad voor het stopcontact en een kabel die u achterin de computer moet steken.

Tenslotte ligt er nog een kort kabeltje in dat nodig is om de computer met de televisie te verbinden.

Heeft u gecontroleerd of alles aanwezig is, dan kunt u uw computer aansluiten. Voordat u begint moet u zich ervan overtuigen dat de



Figuur 1.4: De achterzijde van de Atari 800XL, waar de aansluitingen gemaakt moeten worden.

computer en de televisie uitgeschakeld zijn. Het kan geen kwaad als u de draden aanpakt, maar het is een goede gewoonte de stroom uit te schakelen voordat de verbinding gemaakt wordt.

Laat u niet van de wijs brengen door de vermelding POWER aan de rechter onderkant van het toetsenbord van de Atari 600XL en 800XL. Dit heeft betrekking op het rode indicatielampje dat aangeeft of de stroom aanstaat. De echte aan/uit schakelaar bevindt zich aan de linker achterkant van de computer. Bij de modellen 1200XL, 1400XL en 1450XL aan de linkerkant, naast de cartridge opening. Op de modellen 400 en de 800 bevindt zich de schakelaar aan de rechterkant. Bij de schakelaar staat altijd POWER of PWR ON/OFF.

We beginnen met de televisie. Trek de antenne uit. Als u centrale antenne hebt is uw televisie waarschijnlijk aangesloten met een ronde stekker. U kunt deze aansluiting gebruiken als de stekker van het meegeleverde kabeltje erin past. Als het niet past hebt u waarschijnlijk een adaptor nodig. Deze krijgt u in de meeste radio en televisiezaken.

Neem de zwarte kabel die meegeleverd is. Deze heeft een lengte van 1,80 m en heeft een ronde metalen stekker aan allebei de kanten. Steek een kant met de opening in de achterkant van de computer waar TV staat. Op alle nieuwe XL modellen bevindt zich deze opening van achteren gezien aan de rechter kant. Steek de andere stekker in de antenne-ingang van uw televisie.

Sluit nu de voeding aan. Bij alle Atari XL modellen steekt u de kabel van de voeding in de opening PWR IN, die zich vanaf de achterkant gezien, rechts bevindt. Daarna steekt u de stekker in het stopcontact. De verbindigen moeten zijn als getoond op figuur 1.5.

U kunt het signaal van de computer ontvangen op kanaal 2 of 3. U heeft geen last van storingen wanneer u een kanaal kiest dat nog niet door een televisiestation gebruikt wordt. Schakel de kanalenkiezer aan de achterkant van de computer in en stel dan de televisie af op hetzelfde kanaal.

Als u alles goed aangesloten en ingeschakeld hebt moet er een blauw scherm verschijnen waarop linksboven het woordje READY staat. Als het beeld een beetje wazig is moet u uw televisie beter afstellen.

Als het beeld niet donkerblauw is moet u de kleur bijstellen. Misschien is het beter de automatische kleurafstelling uit te schakelen omdat men bij sommige televisietoestellen dat met de hand beter kan doen.

Maak u niet ongerust als de kleur van het scherm na een paar minuten anders wordt. Atari computers regelen dat automatisch als ze een paar minuten niet worden bediend. Dit is erg handig en spaart uw televisie. Als de computer de kleur niet zou veranderen, zou het kunnen gebeuren dat de letters in het beeldscherm branden. Zogauw u op een toets drukt wordt het scherm weer blauw.

Als er geen afbeelding verschijnt of als het beeld niet duidelijk is kan het zijn dat u iets fout hebt gedaan. Het probleem zou een van de volgende oorzaken kunnen hebben.

- U bent vergeten de computer of de televisie aan te zetten of hebt vergeten de stekker in het stopcontact te steken.
- De kabel van de computer naar de televisie is niet of niet juist aangebracht. Druk ze nog eens goed in de openingen.
- U hebt de kabels vanaf de schakelaar naar het VHF-scherm niet of niet juist verbonden.
- De schakelaar staat niet op **COMPUTER**.
- Uw televisie staat niet op hetzelfde kanaal als uw computer (2 of 3). Controleer dit door het andere kanaal te kiezen.

Als u dit allemaal gecontroleerd hebt en uw computer doet het nog steeds niet dan moet u contact opnemen met uw leverancier. Misschien is de computer defect, of ontbreekt er iets. Uw Atari leverancier kan u advies geven en vervangt eventueel defecte apparatuur.

Randapparatuur

Met de televisie en de toebehoren die bij de computer geleverd zijn, hebt u alles wat u nodig hebt om dit boek te gebruiken. U kunt uw computer laten werken, BASIC programma's schrijven en programma's in cartridge gebruiken die door andere firma's worden verkocht.

Er is echter een verscheidenheid aan apparatuur die u aan uw computer kunt aansluiten, waardoor u meer mogelijkheden krijgt. U hebt deze apparatuur (*randapparaten*) niet perse nodig, maar sommige zijn erg handig.

Als u van plan bent spelletjes te spelen, wilt u misschien een *joystick* hebben. Dit is een hendel die u in verschillende richtingen kunt bewegen. Hiermee kunnen figuren op het scherm bestuurd worden. Er zit ook een rode *vuurknop* op die u bij sommige spelen nodig hebt om schoten af te lossen. Op uw Atari kunt u een tweede speelstuurknop aansluiten, zodat u met z'n tweeën kunt spelen.

Misschien wilt u een Atari Programma Recorder hebben. Hiermee kunt u programma's en gegevens *opslaan* op een cassettebandje, zodat ze niet verloren gaan als u de computer uitschakelt. De volgende keer kunt u ze weer in de computer *laden* om er opnieuw mee te werken. U hebt de recorder ook nodig om spelletjes mee te spelen en met programma's te werken die op cassettes worden verkocht.

Computer cassettebandjes zijn langzaam en omslachtig. Als u met uw computer niet alleen spelletjes wilt spelen, maar ook serieuze zaken wilt aanpakken kunt u beter een disk drive kopen. Hiermee kunt u snel een heleboel programma's en gegevens opslaan en er opnieuw mee werken. Een programma dat enkele minuten nodig heeft om van cassette gelezen te worden hebt u in enkele seconden van een diskette in de computer geladen. Sommige programma's voor gevorderden worden alleen op diskette verkocht.

De cassetterecorder en disk drive komen in de hoofdstukken 9 en 10 aan de orde. Misschien wilt u af en toe al eerder deze apparatuur gebruiken, bijvoorbeeld om een lang programma op te slaan. Op zo'n moment is het goed eerst deel drie door te lezen waarin het gebruik van deze extra apparatuur behandeld wordt. Op deze manier kunt u indien u een cassetterecorder of disk drive hebt het gebruik ervan leren, zonder dat degene die er geen hebben afgeleid worden.

U wilt misschien ook een printer (*drukker*) op uw systeem aansluiten. Hierdoor kan de computer getallen en tekst op papier afdrucken. Dit is erg handig als u een geschreven copie van uw programma wilt hebben of uw computer voor tekstverwerking gebruikt.

Atari heeft verschillende soorten printers. De goedkoopste is model 1020 (*printer/plotter*), die tekst en grafieken op een 10 cm brede rol afdrukt. Voor iets meer geld krijgt u een printer voor papier met het formaat A4. Model 1025 is een *matrix printer*, waarbij de letters uit een rij punten bestaan. Hij drukt snel, maar het resultaat is niet erg mooi. Een betere kwaliteit krijgt u met model 1027, een *daisy-wheel printer*, die tekens afdrukt vergelijkbaar met een gewone typemachine.

De programma recorder, disk drive en printer worden allemaal in dezelfde opening die zich aan de achterkant van de computer bevindt aangesloten, gekenmerkt met PERIPHERAL. Als u meer dan een randapparaat tegelijk aan wilt sluiten, kunt u ze aan elkaar koppelen. Breng een kabel aan tussen de PERIPHERAL opening in de computer en een van de openingen aan de achterkant van het eerste apparaat. Verbind dan met een kabel het tweede apparaat met de nog niet gebruikte opening aan de achterkant van het eerste apparaat. Op deze manier kunt u diverse verschillende randapparaten op uw computer aansluiten. Er is geen voorgeschreven volgorde. U moet er alleen op letten dat de programma recorder, altijd aan het einde van de keten moet staan.

Er is nog iets dat u met betrekking tot de serie randapparatuur van Atari moet weten. Toen Atari de nieuwe XL typen introduceerde, hebben ze ook de serie van randapparatuur opnieuw opgezet. Gelukkig heeft de firma er wel voor gezorgd dat de oude en nieuwe apparaten uitwisselbaar (*compatibel*) zijn. Alle oude randapparatuur kunt u op de nieuwe computers aansluiten en de nieuwe randapparatuur op de oude serie. Figuur 1.6 geeft een overzicht van de modellen die met

Randapparaat:	oud model	nieuw model
Programma recorder	410	1010
Disk drive	810	1050
Printer (10 cm rol)	820	1020
Printer (matrix)	825	1025
Printer (daisy-wheel)	—	1027

Figuur 1.5: De oude en nieuwe randapparatuur.

elkaar overeenstemmen. Wanneer u twijfelt of u uw apparatuur op elkaar kunt aansluiten kunt u voor alle zekerheid ook eerst uw leverancier vragen.

De voorbereidende werkzaamheden zijn nu afgesloten. U heeft uw computer geïnstalleerd en hij staat klaar voor gebruik. In het volgende hoofdstuk zult u hem gaan gebruiken. Veel succes en plezier.

Hoofdstuk 2

Commerciële Software

In het eerste hoofdstuk hebt u kennis gemaakt met uw Atari computer en de eigenschappen van het model dat u heeft. U hebt iets van de werking van de computer gezien en u weet nu ook dat u er niet bang voor hoeft te zijn. Tenslotte hebt u alles geïnstalleerd en aangezet.

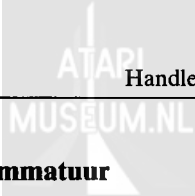
Op het scherm staat **READY** en u wilt nu eindelijk wel eens beginnen.

Het is het beste te beginnen met gekochte programma's die door anderen zijn geschreven. Vaak is het mogelijk gewoon het gekochte programma in de computer te steken en de rest gaat vanzelf. U hoeft dan niets over de werking van een computer te weten en u hoeft geen programmeerkennis te hebben.

Er zijn diverse soorten programmatuur in de handel gebracht door Atari als ook door andere fabrikanten. Spelletjes zijn het populairst, maar er zijn ook uitstekende programma's te verkrijgen op gebieden als onderwijs, eigen huishouding en muziek.

Dit hoofdstuk begint met een beschrijving van hetgeen u kunt kopen. Hoewel dit geen echte koopgids is, zult u er toch enkele tips over goede programma's in vinden. Ook komen er punten ter sprake waarop u bij het kopen moet letten, zodat u goede en slechte programma's van elkaar kunt onderscheiden.

Het tweede deel van dit hoofdstuk behandelt het gebruik van gekochte programma's. Bij de meeste programma's is er een instructieboekje, dat de details uitlegt, maar ik zal een algemene leidraad geven, om u op weg te helpen.



Het kopen van programmatuur

Atari zelf heeft meer dan 250 programma's voor hun computers gemaakt en zegt dat er meer dan 2000 langs andere kanalen beschikbaar zijn. Bij zoveel keus zult u wel geen moeilijkheden hebben iets geschikts voor uw behoefte te vinden.

Atari levert de programma's op cartridge, cassette en diskette. Spelletjes worden meestal op cartridge verkocht, terwijl de serieuze programma's hoofdzakelijk op cassette of diskette verkrijgbaar zijn.

Een cartridge bewaart het programma in een elektronische schakeling welke door de computer direct gelezen kan worden. Omdat ze direct in de computer worden gestoken en niet kunnen worden uitgewist, zijn ze over het algemeen handig en de levensduur is langer dan bij andere vormen van programmatuur.

Cassettes die al zijn opgenomen worden gebruikt in de Atari Program Recorder. Op cassettes wordt het programma gecodeerd als een wisselende hoge pieptoon. De computer kan deze informatie decoderen, opslaan en gebruiken. Cassettebandjes hebben een aantal belangrijke nadelen. Voordat u ze kunt gebruiken moeten ze in het computergeheugen worden geladen. De laadprocedure is vrij ingewikkeld, zelfs als u de aanwijzingen in hoofdstuk 9 volgt. Cassettes zijn langzaam en het duurt wel vijf minuten om een heel programma in te lezen. Verder gaat een cassette ook niet zo lang mee als een cartridge en ze kunnen op den duur door magnetische velden of mechanische slijtage onleesbaar worden.

Een disk drive is het meest bruikbaar. Deze werkt met schijven van 5 1/4 inch (13,5 cm) die men *diskettes* noemt. Ze zijn bedekt met een magnetisch oppervlak waarop grote hoeveelheden informatie kan worden opgeslagen. Ook diskette programma's moeten in de computer worden geladen voordat u ermee kunt werken, maar de procedure is sneller en betrouwbaarder. Van de meeste firma's waarvan u programmatuur op cassette kunt kopen krijgt u deze ook op diskette. De diskettes en disk drive worden in hoofdstuk 10 besproken.

Programma's voor uw Atari kunt u op de meeste plaatsen kopen waar ook de computers en video-spelen worden verkocht. Zelfs in warenhuizen kunt u software kopen vooral de populaire spelen. Tegenwoordig hebben ook diverse radio- en televisiezaken program-

matuur voor homecomputers. In grote steden zijn er gewoonlijk speciaalzaken voor programmatuur. Daar maakt u ook de meeste kans om de minder gangbare programma's te kunnen krijgen.

Atari's eigen programma's zijn over het algemeen erg goed. Atari heeft terecht een goede reputatie wat videospelen en huishoudelijke programma's betreft.

Ofschoon er bij Atari ook wel eens een middelmatig programma tussendoor slijpt zijn ze over het algemeen van zeer goede kwaliteit. Atari heeft ook een service die APX (Atari Program exchange) heet. Ze levert programma's die door gebruikers zijn gemaakt. De meeste zijn wel bruikbaar, maar zijn meestal niet foutloos. Meestal worden de APX programma's op diskette verkocht.

U kunt ook van andere firma's programma's kopen voor uw Atari computer. Atari heeft dit bewust aangemoedigd in de overtuiging dat een breed aanbod van programma's de verkoop van hun computers bevordert. Deze onafhankelijke maatschappijen heten *third-party fabrikanten*.

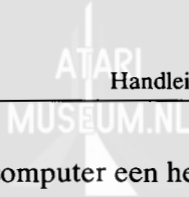
Als u uw programma's kant en klaar koopt zult u merken dat er tussen de echte goede programma's ook slechte zitten. In de zich snel ontwikkelende computerwereld zijn sommige maatschappijen in staat geweest hun inferieure produkten te verkopen door een mooie verpakking. Zelfs Atari heeft wel eens programma's op de markt gebracht die nog niet helemaal af waren. Omdat er meestal geen mogelijkheid is om programma's uit te proberen, is een verkeerde keuze snel gemaakt.

Er zijn verschillende manieren om u te oriënteren. Addison-Wesley heeft een lijst van Atari programma's gepubliceerd: *The Book of Atari Software*. Ondanks het feit dat het formaat onhandig is, geeft het boek een korte beschrijving en beoordeling van de meest verkrijgbare programma's.

Ook tijdschriften kunnen hulp bieden.

Spelen

Ik heb al voorgesteld met een spelletje te beginnen als dit de eerste keer is dat u met een computer in aanraking komt. Het serieuze werk kan later komen.

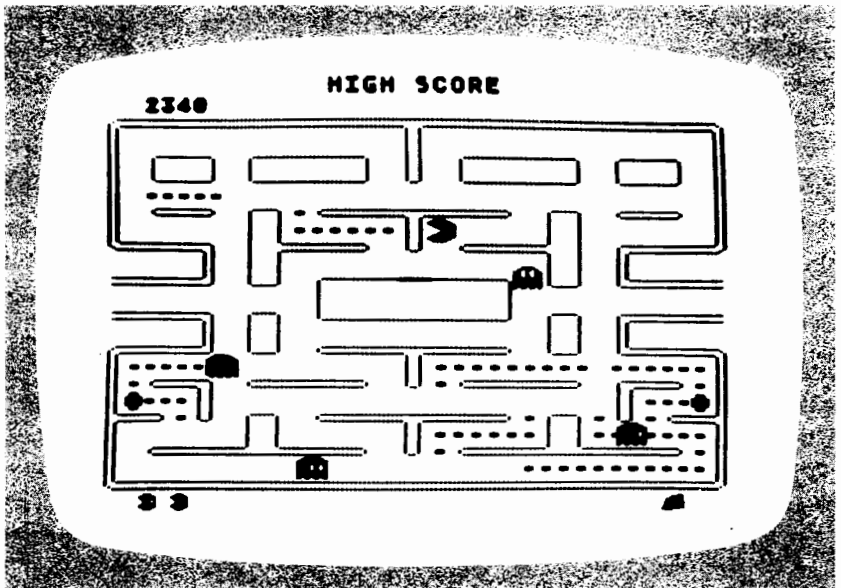


U kunt voor uw Atari computer een heleboel verschillende soorten spelletjes kopen.

Atari heeft een naam gekregen door hun uitstekende videospelen.

Meestal hebt u voor videospelen een joystick nodig. De joysticks van Atari zijn bij de meeste leveranciers te koop. Als u een fanatieke speler bent, wilt u misschien geld uitgeven voor een joystick voor spelletjes met versnelling, voor meer comfort en betere controle. Voor bepaalde spelletjes zou het goed zijn als u de beschikking zou hebben over een speciale *trackball* waarmee u een bal in een bepaalde richting kunt rollen.

Van de spelletjes met versnelling is van Atari Pac-Man het bekendste. Figuur 2.1 toont de homecomputer versie van het spel. Het is de bedoeling de kleine gele man door de doolhof te bewegen en daarbij alle stippen op te eten, zonder gepakt te worden door een van de vier rovende monsters. U kunt tijdelijk een veiliger heenkomen zoeken door een van de grotere 'energie' stippen in de vier hoekpunten op te



Figuur 2.1: Het Pac-Man spel

eten. Hierdoor kunt u op de monsters jagen en hun opeten in plaats van dat zij u pakken. Na een korte tijd moet u weer stippen eten en proberen de monsters te ontwijken.

Pac-Man is een goede koop. De Atari homecomputer versie is erg goed en heeft alle belangrijke eigenschappen van de videospelen in zich verenigd. Zelfs als u geen fanatieke video-speler bent, krijgt u bij Pac-Man u een goede demonstratie van wat uw computer allemaal kan.

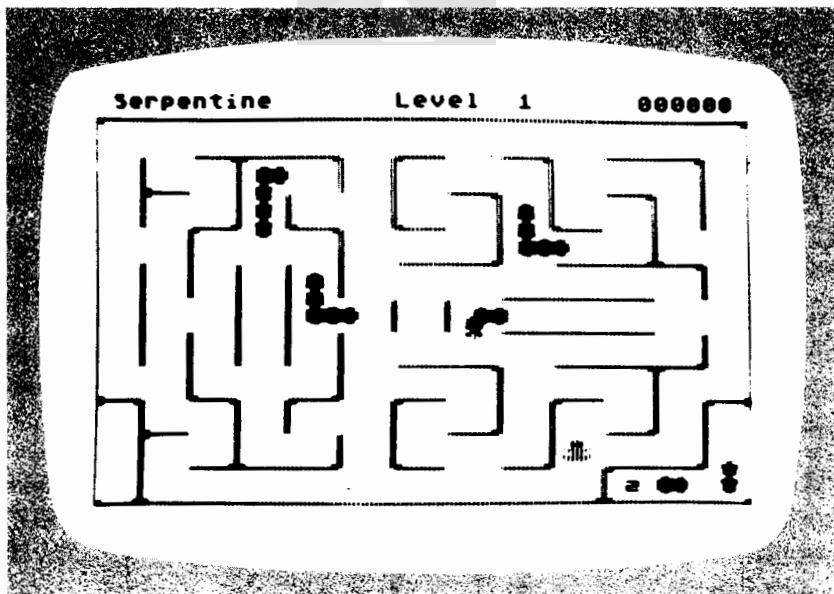
Atari verkoopt diverse andere videospelen voor hun homecomputers. Centipede en Missile Command zijn bekende titels uit de beeldrijke videospelen. In Star Raiders, een origineel Atari spel, bent u als piloot van een ruimteschip in gevecht met een vloot van vijandelijke aanvallers. U moet een strategie ontwerpen om uw brandstof en beschermingsschilden zoveel mogelijk te sparen. Als u zich in de lucht bevindt ziet u de sterrenhemel en de vijandelijke schepen voorbij gaan.

Van de spelen die door andere firma's voor Atari computers op de markt zijn gebracht is Serpentine van Brøderbund Software mijn lievelingsspelletje. In dit spel moet u een kleine slang door een doolhof manoeuvreren en proberen drie vijandelijke slangen die van achteren komen te pakken zonder dat zij u pakken. Brøderbund verkoopt ook het populaire spel Choplifter!, waarin u met een heli-copter achter de vijandelijke linies vliegt en 64 gijzelaars probeert te redden.

Maar er is nog meer dan alleen maar videospelen. Schaken is een van de interessantste spelen die u met een computer kunt doen. Atari's Computer Schaak cartridge is een heel aardig spel en is eenvoudig in het gebruik.

Veel mensen houden van avonturen-spelen. Anders dan bij videospelen zijn bij avonturenspeletjes gewoonlijk geen snelle acties en vlugge beschietingen nodig. In plaats hiervan wordt U in een denkbeeldige situatie geplaatst en moet u een schat zoeken of een probleem oplossen. De twee populairste zijn Zork, van Infocom, en Scott-Adams Adventure Series, van Adventure International.

Adventure speelt u door het intypen van korte commando's zoals bijvoorbeeld „climb stairs” (trap op) of „look under the table” (kijk on-



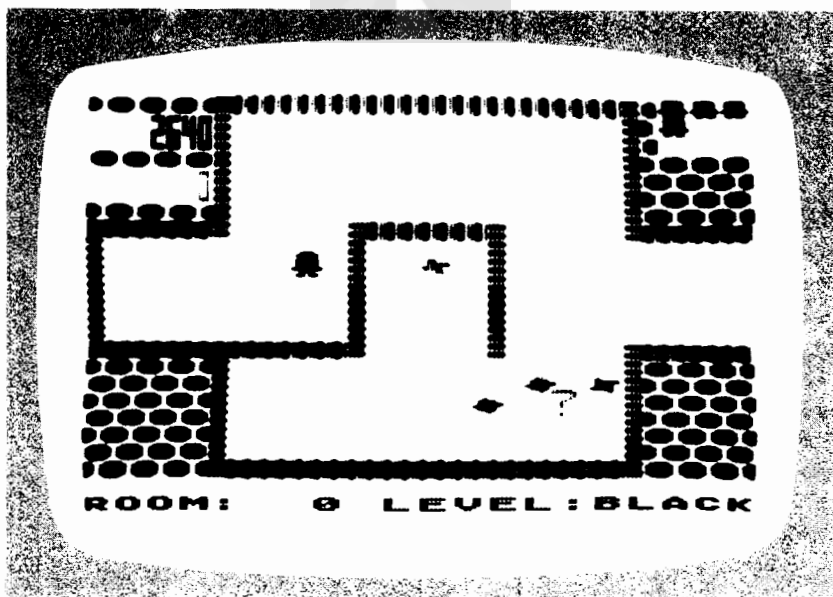
Figuur 2.2: Het Serpentine spel van Brøderbund Software

der de tafel). Als de computer u begrijpt, zal hij de opdracht uitvoeren en het resultaat meedelen. U moet nogal veel fantasie hebben om er achter te komen hoe u in deze denkbeeldige wereld een obstakel kunt passeren.

Nog een spelletje dat de moeite waard is is Shamus, van Synapse Software. Het is afgebeeld op figuur 2.3 en is een combinatie van het beste uit de video en de avonturen-spelen. In een onbekende kerker moet u naar sleutels en sloten zoeken om te kunnen ontsnappen. In elke ruimte moet u het opnemen tegen reusachtige monsters, door te schieten en uit te wijken als in een videospel. Hier zijn de snelle actie van een videospel en de voorzichtige strategie van een kerkeravontuur gecombineerd in een spel.

Serieuze programma's

Atari en andere firma's verkopen vele handige programma's die bij vele werkzaamheden van pas komen. Onderwijs, privé-financiën en



Figuur 2.3: Het Shamus spel van Synapse Software

tekstverwerking zijn slechts een paar onderwerpen waarvoor u uw computer kunt gebruiken.

Veel mensen kopen programma's om hun kinderen bij hun huiswerk te helpen. Een computerprogramma kan een kind individuele aandacht geven met kleurrijke, stimulerende en leerzame illustraties. Het controleert stap voor stap wat het kind heeft gedaan en zegt waar de fouten zitten. Sommige mensen vinden ook dat als een kind vroeg aan een computer went, het er later als de computer een belangrijke rol in zijn leven gaat spelen er minder moeite mee zal hebben.

Van de studieprogramma's verkrijgbaar voor de Atari computer geef ik de voorkeur aan die van Spinnaker Software. Van deze firma is Facemaker, ontwikkeld voor ongeschoolden, een van de meest onderhoudende programma's, goed in het aanleren van de basisvaardigheden van geheugen, herkenning en creativiteit. Met dit programma kan uw kind een gezicht op het scherm brengen door te kiezen uit een aantal kapsels, wenkbrouwen, neuzen en monden. Het kan ook een

geheugenspelletje spelen; het moet dan een steeds langere serie ge-laaitsuitdrukkingen onthouden. Spinnaker verkoopt diverse andere goede studieprogramma's, waaronder Kindercomp en Snooper Troops.

Als u uw kinderen wilt leren programmeren moet u de aanschaf van Atari's PILOT eens overwegen. Dit is een vereenvoudigde computertaal, waarmee kinderen de computer leren kennen door grafische ontwerpen op het scherm te maken. U kunt het als een insteek-cartridge kopen.

Gecompliceerde financiële berekeningen zijn een plaag voor vele huiseigenaren. Er bestaan een aantal computerprogramma's die berekeningen met betrekking tot bijvoorbeeld renteaflissingen, budgettering en het bijhouden van het huishoudboekje vereenvoudigen. U zult er versteld over staan hoeveel eenvoudiger u uw financiële problemen oplost.

Jammer genoeg vereisen de twee beste financiële programma's van Atari een disk drive. Het Home Filing Manager programma is een elektronische kaartenbak. Als u iets wilt weten dan kunt u dit op de gewenste manier in uw bestand opzoeken. U kunt er bijvoorbeeld gegevens uit halen met betrekking tot een bepaald telefoonnummer of een bepaalde naam. Het beeld dat het programma toont lijkt op een mooie kaartenbak met paperclip die de belangrijke gegevens aanstipt. Het Home Filing Manager programma is handig voor hoe dan ook ingerichte bestanden, zoals een adressenlijst of de cijferlijst van een onderwijzer.

VisiCalc, dat alleen werkt met een disk drive en een geheugen met ten minste 32 K is een andere mogelijkheid om uw financiën te beheren. Dit bekende programma is het eerste van de elektronische werkbladen (*spreadsheets*), die in de zakenwereld populair zijn. Het programma geeft u een grote vrije tabel met rijen en kolommen. U kunt waar u wilt getallen en tekst in de blokken van de tabel typen. Daarna kunt u de computer op deze blokken gecompliceerde handelingen en berekeningen laten verrichten. Het is een nuttig programma voor iedereen die op de hoogte moet blijven van veranderingen van getallen.

Ook het Atari Writer tekstverwerkend programma is erg handig. Met deze cartridge gedraagt uw computer zich als een intelligente typemachine. U kunt de tekst in het geheugen van de computer typen. Daarna kunt u hem corrigeren door woorden in te voegen, te verwijderen en tekstblokken te verplaatsen. Dit programma heeft een printer nodig en een disk drive of een programma recorder.

Er zijn nog meer programma's te koop om nog andere dingen met uw computer te doen. Muziekprogramma's, zoals de Atari Componist (Composer), geven de mogelijkheid melodieën te spelen via het geluidssysteem van de computer. Met Atari's verfprogramma (Paint) kunt u op het beeldscherm kleurrijke motieven tekenen.

Het gebruik van software

Er is niets bijzonders voor nodig om kant en klare programmatuur te kunnen gebruiken. U hoeft alleen uw computer aan te zetten, enkele instructies te geven en het programma te laten lopen. Als het programma informatie nodig heeft dan wordt dat gewoonlijk met een simpele vraag op het scherm aangegeven. De beste programma's zijn zo opgebouwd dat zij zelfs helpen uw gedachten te ordenen en u hulp geven als u die nodig hebt.

Met een cartridge hoeft u alleen maar het programma in de computer te steken en hem aan te zetten. Bij de Atari 600XL en 800XL moet u de cartridge in de grote opening stoppen aan de bovenkant van de computer, die bedekt is met zilverachtige klepjes. Bij de 1200XL en 1400XL schuift u de cartridge in de grote opening aan de linkerkant. Op de uit de productie genomen modellen 400 en 800 opent u de kleine klepjes die zich aan de bovenkant van de computer bevinden en steekt er dan de cartridge in.

Programma's op cassette of diskette moet u eerst in het computergeheugen laden voordat u ermee kunt werken. Verbind eerst de disk drive of de cassetterecorder met de computer, door de zwarte kabel in de PERIPHERAL aansluiting in de achterkant te steken. Bij een cassette moet u meestal de computer aanzetten en een instructie intypen om het programma te laden (LOAD). Met een diskette laadt de computer gewoonlijk het programma automatisch op het moment dat u de computer aanzet en de diskette zich in de disk drive bevindt.

De meeste softwarepakketten bevatten gedetailleerde instructies met betrekking tot het gebruik van het programma. Als u er geen hebt kijk dan eens naar hoofdstuk 9 en 10.

Als u de 600XL of een van de andere XL computers hebt kan het zijn dat u problemen hebt met enkele programma's die oorspronkelijk voor de oudere Atari computers geschreven zijn (de 400 en 800). Hoewel de meeste programma's uitwisselbaar zijn tussen de oude en nieuwe modellen, zijn er een paar die dat niet zijn. In dat geval hebben de softwareleveranciers meestal een aangepaste versie gemaakt die op alle computers past. Neem contact op met uw leverancier om het programma te ruilen.

Bij het gebruik van sommige oudere programma's op de XL modellen, moet u de OPTION toets ingedrukt houden als u de computer aanzet. Dit moet omdat de meeste oudere programma's ervan uitgaan dat de BASIC programmeertaal cartridge niet in de computer zit voordat u het programma laad. Daar BASIC bij de nieuwe computers is ingebouwd en dus niet verwijderd kan worden, moet u het met de OPTION toets uitschakelen.

Als u eenmaal het programma heeft geladen, gaat het vanzelf. De betere programma's geven volledige aanwijzingen op het scherm. U kunt meestal de instructieboeken raadplegen voor verdere uitleg.

Wat er nog komt

In dit hoofdstuk werd verteld hoe u programma's kunt gebruiken die door iemand anders geschreven zijn en verkocht worden. Het is een van de dingen die u zonder ervaring te hebben met uw Atari computer kunt doen. Door kant en klare software te gebruiken kunt u uw computer terwijl u er mee werkt leren kennen.

U kunt veel meer met uw computer doen wanneer u leert instructies te geven en zelf programma's te schrijven. U kunt dan uw computer zelf besturen zonder de weg te moeten gaan die een ander voor u uitgestippeld heeft.

De volgende twee hoofdstukken geven een korte beschrijving van de beginselen van het computergebruik. U leert het toetsenbord kennen en hoe u woorden op het scherm zet. U raakt met de beginselen



van het grafisch systeem van Atari vertrouwd zodat u ook figuren kunt tekenen.

In deel 2 behandelen we dan het echte programmeren. De concepten uit de hoofdstukken 3 en 4 worden hierin tot een groot geheel samengevoegd. Als u alle voorbeelden in u opneemt hebt u na de bestudering van deze handleiding uw computer goed leren kennen.

Hoofdstuk 3

Toetsenbord en beeldscherm

In dit hoofdstuk behandelen we het toetsenbord van de computer. Over het algemeen zult u het gebruiken om woorden en getallen in te typen zoals bij een typemachine. Er zijn echter een paar speciale toetsen waarvan u de functies moet kennen om de computer te kunnen bedienen. Ze worden op de volgende bladzijden uitgelegd.

Tegelijkertijd leert u hoe u met de tekst om moet gaan die de computer op het scherm zet. U zult merken dat het scherm twee functies heeft. Het reageert op het toetsenbord en geeft alles wat er ingetypt wordt weer, zodat u kunt zien wat u schrijft. Tevens laat het de antwoorden die de computer op uw instructies geeft zien.

In dit hoofdstuk beginnen we met een eenvoudige instructie die PRINT heet. Met deze instructie kunt u een regel, tekst of getallen op het scherm afdrukken. Naarmate u deze instructie meer en meer beheerst kunt u hiermee tekst neerzetten waar u maar wilt en ongewenste gegevens van het scherm verwijderen.

Dit lijkt misschien onbelangrijk. Waarom zou u zoveel moeite doen om de computer alleen maar het woordje HALLO te laten zeggen? Waarom zou u geïnteresseerd zijn in instructies om het scherm schoon te maken? Wacht maar eens af. Dit zijn de grondbeginselen voor de rest van het boek. Voordat u gecompliceerdere zaken kunt, moet u eerst weten hoe instructies in de computer ingetypt worden en hoe hij daarop reageert. Daarna bent u in staat interessantere dingen te doen.

Het toetsenbord

Het toetsenbord van de Atari is afgebeeld op figuur 3.1. Het lijkt veel op een normale typemachine met een paar toevoegingen. De letters zitten allemaal op dezelfde plaats; alleen enkele symbolen zitten er-

gens anders. Het is u misschien opgevallen dat er een paar onbekende toetsen aan de linker en rechterkant van het toetsenbord zitten, maar overigens ziet het er uit als een gewone typemachine.

Zet uw computer en televisie aan. Als alles goed is aangesloten dan is het scherm donkerblauw en in de linker bovenhoek staat het woordje **READY**. Als het scherm er niet zo uitziet moet u hoofdstuk 1 opnieuw lezen.

Het kleine witte rechthoekje onder de R van **READY** is de cursor. Als de computer letters van het toetsenbord accepteert, gebruikt hij het rechthoekje om de plaats van de volgende letter aan te geven. Typ een letter, welke maakt niet uit. Deze komt op de plaats van de cursor te staan en de cursor springt een plaatsje naar rechts.

Het toetsenbord heeft een paar bijzonderheden waar apart aandacht aan moet worden besteed. Ten eerste zijn er toetsen voor de cijfers 0 en 1. Bij een typemachine, bent u er misschien aan gewend hiervoor de letters 0 en I te gebruiken. Dat gaat bij een computer niet: U moet cijfers gebruiken als er van hem een getal verwacht wordt.

Automatische herhaling (auto-repeat) is een tweede afwijking. Als u een toets langer dan een seconde ingedrukt houdt, begint deze te repeteren. U kunt een serie Z's over het scherm typen door alleen maar de Z toets ingedrukt te houden. Het herhalen stopt wanneer u de toets loslaat. Dit is erg handig bij de spatiebalk als u naar het midden van een regel wilt.



Figuur 3.1: Het toetsenbord van de Atari 800XL.

Als u een typefout maakt kunt u deze corrigeren met de BACK SPACE toets, rechtsboven op het toetsenbord. Bij het indrukken van deze toets springt de cursor een plaatsje naar links en verwijderd de letter die daar stond. U kunt hierna de juiste letter intypen en doorgaan alsof er niets gebeurd is.

U zult gemerkt hebben dat alle getypte letters als hoofdletters worden weergegeven. Atari computers kunnen ook kleine letters weergeven, maar bij instructies vereist hij hoofdletters. Daarom begint de computer met hoofdletters.

Als u ook kleine letters wilt typen, moet u de CAPS-toets aan de rechterkant van het toetsenbord indrukken. De computer geeft de ingetypte letters nu als kleine letter weer. Zoals bij een typemachine kunt u ook afzonderlijke letters als hoofdletter schrijven. Door een van de SHIFT toetsen te gebruiken. Om terug te keren naar hoofdletters moet u de CAPS-toets nogmaals indrukken. (Op de Atari 400 moet u de SHIFT- en CAPS-toets tegelijk indrukken om terug te keren naar hoofdletters.) De SHIFT-toets geeft de CAPS stand niet 'vrij', zoals op een typemachine.

Om de symbolen te typen die op de bovenste helft van de niet-letter toetsen staan, moet u de SHIFT-toets gebruiken, *zelfs wanneer hij in de CAPS stand staat*. Tot deze symbolen behoren ook het uitroepteken (!) en de aanhalingstekens (").

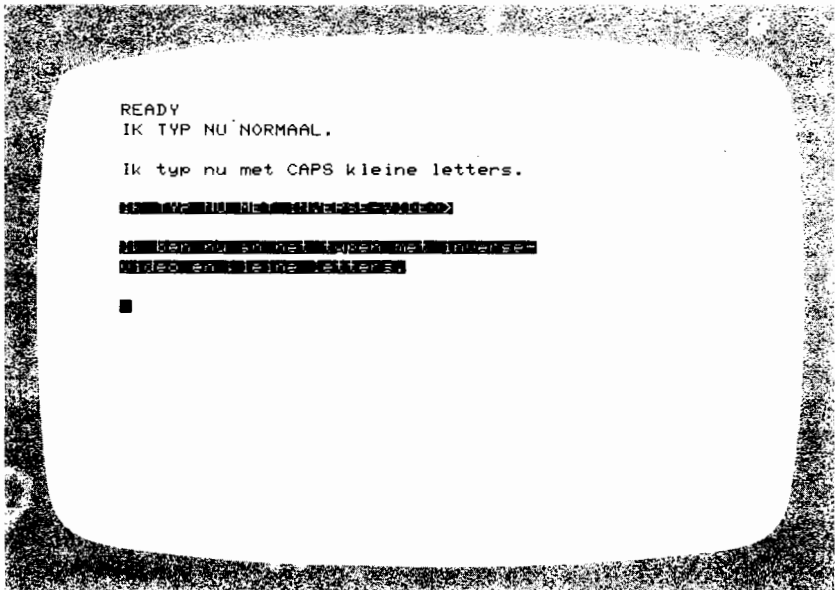
U kunt een andere speciale toets gebruiken om een interessant effect te krijgen. Atari computers hebben een *inverse-video* toets, die de wijze waarop de letters op het scherm komen verandert. Deze toets, die er uitziet als een vierkantje met een diagonaal (■), bevindt zich rechtsonder naast de SHIFT-toets op de Atari 600XL en 800XL. Op de 1200XL en de in ontwikkeling zijnde 1400XL en 1450XL zit de inverse-video toets naast de BREAK-toets tussen de rij van zilverachtige toetsen aan de bovenkant. Op de oudere Atari modellen 400 en 800 staat op deze toets het Atari teken (A) en bevindt zich rechtsonder naast de SHIFT-toets.

Druk eens op de inverse-videotoets en typ enkele letters. Deze verschijnen als blauwe letters in een witte rechthoek op het scherm. Tijdens het typen groeit de rechthoek uit tot een geheel. Tijdens het werken met de inverse-video stand kunt u typen als op de gewone

stand en op de gebruikelijke manier wisselen tussen hoofdletters en kleine letters. Terugschakelen naar het typen van wit op blauw doet u door de inverse-video toets nogmaals in te drukken.

Figuur 3.2 toont de vier verschillende mogelijkheden hoe de tekst op het scherm verschijnt bij gebruik van de combinaties van de inverse-video en CAPS-toetsen. De eerste regel is direct na het aanzetten getypt. Alle letters zijn hoofdletters en in normaal beeld. Voor de tweede regel is de CAPS-toets eenmaal ingedrukt. De hoofdletters die nog in de regel voorkomen zijn geschreven met de SHIFT-toets. Voor de derde regel zijn de inverse-video en CAPS-toetsen ingedrukt, om te laten zien hoe de tekst eruit ziet als u de computer direct nadat u hem hebt aangezet op inverse-video zet. Voor de vierde regel is het scherm weer met de CAPS-toets op de kleine letterstand gezet, maar is tevens op de inverse-video blijven staan.

Figuur 3.2 laat nog een belangrijk aspect van de Atari computer en het toetsenbord zien. Het vierde voorbeeld is langer dan de 38 tekens



Figuur 3.2: Kleine letters en inverse-video tekst op het beeld.

die er op een regel passen. In plaats van buiten het beeld te raken als een regel te lang is, springt de cursor automatisch naar de volgende regel. Dit is een voordeel waardoor het mogelijk wordt instructies te typen die langer zijn dan een regel. Soms is het echter onoverzichtelijk omdat een woord wel eens in het midden wordt afgekapt.

Als u gewend bent op een elektrische typemachine te schrijven zult u geneigd zijn de RETURN-toets te gebruiken aan het einde van de regel. Op de computer heeft de RETURN-toets echter een speciale functie: hij geeft de regel als een instructie door aan de computer. Tot nu hebt u nog niets over instructies geleerd en moet u deze toets niet gebruiken. Als u het toch doet schrijft de computer het woordje ERROR (fout) om te laten zien dat hij het er niet mee eens is.

U kunt zo lang de BREAK-toets gebruiken in plaats van de RETURN-toets om een regel te beëindigen. Hij zegt de computer naar het begin van de volgende regel te gaan, zonder op de ingetypte regel te reageren. Op de meeste Atari computers bevindt zich de BREAK-toets rechtsboven op het toetsenbord. Op de 1200XL en 1400XL zit hij aan de rechterkant van de zilverachtige toetsen die zich aan de bovenkant van het toetsenbord bevinden.

Cursorbesturing

U hebt de getypte letters op het scherm zien komen. De letters verschijnen van links naar rechts en de cursor schuift op naar rechts en geeft aan waar u op het scherm bezig bent.

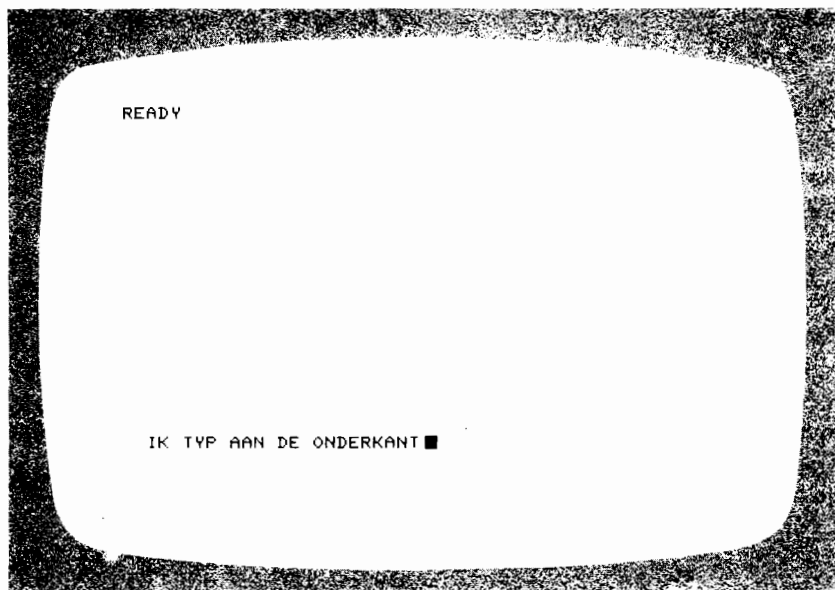
U kunt meer dan alleen maar van links-naar-rechts gaan. U zult vaak de opmaak (editing) instructies moeten gebruiken om dat wat er op het scherm moet gebeuren te besturen. Door bepaalde toetsen en toetscombinaties te gebruiken kunt u de cursor naar iedere plaats op het scherm sturen die u wilt en zelfs letters corrigeren die u al getypt hebt.

Zoals u weet, geeft de cursor de plaats aan waar de volgende letter komt. Als u de cursor verplaatst kunt u de volgende letter daar typen. Als u doorgaat met typen komen de letters weer op een rij van links naar rechts beginnend, op de plaats waar u begon.

Om de cursor te kunnen besturen moet u de CONTROL-toets, die vlak boven de SHIFT-toets aan de linkerkant van het toetsenbord zit, gebruiken. De CONTROL-toets heeft geen betekenis op zich. Hij wordt altijd in combinatie met een andere toets gebruikt. Om deze speciale instructies te geven moet CONTROL ingedrukt zijn. Zolang hij ingedrukt is interpreteert de computer de toetsen als instructies in plaats van letters. Om een instructie te geven moet u de gewenste toets indrukken.

(Als u de CONTROL-toets gebruikt samen met een van de lettertoetsen verschijnt er een *grafisch teken* op het scherm. Het kan gebruikt worden om figuren op het scherm te tekenen, maar dat is niet eenvoudig. In dit boek worden de grafische tekens niet behandeld, maar als u wilt kunt u ze wel gebruiken.)

De belangrijkste van deze speciale instructies zijn de vier pijltoetsen die zich rechts van de letters op het toetsenbord bevinden. Met deze pijlen kunt u de cursor over het scherm verplaatsen. De pijl naar



Figuur 3.3: De pijltoetsen maken het mogelijk op andere plaatsen op het scherm te typen.

links bijvoorbeeld, zet de cursor een plaatsje terug; de pijl naar beneden beweegt de cursor een regel recht naar beneden. Bij het verplaatsen van de cursor met de pijltoetsen, gaat hij over de letters zonder ze te veranderen; de letters hebben alleen een hogere intensiteit om aan te geven waar de cursor zich bevindt. U kunt de cursor snel verplaatsen door de pijltoetsen ingedrukt te houden. Als u de cursor van het scherm af probeert te krijgen zal hij aan de andere kant weer tevoorschijn komen.

De pijl naar links, op de toets met het plusteken, heeft de meeste mogelijkheden van de vier. Om hem te gebruiken moet u de CONTROL-toets ingedrukt houden. U merkt dat de plus-toets drie verschillende functies heeft. Alleen ingedrukt is het een plusteken (+). Ingedrukt samen met de SHIFT-toets is het de schuine streep-terug (back slash \). Ingedrukt met de CONTROL-toets springt de cursor een plaatsje terug, zoals hierboven al werd uitgelegd.

Met de pijltoetsen kunt u de cursor over het gehele beeldscherm verplaatsen. Zo kunt u bijvoorbeeld de pijltoetsen gebruiken om op de onderkant van het scherm een boodschap te typen (zie figuur 3.3).

Om te zien hoe de pijltoetsen werken moet u de volgende regel typen:

IN DEZE REGEL STAAT EEN FOETJE

Druk samen met de CONTROL-toets vier keer op de pijl naar links zodat de cursor op de E komt te staan. Laat dan de controltoets los en typ een U. De U komt in plaats van de E en het woord is gecorrigeerd.

Deze manier van corrigeren is erg handig. Iederen maakt weleens een fout. U kunt, dat klopt, de regel ook helemaal opnieuw intypen, maar meestal is het eenvoudiger de cursor terug te bewegen en de fout te corrigeren.

En wat moet u doen als u een letter vergeten bent? Veronderstel dat u getypt hebt:

IN DEZE REGEL STAT EEN FOUTJE

Als u slechts teruggaat en de letter die ontbreekt intypt, komt deze op het scherm, maar vervangt tevens een andere letter. Dit is te voorkomen met de INSERT instructie op de > toets die zich in de rechterbovenhoek van het toetsenbord bevindt. Verplaats de cursor met de 'pijl naar links toets', totdat hij op de A staat. Druk dan op de CONTROL-toets en sla INSERT aan. De A en de volgende letters schuiven een plaatsje naar rechts, en er ontstaat een spatie die door de cursor wordt bedekt. U kunt nu de A intypen en de correctie is gemaakt. Als u meer dan een letter moet toevoegen moet u de INSERT-toets voor iedere letter aanslaan.

U hebt de DELETE/BACK SPACE toets al gebruikt om fouten te corrigeren. Deze toets heeft in combinatie met CONTROL of SHIFT nog twee bijzondere betekenissen, waarmee u op een andere manier letters kunt verwijderen.

Om letters in het midden van een regel te verwijderen zonder dat er een ruimte ontstaat, moet u de DELETE/BACK SPACE samen met de CONTROL-toets gebruiken. Deze combinatie verwijdert de letters die door de cursor bedekt worden en sluit de ruimte die op de regel is ontstaan. Om te zien hoe dat werkt typen we nu:

IN DEZE REGEL STAAAT EEN FOUTJE

Ga nu terug met de cursor met behulp van CONTROL en de 'pijl naar links toets', totdat deze op een van de A's in STAAAT staat. Houd de CONTROL-toets ingedrukt en sla DELETE aan. De overbodige A verdwijnt en „AT” schuift op naar links om de ruimte op te vullen. Let op het verschil tussen deze en de gewone BACK SPACE instructie: BACK SPACE zet de cursor een plaatsje terug en verwijdert de linkerletter door tegelijkertijd een ruimte in de regel achter te laten.

Als u SHIFT in plaats van CONTROL samen met de DELETE/BACK SPACE-toets gebruikt, verwijdert u de gehele regel waarop de cursor staat. Dit is erg handig om overbodige regels van het scherm te halen. U moet echter wel goed opletten dat u niet met SHIFT-DELETE een hele regel verwijdert, wanneer u CONTROL-DELETE of alleen maar BACK SPACE bedoelt.

Aan de andere kant van de INSERT-toets bevindt zich een toets waarop CLEAR staat. Hiermee kunt u het gehele scherm schoonve-

gen. Om deze functie te gebruiken moet u de CONTROL-toets ingedrukt houden en dan CLEAR aanslaan. Alle letters worden van het scherm gewist en de cursor springt naar linksboven. U kunt deze instructie gebruiken om schoon schip te maken als naar uw mening het scherm een warboel is geworden. Bij sommige voorbeelden in het boek zal ik u vragen CONTROL-CLEAR te gebruiken voordat u een instructie geeft. Als u dit niet doet ziet het scherm er niet zo uit als ik het beschrijf.

Figuur 3.4 geeft een overzicht van de zes opmaaktoetsen. Oefen hiermee bij het doorwerken van de volgende bladzijden. U zult snel en eenvoudig overal op het scherm fouten kunnen corrigeren.

We hebben nu de belangrijkste toetsen van het Atari toetsenbord behandeld. We kunnen nu teksten typen, de cursor verplaatsen en correcties op het scherm maken, door gebruik te maken van de Atari opmaaktoetsen. We hebben de RETURN toets nog overgeslagen omdat deze een speciale functie heeft. Dat gaan we nu bekijken.

TOETS	FUNCTIE
alleen BACK SPACE	Wist het laatste teken en u kunt eroverheen typen.
CONTROL + pijltoets	Verplaatst de cursor.
CONTROL + INSERT	Voegt een spatie in een regel in.
CONTROL + DELETE/ BACK SPACE	Verwijdert een spatie uit een regel.
SHIFT + DELETE/ BACK SPACE	Veegt een hele regel uit.
CONTROL + CLEAR	Veegt het hele scherm schoon.

Figuur 3.4: De toetsen om het scherm op te maken.

RETURN en de foutmelding

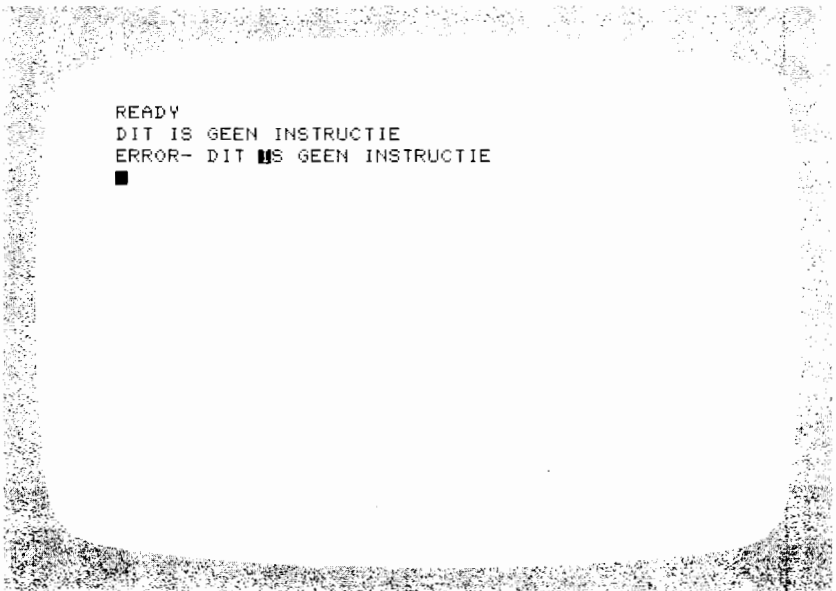
Typ de eerste regel van de tekst uit figuur 3.5. Druk nu op de RETURN-toets. Er verschijnt een ERROR (fout) melding, op het scherm.

Wat gebeurt er? De RETURN toets heeft een speciale functie. Hij laat de computer de ingetypte regel als een instructie lezen. Als de computer de instructie begrijpt wordt ze uitgevoerd. Begrijpt hij de instructie niet dan gaat hij ervan uit dat u iets fout gedaan heeft en drukt de foutmelding af (ERROR message).

U kunt de zin wel juist vinden. Daar ben ik het mee eens, maar de computer niet. Computers eisen dat u alles precies doet zoals zij dat willen. De meeste Engelse zinnen (en Nederlandse) passen niet in het taalgebruik van de computer. Verderop in dit hoofdstuk leert u enkele instructies die voor de computer een betekenis hebben.

Laat u door het woordje ERROR (fout) niet van de kaart brengen. Het is voor de computer een uiting van onbegrip. In dit voorbeeld, hebben we iets ingetypt dat voor ons wel zin heeft, maar voor de computer niet. De fout ligt bij hem niet bij ons.

Fouten behoren tot de normale gang van zaken bij een computercursus. U hoeft zich er niet voor te schamen. Uw computer leest alles no-



Figuur 3.5: U krijgt een foutmelding, als de computer uw instructie niet begrijpt.

gal *letterlijk* en helemaal niet verdraagzaam. Hij kan alleen begrijpen wat u bedoelt als u het precies zo zegt zoals hij dat verwacht. Als u ook maar één komma overslaat, kan hij u niet begrijpen.

Meestal helpt de computer u bij het zoeken naar de fout, door de eerste letter die hij niet begrijpt in inverse-video op het beeld te zetten. Uw fout bevindt zich meestal links van dit teken. Soms geeft de computer een *foutcode* (*error code*) zoals deze:

ERROR 3

Als u zo'n bericht krijgt, kunt u dat in bijlage C van dit boek opzoeken. Het codenummer 3 betekent een *waarde ongeldig*, dit betekent dat u ergens in uw instructie een ongeldig getal hebt gebruikt.

Als u de regels uit dit boek overtypt moet u er goed op letten dat u ze heel nauwkeurig overtypt tot en met het laatste aanhalingsteken. Als u nadat u een regel ingetypt hebt een foutmelding krijgt moet u hem letter voor letter controleren met wat er in het boek staat: misschien heeft u een woord verkeerd gespeld of een leesteken vergeten.

U moet er tevens op letten dat bij het intypen van een instructie de computer in de hoofdletterstand en de normaal wit-op-blauw-stand staat. Verschijnen de letters als kleine letters of in inverse video op het scherm dan zal de computer ze niet als een instructie interpreteren. Er is maar een uitzondering op deze regel en die bespreken we hierna. Als u niet zeker weet of de computer in de goede stand staat gebruik dan de RESET-toets of zet de computer uit en weer aan.

Uw eerste instructie: PRINT

Tot nu toe heeft u alleen maar letters en woorden op verschillende plaatsen op het scherm getypt en uw computer als een kladblok gebruikt. Zo kunt u het toetsenbord leren kennen, maar u leert niet echt uw computer te besturen. Hiervoor moet u een instructie gebruiken die de computer kent.

In dit deel leert u uw eerste instructie: de PRINT (druk af) instructie. Hiermee kunt u een serie letters, getallen of zelfs de uitkomsten van een berekening op het scherm brengen. Op het ogenblik gebruiken

we het alleen om de computer woorden en zinnen te laten uittypen, maar later kunt u het op vele andere manieren gebruiken. Door met deze eenvoudige voorbeelden met PRINT te oefenen, leert u tevens wat een instructie is en de uitwerking ervan op de computer.

Om een instructie te geven, moet u een regel op het scherm typen en RETURN indrukken. De computer leest dan de regel die u ingetypt hebt en probeert de instructies op te volgen. Als de computer de regel niet begrijpt geeft hij een foutmelding.

Typ de volgende regel nauwkeurig over

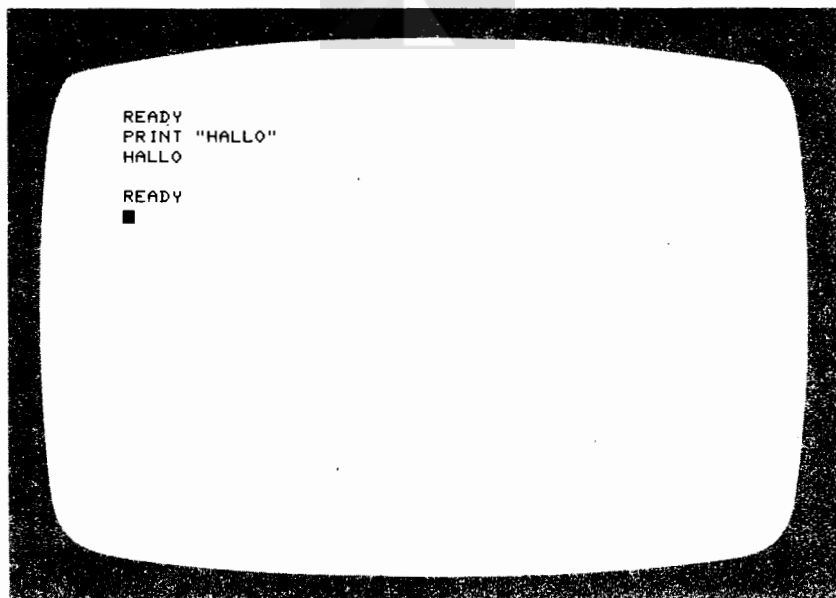
PRINT "HALLO"

Druk hierna de RETURN toets in. De computer moet reageren met HALLO, zoals op figuur 3.6. (Als u een ERROR kreeg na het indrukken van de RETURN toets, heeft u waarschijnlijk een typefout gemaakt. Controleer of u PRINT goed gespeld hebt en let op de aanhalingstekens voor en na HALLO.)

U hebt uw eerste instructie gegeven! De computer herkende PRINT als een *sleutelwoord* (*keyword*) met een speciale betekenis. Als de computer dit woord tegenkomt weet hij dat hij de rest van de regel moet doorzoeken en de boodschap tussen de aanhalingstekens afdrukken. Het woordje HALLO wordt gevonden, dus dit is de tekst die afgedrukt wordt. Hebt u gezien dat de computer de aanhalingstekens niet overtypet? Die betekenen dat het om het woordje HALLO gaat. Nadat de boodschap afgebeeld is, voegt de computer het woord READY (klaar) toe, om aan te geven dat de uitvoering ten einde is en er een nieuwe instructie ontvangen kan worden.

De resultaten mogen dan niet indrukwekkend lijken, u hebt een belangrijke stap gezet met uw eerste instructie. U hebt de computer gevraagd iets te doen, op een wijze die hij kon begrijpen. Hij heeft uw instructie gelezen en de resultaten op het scherm afgedrukt. Deze stap was het begin van de beheersing van uw computer.

U kunt diverse variaties op deze simpele PRINT opdracht proberen. U mag alles wat u maar wilt tussen de aanhalingstekens plaatsen en het wordt precies zo afgedrukt als u het intypte. U kunt ook kleine letters en inverse-video tussen de aanhalingstekens gebruiken (maar



Figuur 3.6: Hoe de computer op uw eerste instructie reageert.

nergens anders in de instructie). Ook spaties die tussen de aanhalingstekens staan worden door de computer weergegeven. Figuur 3.7 toont enkele mogelijkheden.

Als u variaties op de basisinstructie PRINT intypt, kunt u tevens met de opmaaktoetsen oefenen. Gebruik CONTROL-↑ om de cursor op een regel met een PRINT opdracht te zetten. Verander iets tussen de aanhalingstekens of voeg iets toe en druk op RETURN. Uw gewijzigde instructie wordt opnieuw uitgevoerd en de computer typt de nieuwe tekst over de oude.

Dit toont aan hoe handig het beeldscherm is. Iedere regel die er nog op staat is 'levend' en kan opnieuw gebruikt worden. Het enige dat u moet doen is de cursor terugbrengen naar de regel waarin u iets wilt veranderen, de wijziging doorvoeren en op RETURN drukken - de cursor mag zich overal op de regel bevinden. De computer zal de regel exact uitvoeren zoals deze op het scherm staat. Alle veranderingen die u maakt worden verwerkt.

```

READY
PRINT "U KUNT ALLES AFDrukKEN"
U KUNT ALLES AFDrukKEN

READY
PRINT "U kunt in kleine letters afdruk
ken"
U kunt in kleine letters afdrukken

READY
PRINT "IN IN WERKSTEL AFDrukKEN "
IN IN WERKSTEL AFDrukKEN

READY
PRINT "Als U wilt, kunt U zelfs op"
PRINT "twee regels afdrukken"
Als U wilt, kunt U zelfs op
twee regels afdrukken

READY
PRINT

READY
■

```

Figuur 3.7: Enkele voorbeelden van de PRINT instructie.

Dit is vooral handig als u een ERROR krijgt bij een lange instructie. In plaats van alles over te typen brengt u de cursor naar de fout en corrigeert deze. Daarna kunt u de RETURN-toets indrukken en de regel opnieuw gebruiken.

De vierde PRINT opdracht op figuur 3.7 gaat over het einde van een regel. Dat lukt, zolang u de gehele opdracht als een geheel intypt, zonder op BREAK of RETURN te drukken. Als uw boodschap nog niet af is als de cursor de rechterkant van het scherm bereikt moet u gewoon doortypen: de computer springt automatisch naar de volgende regel. U kunt instructies intypen die drie regels lang zijn. De computer waarschuwt u met een piepsignaal als u die grens bereikt hebt.

De laatste regel van figuur 3.7 toont een belangrijk aspect van de PRINT instructie. Als u PRINT zonder een mededeling erachter typt, drukt de computer een lege regel op het scherm (in feite twee, als u die meetelt die automatisch boven het woord READY staat).

Het is mogelijk meer dan één instructie tegelijk te geven. In plaats van RETURN aan het einde in te drukken, kunt u een dubbelepunt (:) typen en daarna de tweede instructie. Op die manier kunt u zoveel instructies geven als er op drie regels passen. Als u klaar bent en op RETURN drukt, worden alle instructies in de aangegeven volgorde uitgevoerd.

Figuur 3.8 toont hoe u lege regels en samengestelde instructies kunt gebruiken om de verschijning op het beeld te verbeteren. De instructie begint met drie simpele PRINT opdrachten, zonder tekst, om drie blanco regels op het scherm te krijgen. De vierde PRINT opdracht begint met een paar spaties zodat de tekst gecentreerd op de regel komt. De vijfde PRINT voegt een tweede regel toe, die met enige spaties van links begint. Een laatste PRINT zorgt voor een extra blanco regel op het scherm tussen de boodschap en het woordje READY.

De PRINT opdracht is een van de belangrijkste instructies die u leert. Het maakt niet uit hoe diepgaand u met uw computer experimenteert, u moet altijd uw resultaten op het scherm weergeven. Dit lukt met de PRINT opdracht op de door u gewenste manier.

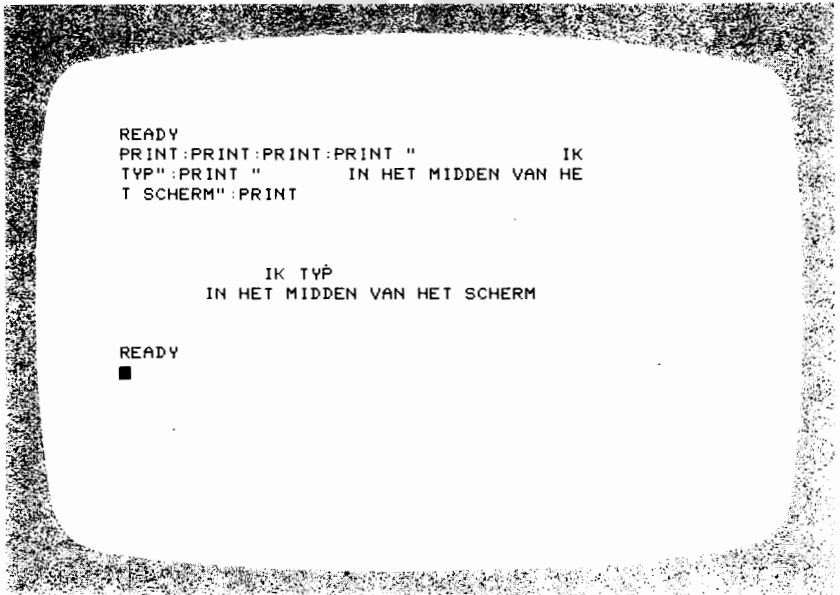
De ESC-toets

Zoals we nu hebben gewerkt, blijft de PRINT opdracht op het scherm staan terwijl de boodschap wordt afgebeeld. Stel dat u alleen de boodschap op het scherm wilt hebben. U heeft misschien de neiging de CONTROL-CLEAR toets te gebruiken als deel van de tekst, zoals:

```
PRINT "(CONTROL-CLEAR) HALLO"
```

Jammer genoeg lukt dat niet. De computer maakt het scherm schoon direct nadat de CONTROL-CLEAR toets wordt ingedrukt en u krijgt niet de kans de instructie met de boodschap HALLO af te maken.

Wat we willen is een manier om de CONTROL-CLEAR functie op te slaan als deel van de afgedrukte boodschap. Als dit lukt, dan moet de computer wachten met het scherm schoon te maken totdat u de in-

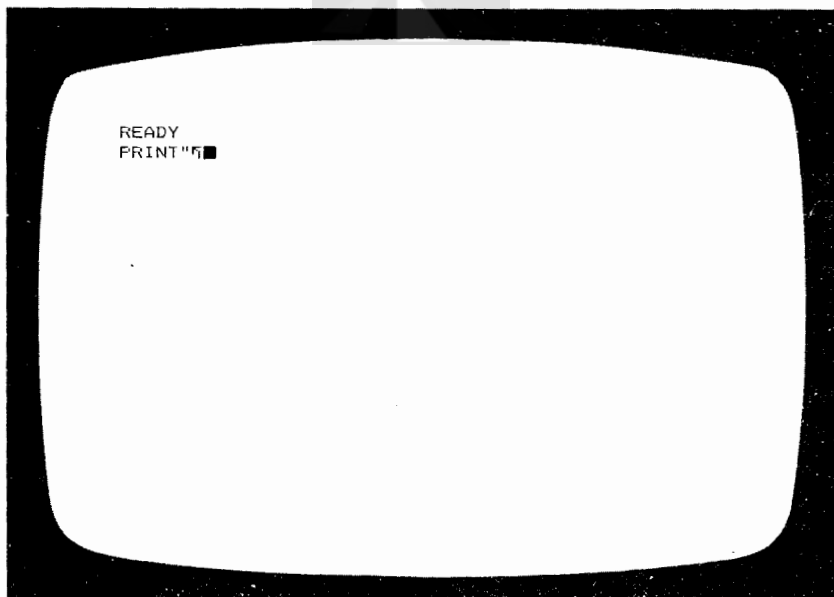


Figuur 3.8: Blanco regels en samengestelde PRINT opdrachten zorgen voor een overzichtelijk scherm.

structie af hebt. Daarna, bij het indrukken van de RETURN toets, moet hij de opgeslagen CONTROL-CLEAR opdracht uitvoeren vlak voor de boodschap wordt gedrukt.

Hiervoor moet u de ESC-toets gebruiken (die zich linksboven op het toetsenbord bevindt). ESC staat voor „escape” (ontsnapping) en wordt gebruikt om het effect van het volgende besturingsteken (control character) in de PRINT opdracht uit te stellen.

We zullen eens proberen met de ESC-toets in een PRINT opdracht het woordje HALLO op een leeg scherm af te drukken. Typ het woord PRINT en een openings-aanhalingsteken. Druk hierna eenmaal op de ESC-toets. Het lijkt of er niets gebeurt omdat de ESC-toets zelf niet afgebeeld wordt. Druk nu op CONTROL-CLEAR. In plaats van dat het scherm wordt schoongewist, drukt de computer een gebogen pijl in een witte rechthoek. Dit symbool toont dat CONTROL-CLEAR is opgeslagen als deel van de regel en in werking moet treden als de boodschap wordt afgedrukt. Eindig de boodschap



Figuur 3.9: Het gebruik van de ESC-toets voor een besturingsteken in een afgedrukte regel.

met het typen van het woordje HALLO en de sluitings-aanhalingstekens, zodat de instructie er uitziet als op figuur 3.9.

Druk nu op de RETURN-toets. De computer drukt nu de boodschap HALLO af, maar maakt eerst het scherm schoon. Figuur 3.10 toont het resultaat. Deze groet is beter te lezen dan die op figuur 3.6, waar de PRINT opdracht op het scherm bleef.

CONTROL-CLEAR is in feite een teken geworden, dat deel uitmaakt van de boodschap. Wanneer de computer het opgeslagen teken afdruckt (PRINT) reageert hij hetzelfde als wanneer u het zou hebben intypt op het punt waar het in de boodschap staat. Alle woorden in de boodschap na CONTROL-CLEAR worden afgedrukt nadat het scherm schoon is gemaakt, niet ervoor.

Hebt u gemerkt dat u ESC-CONTROL-CLEAR moet typen als deel van de boodschap tussen de aanhalingstekens. Als u ESC-CONTROL-CLEAR ergens buiten de aanhalingstekens typt krijgt u een foutmelding.

De ESC-toets kan gebruikt worden om het resultaat in een PRINT opdracht van iedere besturingstoets (CONTROL-toets) uit te stellen. Door combinaties van de vier pijltoetsen op te slaan kunt u de computer bijvoorbeeld overal op het scherm tekst neer laten zetten. Elke keer als de computer een opgeslagen pijltoets ontmoet, zal hij een stap in die richting maken voordat hij doorgaat met het afdrukken van de regel.

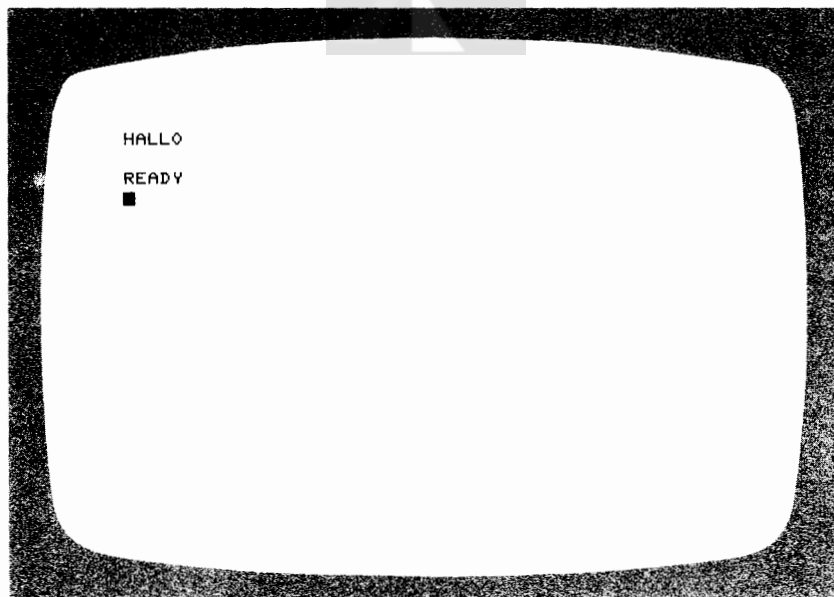
Figuur 3.11 geeft een voorbeeld waarin de pijl naar beneden op deze manier is gebruikt. Typ de PRINT opdracht exact op de manier zoals deze aan de bovenzijde van het scherm staat. Bij ieder ↓ symbool moet u ESC en daarna CONTROL-↓ typen.

Wanneer U de RETURN-toets aan het einde van de regel indrukt, drukt de computer de boodschap tussen de aanhalingstekens af precies zoals u hem opgeslagen hebt. Hij komt eerst de opgeslagen pijl naar beneden tegen. Die zorgt ervoor dat hij een regel omlaag gaat voordat het woordje HOGER wordt afgebeeld. Daarna komt hij nogeens twee pijlen naar beneden tegen. Deze zorgen ervoor dat de computer nogmaals twee regels overslaat voordat hij het woordje LAGER afdrukt. U kunt met losse PRINT opdrachten hetzelfde resultaat krijgen, maar dat is veel gecompliceerder. Deze ESC en CONTROL tekens vereenvoudigen de plaatsbepaling van een tekst.

De speciale functietoetsen

U staat misschien verbaasd over de speciale toetsen zoals START, OPTION, SELECT en HELP. Op de Atari 600XL en 800XL bevinden zich deze vierkante zilverachtige knoppen aan de rechterkant van het toetsenbord. Op de 1200XL, 1400XL en 1450XL zijn deze toetsen verdeeld over de zilverachtige strip vlak boven het toetsenbord. Op de oudere Atari computers zijn het de speciale toetsen met gele knoppen aan de rechterzijde van het toetsenbord.

Behalve wanneer u een programmeur met ervaring bent, zult u deze toetsen alleen gebruiken bij programma's die u in de winkel koopt. Veel van de Atari spelletjes gebruiken de START, OPTION en SELECT toetsen als één een-knop instructie voor nieuw spel, moeilijkheidsgraad of het veranderen van de speelwijze. U hebt de OPTION toets ook nodig om de BASIC taal uit te schakelen bij het laden van



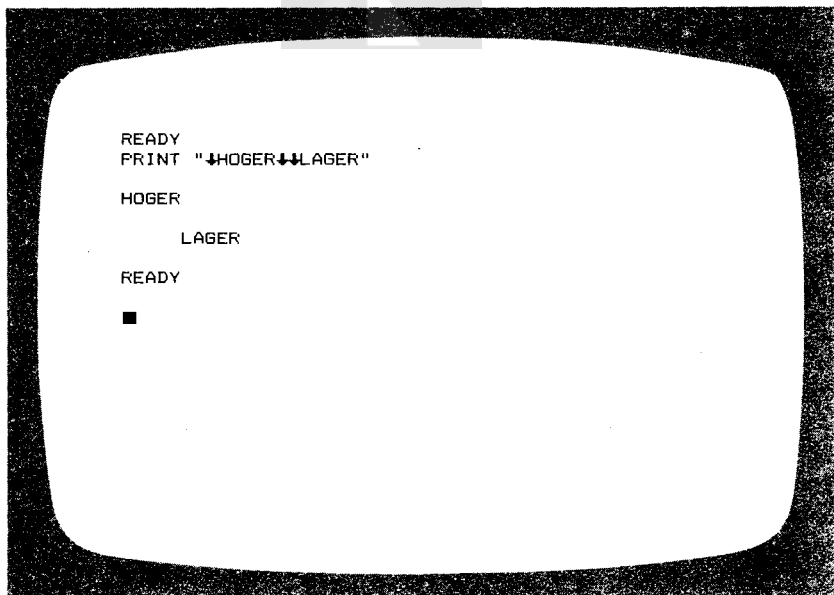
Figuur 3.10: ESC-CONTROL-CLEAR maakt het scherm voor een PRINT schoon.

bepaalde diskette programma's. Het is mogelijk de toetsen op te nemen in eigen geschreven programma's. Dit is echter voor dit boek te technisch.

HELP is de nieuwe toets die Atari heeft toegevoegd toen ze de vormgeving vernieuwden. Op het ogenblik zijn er nog niet veel programma's die deze toets gebruiken, maar er zijn er een paar in ontwikkeling.

De Atari 1200XL en 1400XL computers hebben ook nog vier *programmeerbare functietoetsen*, met opschrift F1 tot F4. Deze kunnen op dezelfde manier als de START, OPTION en SELECT toetsen geprogrammeerd worden, maar ook dit is te technisch.

De RESET-toets is de laatste uit deze groep. In hoofdstuk 1 is deze toets beschreven als een soort paniek toets, waarmee u de controle over uw computer terugkrijgt als u niet meer weet wat er gebeurt. Als u op de RESET-toets drukt, stopt de computer met hetgeen hij op



Figuur 3.11: Door pijlen naar beneden in uw PRINT opdracht op te nemen kunt u tekst op meer dan één regel afdrukken.

dat moment doet en schakelt terug in de stand waarin hij stond toen u hem aanzette.

Als een „her-start” toets is de RESET-toets erg handig. In de volgende hoofdstukken zullen we vaak dingen doen die het beeldscherm wijzigen of de computer in vreemde standen plaatst. In plaats van te overleggen welke instructies u moet geven om de computer naar zijn uitgangspunt terug te laten keren, is het vaak eenvoudiger de RESET-toets te gebruiken. Het scherm wordt weer blauw en het woordje READY verschijnt in de hoek.

RESET is echter een drastische stap. Hij stopt alles en doet veel van het gedane werk teniet. Het kan zijn dat u dat wilt, maar u moet het zeker weten voordat u erop drukt. Wees vooral voorzichtig wanneer u programma's in het geheugen hebt: de RESET-toets kan delen van lange programma's uitwissen.



Vrije oefeningen

Nu u de inleiding over het toetsenbord en beeldscherm hebt doorgekeken, wilt u misschien enkele oefeningen doen. Dit is een *aanvulling*, een aanloop tot verdere experimenten, iets om over na te denken. Ik hoop dat u ze probeert, maar ze zijn niet perse nodig om verder te kunnen lezen. De antwoorden staan achterin het boek achter de bijlagen.

1. Gebruik de pijltoetsen om op het scherm blokletters te schrijven met het sterretje (*). Beweeg de cursor, plaats sterretjes waar u ze wilt hebben. Vergeet niet alleen de pijltoetsen te gebruiken: U verknoeit uw tekening met een ERROR als u op RETURN drukt.
2. Op pagina 39 zagen we hoe we meer dan een PRINT opdracht in een enkele instructie kunnen stoppen; door ze door een dubbele-punt (:) te scheiden. Gebruik drie PRINT opdrachten om het volgende op het scherm te krijgen:

EEN
TWE
DRIE

Kunt u hetzelfde resultaat verkrijgen met een enkele PRINT opdracht door gebruik te maken van de ESC en de pijltoetsen?

3. Vervang op figuur 3.11 de pijlen naar beneden door pijlen naar boven. Wat gebeurt er als u op RETURN drukt? Waarom drukt de computer het woordje READY direct boven de opdracht?

Samenvatting

Het toetsenbord en het scherm zijn met elkaar verbonden. Met het toetsenbord kunt u letters en cijfers intypen die tegelijkertijd op het scherm verschijnen. U kunt ook CONTROL tekens typen waardoor de cursor zich over het scherm beweegt of waardoor het scherm verandert. U kunt deze CONTROL-toetsen gebruiken om een regel die u getypt hebt te veranderen, tekens in te voegen of te verwijderen, of het scherm schoon te maken.

Met de PRINT opdracht, vraagt u de computer boodschappen op het scherm te laten zien. U kunt zelfs besturingstekens (control character) in de boodschap opnemen, door gebruik te maken van de ESC-toets, zodat de computer het scherm schoonmaakt voordat de boodschap afgedrukt of naar een andere plaats op het scherm gestuurd wordt.

U hebt veel geleerd in dit hoofdstuk. U weet nu alles over het gebruik van het toetsenbord en hoe en met wat de computer het scherm vult. Het belangrijkste is dat u uw eerste opdracht hebt gegeven en dat de computer daarop reageerde. Op de volgende bladzijden zullen we vele andere instructies behandelen zodat u uw computer nog beter de baas wordt.

Hoofdstuk 4

Grafieken op de Atari

Als u Pac-Man of een ander videospelletje hebt gespeeld, weet u dat uw computer meer kan dan alleen letters en cijfers op het scherm af te drukken. Hij kan kleurrijke en indrukwekkende bewegende beelden vertonen.

Dit hoofdstuk gaat over het grafische systeem van uw computer. U leert coördinaten te gebruiken en punten op het scherm te zetten. Met enkele eenvoudige instructies kunt u letters afdrukken, kleuren kiezen en beelden tekenen.

De Atari computer heeft 16 verschillende modi om tekst en grafieken af te beelden. Dit hoofdstuk geeft hiervan een overzicht en behandelt er twee uitvoeriger. De andere standen komen in hoofdstuk 8 aan de orde.

De grafische mogelijkheden zijn de meest aantrekkelijke van uw computer: ik hoop dat u er plezier in zult hebben.

Wat zijn grafieken?

Het woord *grafisch* (*graphic*) slaat op elke soort afbeelding op het scherm, als het iets anders is dan alleen maar tekst. Dit kan een tekening zijn zoals b.v. een rechthoek of een kubus. Het kan een balk of een lijn zijn zoals bij een overzicht over de ontwikkeling van de economie. Of het is een spelletje zoals Pac-Man.

Van de 16 grafische standen heten er 5 *tekststanden* (*textmodes*), omdat zij gebruikt worden om letters te typen in plaats van lijnen te trekken. U hebt al een vorm van tekstweergave gezien, de letters die u op het scherm hebt getypt. De vier andere manieren vormen de letters

op afwijkende wijze en u kunt de letters afzonderlijk kleuren. De andere standen zijn wat gecompliceerd in het gebruik, ik behandel ze daarom pas in hoofdstuk 8.

De elf andere zijn echte *grafische standen* (*graphics modes*). Ze werken allemaal volgens hetzelfde principe: u kunt afzonderlijke punten tekenen of in elke richting lijnen over het scherm trekken. De standen verschillen wat het aantal kleuren en de mate van detail of *raster* betreft, die ter beschikking staan.

Waarom al deze standen? Atari heeft gemeend dat het het beste is ieder teken een andere eigenschap te geven. Staafdiagrammen hebben dikke lijnen nodig met relatief weinig detail. Voor een ingewikkelde tekening moet u dunne lijntjes kunnen trekken met veel details. Door uit de verschillende grafische standen te kiezen, kunt u precies dat krijgen wat u wenst.

Een andere reden voor zoveel verschillende standen heeft te maken met het computergeheugen. Wanneer u van de computer een erg fijn raster en veel kleuren verlangt, heeft hij veel geheugen nodig om het beeld op te slaan. Als u een van de duurdere Atari computers hebt of een geheugenuitbreiding is dat geen probleem: zelfs bij het fijnste raster neemt het beeld slechts een fractie van het geheugen in beslag. Als u echter alleen over de basis Atari 600XL beschikt, beslaat het grafische fijne raster meer dan de helft van de beschikbare ruimte. Dit kan problemen opleveren als u een gecompliceerd programma schrijft. Wat de programma's in dit boek betreffen hoeft u zich geen zorgen te maken. Ze werken allemaal op een niet uitgebreide 600XL.

U kiest dus de exacte combinatie van eigenschappen die u nodig hebt. Als u vier kleuren en een fijn raster nodig hebt, is dit geen probleem. Hebt u minder nodig dan kunt u geheugen sparen door een van de andere standen te kiezen met minder details of minder kleuren.

Ter vereenvoudiging behandelt dit hoofdstuk slechts twee van de zestien beschikbare standen. De eerste is de normale tekststand, die u al gebruikt hebt. De andere is stand 7, *vierkleurige grafieken* (*four-color graphics*), waarmee u gedetailleerde beelden op het scherm kunt tekenen.

Dit zijn de belangrijkste van de zestien standen en u kunt ze voor alles wat er in dit boek behandeld wordt gebruiken. Hoofdstuk 8 bespreekt de andere standen en geeft enkele voorbeelden met betrekking tot het gebruik.

Tekst: GRAPHICS 0

Dit is de stand die u tot nu toe gebruikt hebt. Als u de computer aanzet staat hij automatisch op gewone tekst, zodat u niets speciaals hoeft te doen. Misschien wilt u vanuit een andere grafische stand wel eens terugschakelen naar de normale tekststand. Daarom moet u weten wat u dan moet doen.

U schakelt een grafische stand in door het woord GRAPHICS in te typen, gevolgd door het nummer van de stand. De normale tekststand is 0, dus:

GRAPHICS 0

Evenals de PRINT opdracht, is dit een instructie. U moet dus nadat u dit hebt ingetypt op RETURN drukken. Als u dat doet flitst het scherm en opnieuw verschijnt er READAY. U hebt zojuist de stand van stand 0 naar stand 0 gezet. In een later stadium moet u dit typen als u uit een grafische stand terug wilt schakelen naar gewone tekst.

We zullen eens even alles wat we van de tekststand weten op een rijtje zetten:

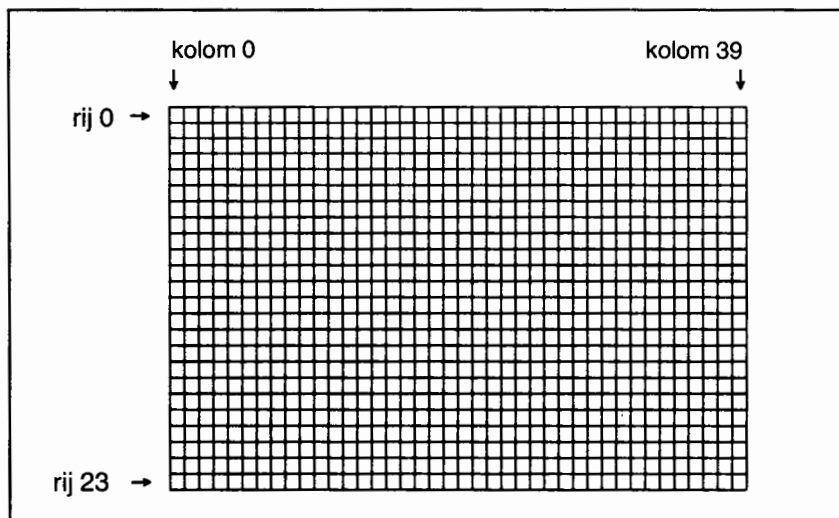
- Met de PRINT opdracht kunnen we de computer vragen een tekst tussen aanhalingstekens op het scherm te zetten. Als we niets anders aangeven zet de computer de boodschap op de regel na de PRINT instructie op het scherm.
- Door het typen van een ESC-CONTROL-CLEAR aan het begin van de tekst tussen de aanhalingstekens in een PRINT opdracht, maakt de computer het scherm schoon voordat de tekst op het beeld komt.
- Door de ESC, CONTROL en een van de pijltoetsen te gebruiken in de PRINT opdracht, kunnen we de computer vragen naar een andere deel van het scherm te gaan voordat de boodschap neergezet wordt. Het is ook mogelijk regelspatiering aan te brengen door lege regels af te drukken, maar dit is erg omslachtig.

De POSITION opdracht

Met de opdrachten die u tot nu toe hebt geleerd kunt u overal op het beeld tekst plaatsen. Misschien vond u het moeilijk om zich met een herhaling van ESC- en CONTROL-toetsencombinaties over het scherm heen en weer te bewegen. Daarom heeft Atari een andere opdracht toegevoegd: de POSITION opdracht (plaats). Hiermee kunt u direct naar elke gewenste plaats op het scherm gaan, voordat u uw tekst afdrukt.

Als u de POSITION opdracht gebruikt moet u de regel en kolom vermelden waarop u de cursor wilt plaatsen. De computer beschouwt het scherm als een rooster met in de breedte 40 kolommen en 24 rijen in de lengte (zie figuur 4.1). Elke regel en kolom zijn voorzien van een getal. De kolommen hebben van links naar rechts de nummers 0 tot 39 en de regels van boven naar beneden 0 tot 23.

Met dit nummersysteem kunt u elk vakje in het rooster aanduiden met een paar getallen. Met het eerste getal geeft u de kolom aan, dus voor de computer het aantal stappen vanaf de linkerkant. Met het tweede getal geeft u de regel aan, om aan te duiden hoe ver u vanaf



Figuur 4.1: De Atari computer beschouwt het scherm als een rooster.

de bovenkant naar beneden wilt. Het gekozen vakje is dat waar de kolom en de regel elkaar ontmoeten.

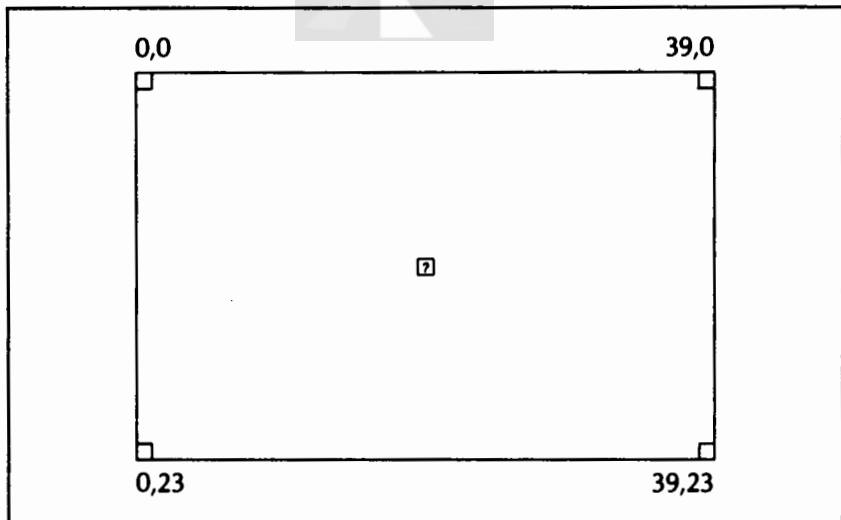
Deze twee getallen worden de *coördinaten* van het vakje genoemd. Het eerste getal is altijd de kolom of de afstand van links; het wordt vaak de *x-coördinaat* genoemd. Het tweede getal heeft altijd betrekking op de regel of de afstand van boven; het wordt de *y-coördinaat* genoemd. De programma's in dit boek gebruiken de letters X en Y om de horizontale en verticale posities aan te duiden. U kunt deze getallen beschouwen als „X spaties overslaan en Y regels naar beneden”.

Figuur 4.2 toont de coördinaten van enkele vakjes op het scherm. Het vierkantje in de linkerbovenhoek is altijd 0,0 - 0 spaties overslaan en 0 regels omlaag. Dit geldt voor het nummersysteem in alle grafische standen van de Atari. Voor het rechthoekje in de rechterbovenhoek moet u 39 kolommen tellen. Dus de x-coördinaat van dit vakje is 39. De y-coördinaat blijft 0, omdat u geen enkele regel naar beneden ging. Daarom schrijven we '39,0'. Dit betekent „39 spaties overslaan en 0 regels omlaag”. Voor het vakje in de linkerbenedenhoek moet u 0 spaties overslaan en 23 regels naar beneden, dus zijn de coördinaten 0,23. Om het vakje beneden rechts te bereiken moet u helemaal naar beneden en naar rechts gaan; de coördinaten zijn daarom 39,23.

Om het vakje in het midden aan te geven, moeten we even naar de figuur kijken. Wat de x-coördinaat betreft moeten we een nummer kiezen in het midden tussen 0 en 39. Dit is 19 of 20, we zullen 19 kiezen omdat we de rechterkant van het scherm niet te vol willen hebben. Op dezelfde manier moeten we een waarde hebben ergens in het midden tussen 0 en 23. Hier kiezen we 11. Deze coördinaten (19,11) bepalen het rechthoekje in het middel van figuur 4.2: 19 spaties overslaan en 11 regels naar beneden.

Met de POSITION opdracht kunt u coördinaten over het gehele scherm vastleggen. Als u hierna een PRINT opdracht geeft begint de boodschap in het vakje dat u bepaald hebt.

Als die coördinaten toevallig op een regel komen waar al tekst staat wordt de nieuwe boodschap over de vorige heen geschreven. Omdat dit moeilijk te lezen is, is het vaak het beste om CONTROL/CLEAR



Figuur 4.2: Voorbeelden van coördinaten in de GRAPHICS 0 (tekststand).

in te drukken voordat u een POSITION opdracht geeft. Als u dit niet doet, kan het zijn dat u niet ziet wat de opdracht gedaan heeft en het televisiescherm zal niet overeenkomen met de afbeelding in het boek.

Nu u coördinaten kent is de POSITION opdracht erg simpel. Om de cursor naar het midden van het scherm te verplaatsen hoeft u slechts te typen:

```
POSITION 19,11
```

Als u nu op de RETURN toets drukt zet de computer de cursor in het midden van het scherm. Maar omdat u niets liet afdrukken in dat vakje, is hij meteen naar de volgende regel gesprongen en schrijft hier READY. De cursor wacht nu hieronder op de volgende tekst. Maak het scherm schoon en typ:

```
POSITION 19,11: PRINT "MIDDEN"
```

Druk hierna op return en het scherm moet er dan uitzien als op figuur 4.3.

Het afgedrukte woord verschijnt een beetje rechts van het midden op het scherm. Dit komt omdat met POSITION alleen het begin van het woord wordt vastgelegd. De M van MIDDEN verschijnt op 19,11; het woord verschijnt er rechts van. Om het woord precies in het midden te krijgen moet u het twee of drie spaties naar links laten beginnen. Probeer andere coördinaten in de POSITION opdracht om het woord meer naar het midden te krijgen.

U maakt kennis met een apart verschijnsel met betrekking tot het scherm als u de volgende regel typt en dan op RETURN drukt:

POSITION 0,20: PRINT "LINKERKANTLIJN"

De tekst komt onderop het scherm, twee plaatsen links van de normale linkerkantlijn. Als u de computer niet precies met POSITION aangeeft dat hij een tekst in kolom 0 of 1 moet beginnen begint hij met de ingetypte regels in kolom 2. Als u uw tekst op een regel met het woordje READY wilt hebben, moet u de tekst met de POSITION opdracht in kolom 2 plaatsen.

Bij het gebruik van de POSITION opdracht kunt u soms de melding krijgen:

ERROR 141

Als u dit opzoekt in de tabel met foutcodes in bijlage C, ziet u dat dit „cursor buiten bereik” betekent. Dit houdt in dat u geprobeerd hebt de cursor op een plaats buiten het beeld te zetten - een van de coördinaten was te groot. Onthoud dat u voor de X en de Y coördinaten maximaal resp. 39 en 23 kunt gebruiken. Als u een foutmelding krijgt, moet u het scherm schoonmaken en opnieuw beginnen.

Experimenteer een beetje met de POSITION opdracht en het plaatsen van de coördinaten. U zult merken dat het handig is in termen van coördinaten te denken als u met grafische mogelijkheden werkt. Ofschoon het aantal kolommen en regels in de andere grafische standen afwijkt, blijft het basis-idee hetzelfde.



Vierkleuren tekening: GRAPHICS 7

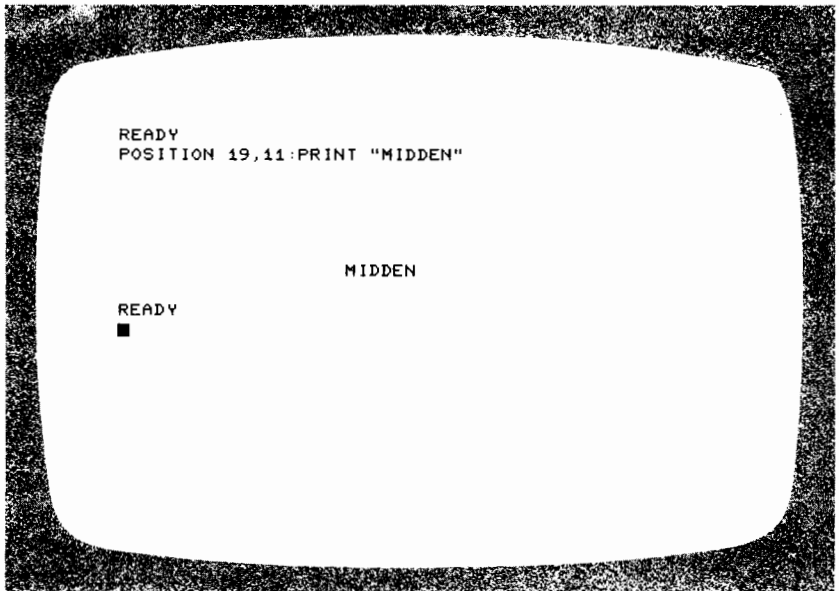
Tot nu toe hebben we alleen met letters gewerkt. Nu gaan we echte grafieken maken door ze op het scherm te tekenen. In de rest van dit hoofdstuk leert u punten te plaatsen, lijnen te trekken en figuren op het scherm te tekenen.

Om figuren te tekenen gebruiken we de grafische stand 7, een van de *vierkleuren grafische* standen. We houden ons nog niet bezig met de standen tussen 0 en 7: die behandelen we later.

Om in de vierkleuren stand te komen moet u

GRAPHICS 7

typen en op RETURN drukken. Het bovenste gedeelte van het scherm wordt zwart en het woordje READY komt onder in beeld.



Figuur 4.3: Gebruik de POSITION opdracht om tekst op de door u gewenste plaats op het scherm te krijgen.

Het zwarte gedeelte van het scherm hebt u nodig voor de grafische beelden die u gaat tekenen. U kunt de cursor niet in dit gedeelte plaatsen, en u kunt er ook geen teksten afdrukken.

U kunt het gebruiken om tekeningen met punten en lijnen te maken. Dit gedeelte heet het *grafische scherm* (*graphic screen*).

De grafieken kunnen niet tot helemaal aan de onderkant van het beeld komen omdat de computer een beetje ruimte over moet laten om instructies te geven en boodschappen af te drukken. Het gedeelte waar vier regels staan heet het *tekstraam* (*text window*). In hoofdstuk 8 leert u hoe u het grafische scherm tot op dit gedeelte van het scherm kunt laten doorlopen. Nu hebben we echter het tekstraam nodig om tijdens het typen de tekst te kunnen volgen.

Het kan zijn dat u tijdens het lezen van dit boek meer tekst wilt zien dan er in het tekstraam past. Met name in de programmeerhoofdstukken zult u vaak een tekst willen bekijken die uitgebreider is en niet op de vier regels past. Als dit het geval is kunt u op RESET drukken of een GRAPHICS 0 opdracht geven, zodat u naar de normale tekststand terugkeert.

De computer geeft u in deze stand vier *verfkwasten* (*paintbrush*), voor iedere kleur een. Drie hiervan zijn tekenkleuren (oranje, groen en blauw) terwijl de vierde een *wisborstel* (*erasebrush*) is, die gebruikt wordt om punten van het scherm te verwijderen. De wisborstel is eigenlijk een zwarte verfkwast die met de achtergrondkleur teken: als u zwart verft verdwijnt het beeld. Het is mogelijk de kwasten andere kleuren te geven dan oranje, groen, blauw en zwart, maar op het ogenblik is dit voldoende.

Nummer van de verfkwast	Standaard kleureninstelling
COLOR 1	Oranje
COLOR 2	Groen
COLOR 3	Blauw
COLOR 0	Wissen (= zwarte achtergrond)

Figuur 4.4: De kleuren van de vier verfkwasten

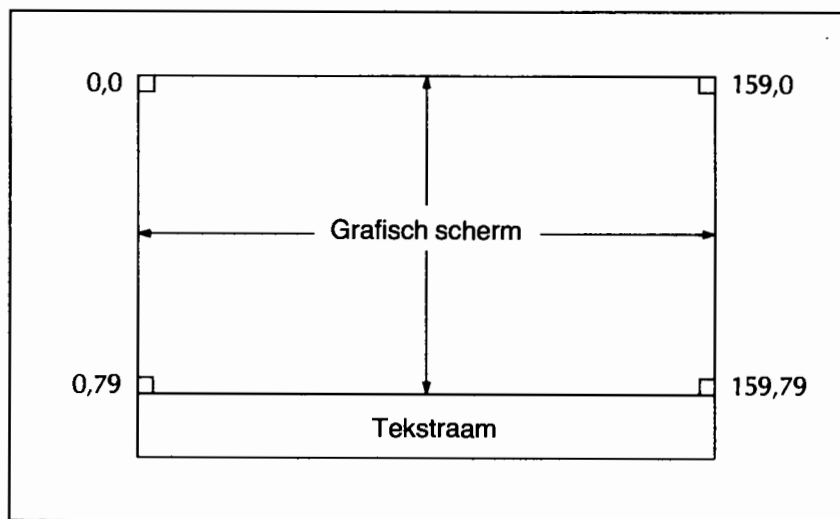
Voordat u begint te tekenen, moet u de kleuren kiezen die u wilt gebruiken. U doet dat door een COLOR instructie te geven zoals:

COLOR 1

Deze kiest verfkwest nummer 1, welke standaard oranje is. Figuur 4.4 geeft een overzicht van de kleuren.

Figuur 4.5 toont de verdeling van het vierkleuren grafisch scherm. Het bovenste deel van het scherm is verdeeld in een fijn rooster van 160 punten breed en 80 punten hoog. U kunt ieder punt in dit rooster met zijn coördinaten benoemen. De grafische coördinaten hebben hetzelfde systeem als de coördinaten in de tekststand. In deze stand loopt dex-coördinaat van 0 tot 159 en de y-coördinaat van 0 tot 79. Figuur 4.5 toont de coördinaten die u moet gebruiken om de punten aan te stippen in de vier hoeken van het grafisch scherm.

U kunt op het grafisch scherm punten of lijnen tekenen door twee instructies te gebruiken. Met PLOT (teken) kunt u een enkele punt op het scherm laten oplichten. Met DRAWTO (trek tot) kunt u een rechte lijn tussen twee punten trekken.



Figuur 4.5: Ontwerp voor een grafieke beeldscherm met vier kleuren

Over het algemeen zet u eerst met PLOT een punt. Hiervoor kiezen we de oranje verfkwast door te typen:

COLOR 1

Typ daarna bijvoorbeeld de regel:

PLOT 10,20

Er moet nu een klein lichtpuntje in de linkerbovenhoek van het scherm verschijnen. Als het niet te voorschijn komt moet u controleren of u de instructie GRAPHICS 7 en een COLOR opdracht om de kleur te kiezen hebt gegeven.

Zorg er ook voor dat uw televisie goed afgesteld is zodat u het puntje kunt zien. Het is kleiner dan een punt aan het einde van een zin.

'10,20' uit de PLOT opdracht zijn de coördinaten van het punt: 10 eenheden van links, 20 eenheden naar beneden. Dit zijn relatief kleine afstanden in het rooster van 160 x 80 van de grafische coördinaten; als u wilt zien hoe klein moet u het punt 0,0 plotten. In de gewone tekststand is '10,20' best wel een grote afstand. Grafische roosters zijn vaak fijner dan tekstroosters.

U kunt zoveel punten op het scherm zetten als u wilt. Experimenteer een beetje zodat u wat gevoel krijgt voor de grafische coördinaten. Elke PLOT opdracht betreft een ander apart punt, ervan uitgaande dat u niet boven het maximale getal gaat (159 voor de x-, 79 voor de y-coördinaten). Als u een verkeerd getallenpaar aangeeft verschijnt er een foutmelding, maar u kunt doorgaan met typen.

Met de PLOT opdracht krijgt u alleen maar losse punten, om lijnen te trekken moet u de DRAWTO opdracht gebruiken. Zoals de naam al aangeeft (trek tot), verbindt de DRAWTO opdracht de laatst getekende punt met de punt die u als bestemming hebt aangegeven. Probeer eens de volgende instructie:

DRAWTO 159,79

Er zal nu een oranje lijn worden getekend vanaf de laatste punt die u neerzette naar de rechteronderkant van het scherm. Afhankelijk van de plaats waar de laatste punt werd gezet, zal de lijn er niet hele-

maal ononderbroken uitzien. Dit „trapeffect” wordt veroorzaakt door het beperkt oplossend vermogen van een televisiescherm en de gekozen stand. Het kan ook zijn dat de lijn niet precies oranje is. Dit is afhankelijk van de hoek waaronder hij getrokken is. Verticale lijnen op de Atari neigen er bleek of purperachtig uit te zien, vergeleken bij de volle kleuren van horizontale lijnen.

PLOT en DRAWTO moeten in combinatie worden gebruikt. Om een lijn te trekken, moet u over het algemeen eerst een PLOT opdracht geven om het startpunt van de lijn op het scherm te krijgen. Daarna gebruikt u de DRAWTO opdracht om de lijn tot zijn eindpunt te trekken. Als u direct begint met een DRAWTO opdracht, zonder eerst een startpunt neer te zetten, zal de computer de lijn laten beginnen bij de coördinaten die het laatst gebruikt zijn.

Als voorbeeld trekken we nu een horizontale lijn over het scherm. Typ de volgende instructies:

```
PLOT 0,20  
DRAWTO 159,20
```

Na de eerste instructie verschijnt er aan de linkerkant op eenderde van boven van het scherm een punt. Met de tweede instructie trekt de computer een rechte oranje lijn naar de rechter overkant.

Stel dat u hieronder een groene lijn wilt trekken. Dan moet u verkwast 2 nemen en een nieuwe lijn trekken. Dit doet u met de volgende drie instructies:

```
COLOR 2  
PLOT 0,30  
DRAWTO 159,30
```

Ziet u dat de Y (verticale) coördinaat veranderd is om de lijn verder naar beneden te krijgen? Hoe kunt u een blauwe lijn trekken iets onder de groene?

Als u de volgende DRAWTO opdracht geeft, zal de computer de nieuwe lijn laten beginnen op het eindpunt van de vorige. Een reeks van DRAWTO opdrachten is net als het trekken van een aantal rechte lijnen zonder de pen van het papier te halen. Het is niet nodig een PLOT opdracht te geven voor elk nieuw deel van de lijn.

Als u een nieuwe lijn wilt trekken die niet met de andere verbonden is, moet u het nieuwe startpunt van de lijn met PLOT bepalen. U moet het zien als volgt: DRAWTO trekt een lijn van het punt waar u bent naar een nieuw punt, terwijl PLOT de verfkwast beweegt zonder een lijn te trekken en op een bepaalde plaats neerzet.

Om dit echt te zien tekenen we het vierkant uit figuur 4.6. Maak het scherm schoon en kies verfkwast 1 met de regels:

```
GRAPHICS 7  
COLOR 1
```

Geef hierna de volgende vijf instructies:

```
PLOT 10,10  
DRAWTO 70,10  
DRAWTO 70,10  
DRAWTO 10,70  
DRAWTO 10,10
```

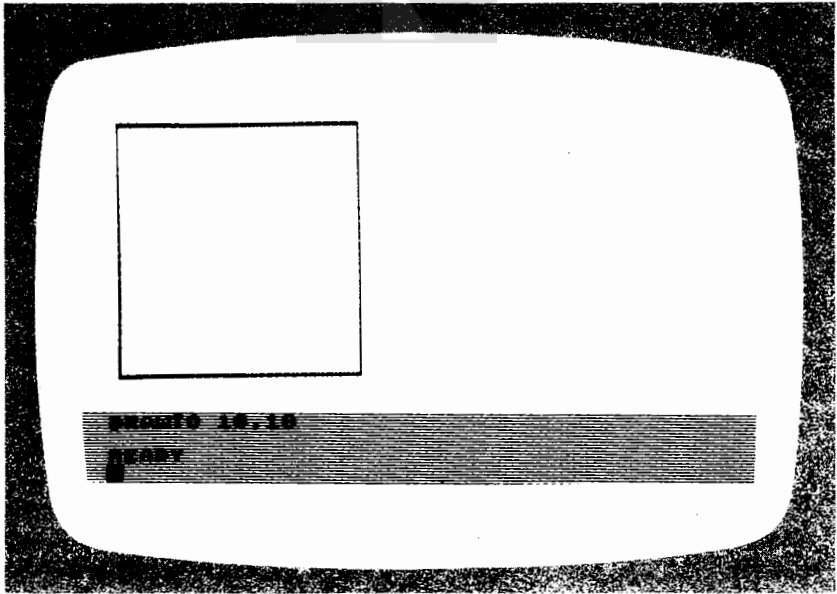
Met de PLOT opdracht verschijnt er een kleine punt in de linkerbovenhoek van het scherm. Elke DRAWTO instructie geeft het vierkant een zijkant, totdat de laatste de verfkwast weer terugbrengt naar het startpunt. Daar de vier zijden van het vierkant met elkaar verbonden zijn, was het niet nodig de verfkwast op te lichten met een PLOT opdracht.

Er zijn diverse manieren om delen van het scherm te wissen. Het eenvoudigste is het gebruik van de wisborstel, verfkwast 0, door in te typen:

```
COLOR 0
```

Als u dit doet, worden uw PLOT en DRAWTO opdrachten *wis* opdrachten, totdat u een andere verfkwast kiest. Probeer het eens met de volgende regels:

```
COLOR 0  
PLOT 10,10  
DRAWTO 70,10
```



Figuur 4.6: Het tekenen van een vierkant op het scherm.

De bovenkant van het vierkant verdwijnt. (Als dat niet gebeurt, zit er waarschijnlijk een fout in een van de coördinaten. Probeer het nog eens door de laatste twee voorbeelden opnieuw in te typen.)

Als u het hele scherm uit wilt wissen gaat dit het beste met:

GRAPHICS 7

Hiermee wordt het scherm in de beginstand teruggezet (gewist) en u kunt weer opnieuw beginnen. U moet ook weer COLOR 1 opnieuw typen voordat u met het kleuren kunt beginnen.

We zijn nu aan het einde van de inleiding van de grafische mogelijkheden van Atari. Er zijn meer mogelijkheden in de grafische stand. Dit was voorlopig een oppervlakkige kennismaking. Hoofdstuk 8 van dit boek beschrijft in het kort het gebruik van de 14 andere grafische standen waarmee u gevarieerde effecten kunt krijgen. Het beschrijft standen waarmee u maar liefst 16 kleuren op het scherm tegelijker-

tijd kunt gebruiken. Hier wordt het werken met de *grafieken met een fijn raster (high-resolution graphics)* uitgelegd, een een-kleuren stand waarmee u fijnere punten neer kunt zetten dan met vierkleuren mogelijk is. Dan wordt nog de *speciale tekststand (special-text)* besproken die andere mogelijkheden tot het afdrukken van tekst, inclusief kleur geeft. Maar voorlopig is dit genoeg.

Vrije oefeningen

1. Gebruik drie POSITION en drie PRINT opdrachten in een lange instructie, zet hiermee uw naam op drie plaatsen op het scherm. Probeer u voor te stellen waar de woorden zullen verschijnen voordat u op RETURN drukt.
2. Herhaal de instructies die u gebruikte om de oranje vierhoek in figuur 4.6 te tekenen, gebruik nu een tweede serie instructies om een blauwe vierhoek binnen de eerste te krijgen.

Samenvatting

In de eerste drie hoofdstukken van het boek hebt u geleerd hoe u uw computer moet installeren en hoe u hem kunt gebruiken met commerciële software pakketten. Daarna hebt u geleerd zelf uw computer een instructie te geven, waarmee u een boodschap op het scherm kunt afdrukken.

In dit hoofdstuk hebt u dat wat u over het grafisch systeem van uw computer moet weten geleerd. U hebt gewerkt met coördinaten in de POSITION instructie, om hiermee op elke plaats op het scherm tekst te krijgen. Daarna hebt u geleerd punten en lijnen neer te zetten in de vierkleuren stand. Elke opdracht kunt u bij alles wat er nog wordt behandeld gebruiken.

Het belangrijkste wat u nog niet leerde is: het opslaan van uw instructies. Voor de procedure om bijvoorbeeld een vierkant te tekenen had u zeven opdrachten nodig die ieder voor zich met behulp van het toetsenbord ingetypt moesten worden. Stel dat u weer een vierkant wilt tekenen, dan moet u alle zeven stappen opnieuw doen.

Daarvoor is er een gemakkelijker manier die in deel 2 behandeld wordt.

Deel 2 **Programmeren**

Een programma schrijven

In deel 1 van dit boek hebt u maar een paar dingen geleerd die uw computer kan. U weet nu hoe u instructies moet geven om teksten af te drukken of simpele figuren te tekenen. Als u instructies gaf voerde de computer ze direct uit.

Er is echter een grote beperking. U moest elke instructie via het toetsenbord intypen. Als u een foutje maakte in een serie van opdrachten, moest u het scherm schoonmaken en helemaal opnieuw beginnen. Ook als u bepaalde taken (zoals het tekenen van een vierhoek) wilde herhalen, moest u de hele serie opdrachten iedere keer opnieuw intypen. Hierdoor bent u gedwongen tot korte procedures.

Met een *programma* (*program*) kunt u een serie opdrachten in het computergeheugen opslaan en ze als een totaalpakket laten uitvoeren. Bij deze instructies zal de computer zo reageren alsof u de instructies intypte via het toetsenbord. En als de computer klaar is blijven de opdrachten bewaard, klaar om opnieuw te gebruiken. Als een van de stappen verkeerd was, of als u een variatie wilt uitproberen, bent u vrij om het programma te veranderen en opnieuw uit te voeren.

Omdat programma's het mogelijk maken een procedure te ontwerpen en deze vele keren opnieuw te gebruiken, zijn ze geschikt om met uw computer meer gecompliceerde taken uit te voeren. Als u gedwongen zou zijn steeds een voor een uw opdrachten in te typen, zou u nooit meer proberen dan

PRINT "HALLO"

Met opgeslagen programma's is het eenvoudig opdrachten aaneen te rijgen tot grote veel omvattende procedures die u altijd opnieuw

kunt gebruiken. Veel mensen maken programma's die uit honderden opdrachten bestaan.

Zoals u in de komende hoofdstukken zult merken, zijn programma's een nieuwe manier om uw computer te besturen. Met een programma kunt u woorden en getallen opslaan en berekeningen uitvoeren. U kunt een stel instructies tot groep vormen en deze herhaaldelijk laten uitvoeren in een lus, of uw computer alleen iets laten doen als aan een bepaalde voorwaarde wordt voldaan. Met dit gereedschap bent u in staat uw computer complexe taken te laten verrichten. Terwijl hij die uitvoert kunt u zich ontspannen.

Dit hoofdstuk laat u zien hoe u programma's in het computergeheugen inbrengt en laat uitvoeren. We geven een paar voorbeelden om u op weg te helpen.

Om programma's te leren schrijven, moet u een *programmeertaal* (*programming language*) leren. Computerontwerpers realiseerden zich al lang geleden dat hun machines geen samenhangende zinnen kunnen begrijpen, omdat die vaak vol zitten met woorden die meer dan een betekenis hebben. Daarom besloten ze een systeem van opdrachten op te zetten dat eenvoudig te leren is en toch specifiek genoeg om door een computer begrepen te kunnen worden.

Bij uw Atari computer moet u Atari's versie van een taal die BASIC heet gebruiken. Dit is de afkorting voor „Beginners Alle doeleinden Symbolische Instructie Code“ als u dat wilt geloven. Zoals de naam al zegt is BASIC gemakkelijk te leren en hierdoor is het de populairste taal geworden voor de kleine computers zoals Atari 600XL en 800XL. U hebt al enkele instructies uit de Atari BASIC taal (onder andere PRINT, GRAPHICS en COLOR) gebruikt. Deze en andere instructies kunt u in programma's gebruiken die u gaat schrijven.

Laten we eens kijken hoe dat gaat.

Programma's opslaan

Uw programma bestaat uit een serie van opdrachten, die in het computergeheugen zijn opgeborgen. Als u klaar bent met het invoeren van de opdrachten geeft u de computer de opdracht RUN om de programma's uit te voeren.



Voordat u ook maar iets doet, moet u op RESET drukken en dan het woordje

NEW

intypen. Als u op de RETURN toets drukt, zal de computer alles uit zijn geheugen wissen en u kunt met een schone lei beginnen en uw programma intypen.

Deze NEW (nieuw) instructie moet iedere keer gegeven worden als u met een nieuw programma begint, zelfs als u op de RESET toets heeft gedrukt. Als u dat niet doet kan het gebeuren dat er programmaregels van een ander programma in het computergeheugen achterblijven. Als die regels in het geheugen blijven kunnen ze tussen het programma komen dat u aan het typen bent en het in de war sturen. Maak er een gewoonte van de NEW instructie altijd te gebruiken.

U kunt iedere Atari instructie als een regel in een programma opslaan. U hoeft er alleen maar een nummer voor te zetten. Dit heet een *opdrachtnummer* (*statement number*) en het is voor de computer een signaal om de instructie op te slaan in plaats van direct uit te voeren.

In hoofdstuk 3 heeft u de eerste instructie via het toetsenbord gegeven:

```
PRINT "HALLO"
```

Toen u op RETURN drukte, voerde de computer de opdracht direct uit en zette het woordje HALLO op het scherm. We proberen nu eens een kleine variatie. Typ de volgende regel:

```
10 PRINT "HALLO"
```

Als u op RETURN drukt lijkt het of er niets gebeurt. Het nummer 10 voor het woordje PRINT zegt de computer alleen maar dat hij de regel in het geheugen moet opbergen. Er zit niets speciaals aan het nummer 10; we zouden ieder willekeurig getal hebben kunnen gebruiken.

U kunt controleren of de regel is opgeborgen door een andere instructie in te typen:

LIST

Hiermee vraagt u de computer alle regels die in het geheugen zitten te laten zien. In dit geval is er alleen regel nummer 10.

Nu vragen we de computer de opgeborgene instructie uit te voeren. Hiervoor moet u typen:

RUN

en op RETURN drukken. U krijgt hetzelfde resultaat als wanneer u de instructie zonder nummer zou hebben ingetypt: de computer zet het woordje HALLO op het scherm. De handeling is uitgesteld, totdat u de RUN instructie geeft.

Als u een tweede maal RUN intypt, zal de computer het woord HALLO herhalen. U kunt dit zo vaak doen als u maar wilt: uw instructie blijft in het computergeheugen totdat u hem uitschakelt of NEW typt om het geheugen schoon te maken.

Om een tweede regel aan het programma toe te voegen, hoeft u slechts een nieuwe instructie te typen met een andere nummer ervoor. Weer kan een willekeurig getal genomen worden, maar omdat we 10 al gebruikten voor het opdracht nummer van de eerste instructie nemen we nu 20 voor de tweede:

```
20 PRINT "OPNIEUW HALLO"
```

Als u op RETURN drukt, zal de tweede opdracht in het geheugen naast de eerste worden opgeborgen.

U kunt zien hoe uw programma eruit ziet door LIST in te typen. De computer toont alle opdrachten die opgeslagen zijn:

```
10 PRINT "HALLO"  
20 PRINT "OPNIEUW HALLO"
```

Elke opdracht is opgeslagen als een eenheid. De computer houdt deze met behulp van de regelnummers op volgorde, ongeacht de volgorde waarin u ze intypt.

Nu hebt u twee regels in het computergeheugen. Wat zal er gebeuren als u RUN intypt? De computer zoekt dan door zijn geheugen en voert de opgeborgen instructies op volgorde uit. Als hij klaar is verschijnt er weer READY. Het resultaat ziet er ongeveer zo uit:

```
HALLO  
OPNIEUW HALLO
```

```
READY
```

U kunt overal in uw programma opdrachten toevoegen. Stel dat we een boodschap tussen deze twee regels willen zetten. Alles wat we moeten doen is een instructie typen met een opdrachtnummer tussen 10 en 20 en het zal op de juiste plaats terecht komen. We kunnen bijvoorbeeld 15 kiezen:

```
15 PRINT "TOT ZIENS"
```

Als u het programma met LIST laat afdrukken, kunt u zien dat de nieuwe opdracht tussen de twee andere staat:

```
10 PRINT "HALLO"  
15 PRINT "TOT ZIENS"  
20 PRINT "OPNIEUW HALLO"
```

Typ RUN om het programma uit te voeren. De nieuwe opdracht wordt op volgorde van zijn nummer uitgevoerd:

```
HALLO  
TOT ZIENS  
OPNIEUW HALLO
```

```
READY
```

U kunt elk nummer van 0 tot 32767 voor uw opdrachtnummers kiezen. U doet er echter goed aan om veelvoudigen van tien te gebruiken

(10, 20, 30 enz.). Dit maakt uw programma overzichtelijk en er blijft genoeg ruimte over voor invoegingen.

Om een regel te verwijderen hoeft alleen het regelnummer te in te typen, verder niets. Stel dat u de laatste regel, OPNIEUW HALLO, kwijt wilt. Typ slechts

20

en druk op RETURN. De opdracht met nummer 20 wordt nu uit het computergeheugen gewist, zoals u met de LIST instructie kunt zien:

```
10 PRINT "HALLO"  
15 PRINT "TOT ZIENS"
```

Als u nu de instructie RUN geeft, voert de computer alleen de overgebleven opdrachten uit:

```
HALLO  
TOT ZIENS
```

```
READY
```

Als u een regel wilt veranderen moet u de nieuwe versie met hetzelfde regelnummer intypen:

```
10 PRINT "HALLO DAAR"
```

Als u op RETURN drukt zal deze regel de oude versie van opdracht nummer 10 vervangen. Het programma ziet er dan zo uit:

```
10 PRINT "HALLO DAAR"  
15 PRINT "TOT ZIENS"
```

Het is belangrijk te onthouden dat elk opdrachtnummer slechts eenmaal gebruikt kan worden. Als u een nummer tweemaal gebruikt zal uw nieuwe opdracht de vorige vervangen.

Als u slechts een kleine verandering in een programmaregel wilt aanbrengen, kunt u de opmaaktoetsen gebruiken, die in hoofdstuk 3

worden behandeld. Zoals u zich herrinnert kunt u iedere instructie die op het scherm staat veranderen. U hoeft er alleen de cursor naar toe te brengen, te corrigeren en op RETURN te drukken om de nieuwe versie in het computergeheugen op te slaan.

U kunt alleen opdrachten reviseren die op het scherm staan. Wat moet er gebeuren als u een regel wilt veranderen die momenteel niet op het scherm staat? U kunt met LIST het hele programma oproepen, maar het is eenvoudiger alleen die regel op te roepen waarin iets verandert moet worden. Een afzonderlijke regel krijgt u met LIST op het scherm:

LIST 15

Nu verschijnt er alleen regel 15 op het scherm:

15 PRINT "TOT ZIENS"

Stel dat u een uitroepteken achter TOT ZIENS wilt zetten. Breng de cursor naar het tweede aanhalingsteken, zoals op figuur 5.1. Druk nu op CONTROL en de INSERT toets om een ruimte te maken tussen TOT ZIENS en het aanhalingsteken. Typ dan een uitroepteken en u hebt de regel die u wilde hebben. (Lees hoofdstuk 3 nog een keer over als u door deze revisieinstructies in de war bent geraakt.)

U hebt nu de regel verbeterd maar hij is nog niet in het computergeheugen opgeslagen. Hiervoor moet u op de RETURN toets drukken waardoor de gecorrigeerde regel de vorige vervangt en in het programma wordt opgenomen. Voor wat de computer betreft maakt het niet uit of u de opmaaktoetsen gebruikt voor de wijziging of de regel helemaal opnieuw typt. Het maakt de computer niets uit of u een hele regel opnieuw typt of met LIST het programma opzoekt en dan alleen een teken verandert. Het maakt niet uit hoe de regel op het scherm is gekomen, de computer accepteert hem als een deel van het programma en slaat hem op. Gebruik LIST om te zien of de verandering verwerkt is:

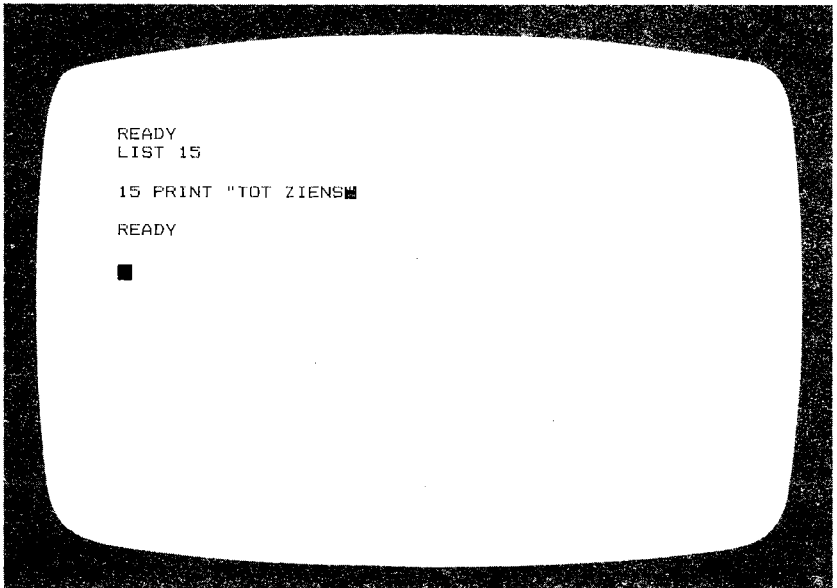
10 PRINT "HALLO DAAR"

15 PRINT "TOT ZIENS!"

Misschien moet u wel eens een instructie als RUN of LIST intypen als de cursor niet op een blanco regel staat. U kunt de computer om een lege regel vragen door de SHIFT en INSERT toets tegelijkertijd in te drukken. U kunt ook CONTROL-CLEAR typen als u het niet erg vindt dat het gehele scherm gewist wordt.

Om een overzicht te krijgen gaan we nog eens na op welke verschillende manieren u opdrachtnummers kunt gebruiken bij het maken van een programma.

- *Toevoegen*: als u een regel met een regelnummer, dat nog niet gebruikt is ingeeft, wordt hij opgeslagen als een nieuwe programma-regel.
- *Invoegen*: als u een regel typt met een nummer tussen twee eerder opgeslagen nummers, zal de regel op die plaats in uw programma ingevoegd worden.
- *Vervangen*: als u een regel ingeeft met een opdrachtnummer dat al eerder gebruikt is, zal de nieuwe versie de oude vervangen.



Figuur 5.1: Het opmaken van een programmaregel.

- *Verwijderen*: als u een opdrachtnummer ingeeft zonder verdere instructie, zal de computer de opdracht die onder dat nummer staat verwijderen.

U weet nu alles wat u moet doen om programma's in de computer te kunnen typen. U weet hoe u programmaregels moet toevoegen, invoegen, vervangen en verwijderen. U weet dat u met LIST een opstelling van uw programma krijgt en hoe u dat moet doen.

Het is tijd voor een praktisch voorbeeld.

Programma: vierhoek tekenen

Aan het einde van hoofdstuk 4 gaf u een serie van zeven grafische instructies om een vierhoek op het scherm te tekenen. Probeer dezelfde serie nog eens:

```
GRAPHICS 7  
COLOR 1  
PLOT 10,10  
DRAWTO 70,10  
DRAWTO 70,70  
DRAWTO 10,70  
DRAWTO 10,10
```


De computer zal overgaan tot de vierkleuren stand en een beeld geven dat er uitziet als op figuur 4.6.

We gaan nu een nieuw programma schrijven. Druk op RESET om terug te keren naar de tekststand en typ de instructie

```
NEW
```

om het computergeheugen schoon te maken. Niet vergeten: een programmaregel blijft zolang in het geheugen totdat u hem òf verwijdert òf met NEW schoon schip maakt òf de computer uitzet.

Wat we nu gaan doen is eenvoudig. We gaan de zeven instructies opnieuw intypen, maar deze keer geven we ze nummers en slaan ze dan als een programma op. Typ deze regels:



```
10 GRAPHICS 7
20 COLOR 1
30 PLOT 10,10
40 DRAWTO 70,10
50 DRAWTO 70,70
60 DRAWTO 10,70
70 DRAWTO 10,10
```

Controleer als u klaar bent of u geen typfouten hebt gemaakt. Typ de regel indien nodig opnieuw of corrigeer met de opmaaktoetsen.

Typ nu RUN in om het programma uit te voeren. Onmiddellijk zal de computer in de grafische stand gaan en de oranje vierhoek tekenen. Het resultaat is hetzelfde als op figuur 4.6, maar de computer deed het nu in een stap in plaats van in zeven.

Bij een procedure van deze lengte ziet u duidelijk het voordeel van een opgeslagen programma. U kunt een fout verbeteren met de opmaaktoetsen in plaats van opnieuw te starten. U kunt de vierhoek opnieuw tekenen met een enkele RUN instructie in plaats van alle zeven stappen opnieuw te typen. Verderop zult u andere voordelen van het schrijven van programma's ontdekken en ook andere toepassingsmogelijkheden.

Programmeertips

Als u uw eigen programma's schrijft, zult u zich een paar dingen aan moeten wennen die het werk vereenvoudigen. Helder denken en goed organiseren staan tussen succesvol programmeren en computernachtmerries. Houdt u aan de volgende drie regels:

1. *Alles goed doordenken.* U bespaart zich een hoop ellende als u een paar minuten besteedt aan het maken van een programmaschema voordat u begint te typen. U hoeft programma's niet volledig uit te schrijven, behalve als u dat prettig vindt, maar u moet voordat u begint een schema maken van de weg die u wilt volgen.
2. *Werk netjes.* Er is niets frustrerender dan het oplappen van een slordig programma. Als u uw programma goed in elkaar zet zodat het eenvoudig te lezen is en het een logische opbouw heeft, begrijpt u het

beter als u er later wijzigingen in aan wilt brengen. Maak gebruik van regelmaat in de opdrachtnummers zoals 10, 20, 30.

3. *Maak notities voor uzelf.* U kunt midden in het programma toelichtingen maken. Toelichtingen die uitleggen wat u gaat doen; ze maken het programma begrijpelijker.

Om op een bepaald punt in uw programma een opmerking te maken moet u een ongebruikt opdrachtnummer en de letters REM (van „remark” = opmerking) intypen. Hierna kunt u op de regel alles schrijven wat u wilt. Voorbeeld: aan het begin van het programma om de vierhoek tekenen wilt u een regel toevoegen die iets over het doel van het programma zegt. Probeer het volgende:

```
5  REM *PROGRAMMA VIERHOEK-TEKENEN*
```

Als u het programma met LIST op het scherm brengt zal deze opmerking bovenaan verschijnen, verder negeert de computer de opmerking.

U kunt deze opmerkingen overbodig vinden, als u alleen maar programma's voor uzelf schrijft: u weet wat u doet en de computer heeft de uitleg niet nodig. Houd uzelf niet voor de gek. Zelfs als u het programma momenteel perfect beheerst weet u het na een maand of zes niet meer als u er iets aan wilt bijwerken. Ook als u het programma aan een kennis geeft moet er genoeg informatie instaan over wat u gedaan hebt.

Figuur 5.2 toont hoe opmerkingen gebruikt kunnen worden om een uitleg van het programma vierhoek tekenen te geven. Sterretjes zijn rond de titel gezet om hem te benadrukken. Lege REM opdrachten op regel 8, 24 en 34 zijn gebruikt om het programma in zijn belangrijkste delen te splitsen. De REM opdrachten bij 9, 25 en 35 geven weer wat de onderdelen doen. Door op deze manier doordacht opmerkingen te plaatsen wordt de samenhang van uw programma overzichtelijker.

Het opsporen van fouten (vlooienvijand!)

Hoe goed u ook oplet bij het schrijven van uw programma's, van tijd tot tijd zult u zeker problemen tegenkomen. Uw programma doet

niet precies wat u wilt of u krijgt een ERROR melding als u het programma laat lopen. Dit noemt men een vlo (*bug*) en u moet op vlooi-jacht (*debugging*).

Allereerst moet u de foutmelding controleren. Als bijvoorbeeld uw scherm er nadat u RUN intypte tijdens het programma vierhoek tekenen uitziet als op figuur 5.3 weet u waar u ongeveer moet zoeken. De computer was halverwege het programma toen er iets fout ging bij opdracht 60.

Wat wil deze foutmelding zeggen?

ERROR 3 AT LINE 60

De 3 in de boodschap is een code die het probleem beschrijft. Zoek dit in bijlage C van dit boek op. U ziet dan dat 3 betekent „waarde ongeldig”. Dit is een van de fouten op de Atari computers die het meeste voorkomen. Vaak worden ze veroorzaakt door het typen van een foutief getal of een getal dat te hoog is voor de plaats waar het gebruikt wordt. Dit is de foutmelding die u krijgt als u met PLOT of DRAWTO een punt met coördinaten buiten het bereik van het grafische scherm opgeeft.

Als u weet waar de fout zit moet u de lijst van het programma met LIST oproepen en controleren. In dit geval is regel 60 de meest voor de hand liggende plaats om de fout te zoeken. Maar een fout in een vorige regel kan ook problemen veroorzaken bij een regel verderop die zelf helemaal goed is.

Controleer het programma regel voor regel. Let vooral op details, een vergeten komma of een verkeerd gespeld woord kan een heel programma blokkeren.

Controleer of er geen regels ontbreken. Het kan gemakkelijk gebeuren dat een instructie bij het intypen vergeten wordt of dat door het typen van een verkeerd regelnummer er een gewist wordt. Overtuig u ervan dat er geen ongewenste regels in het programma voorkomen. U kunt bijvoorbeeld de NEW instructie vergeten zijn voordat u met het intypen begon.

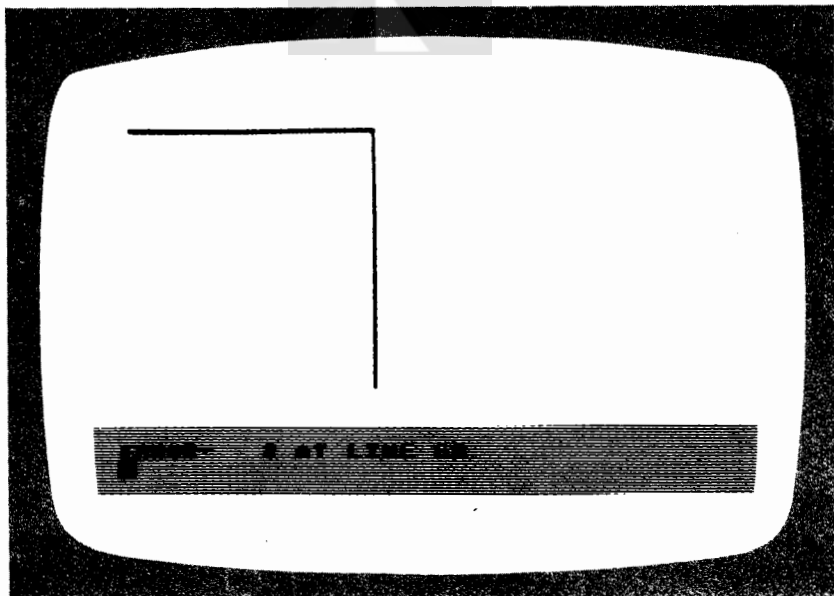
```
4 REM *****
5 REM * PROGRAMMA VIERHOEK-TEKENEN *
6 REM *****
8 REM
9 REM - KLAARZETTEN VIERHOEK-KLEUREN
10 GRAPHICS 7
20 COLOR 1
100 REM
101 REM -- NAAR LINKS BOVEN --
110 PLOT 10,10
120 REM
121 REM -- TEKEN DE VIER ZIJDEN --
130 DRAWTO 70,10
140 DRAWTO 70,70
150 DRAWTO 10,70
160 DRAWTO 10,10
```

Figuur 5.2: Gebruik van REM opdrachten om het programma meer leesbaar te maken.

Als u de fout niet kunt vinden, vraag dan een kennis het programma te controleren. Vaak stelt iemand anders een kleine fout eerder vast. Probeer uw programma aan een ander uit te leggen. Zo kunt u vaak een redenatiefoutje ontdekken dat u over het hoofd zag.

Nadat u alle mogelijkheden hebt doorgenomen denkt u ten einde raad misschien dat de fout aan de computer ligt. Dit is onwaarschijnlijk. Computers zijn erg betrouwbaar in het opvolgen van instructies en zijn zelden de oorzaak van een fout. Gewoonlijk is het een fout in de communicatie tussen u en uw computer: de computer leest iets anders dan u hebt bedoeld dat hij zou moeten lezen.

U moet pas de computer gaan verdenken, wanneer hij foutief reageert in verschillende programma's of in programma's die u al eerder uitgetest hebt. Als uw apparaat kapot is, zijn de problemen over het algemeen overduidelijk en zij beïnvloeden de werking in zijn geheel in plaats van een enkel programma. Neem met uw leverancier contact op als u denkt dat de computer kapot is.



Figuur 5.3: Een typische foutmelding (ERROR).

Niet opgeven. Het kan erg moeilijk zijn de fouten uit een uitgebreid programma te halen, raak dus niet gefrustreerd. Vaak blijkt het een domme fout, wanneer u hem vindt. Maar schaam u niet omdat u iets over het hoofd hebt gezien. 'n Vergissing is menselijk.

Opslaan op cassette

U kunt maar een programma tegelijkertijd in uw computergeheugen opslaan. Als u een ander programma wilt gebruiken, moet u het oude eerst uitwissen met NEW. Bovendien is het programma ook weg als u de computer uitzet.

Als u een programma hebt geschreven dat u regelmatig wilt gebruiken is dit een probleem. U moet het elke keer als u het wilt gebruiken opnieuw intypen, behalve als u de computer altijd aan kunt laten staan en slechts een programma gebruikt.

Met een Atari Program Recorder of Disk Drive kunt u uw programma's blijvend opslaan en ze in het computergeheugen terugladen als

u ermee wilt werken. Dit is erg handig en een waardevolle inverstering als u langere programma's gaat maken.

Op het ogenblik zal ik het nog niet over het gebruik van een programma recorder of disk drive hebben. Alle instructies die u daarvoor nodig hebt vindt u in hoofdstuk 9 en 10. Als u nu al het een en ander over uw disk drive of programma recorder wilt weten, sla dan de volgende hoofdstukken over. U kunt hoofdstuk 9 en 10 doorlezen zonder hoofdstuk 6 tot en met 8 te kennen.

Voor het gemak zal ik de eenvoudigste handelingen uitleggen: opslaan van een programma op cassette en het terugladen. Als u hieraan niet genoeg hebt of als u een disk drive gebruikt in plaats van een cassette, moet u de gedetailleerde instructies uit hoofdstuk 9 en 10 bestuderen.

Begin met het aansluiten van uw Atari Programma Recorder. U kunt zowel het model 1010 als het model 410 gebruiken; beide werken op iedere Atari computer. U moet de recorder in een stopcontact met 220 V en het speciale zwarte snoer in de grote opening aan de achterkant van de computer waarop PERIPHERAL staat steken. Lees hoofdstuk 9 door als u niet weet hoe dit moet.

U kunt nu de programma's uit het computergeheugen op een gewoon cassettebandje opslaan. Dat lukt met elk bandje, zelfs het goedkoopste. Leg het bandje in de recorder en spoel het terug tot aan het begin. Typ dan de volgende instructie:

CSAVE

Dit vertelt de computer dat u een programma op cassette wilt opslaan. Als u op RETURN drukt hoort u via de televisie-monitor twee piepen. Dit wil zeggen dat u op kunt nemen. Druk de PLAY en RECORD knoppen van de programma recorder in. Druk opnieuw op RETURN.

Uw computer begint te fluiten: dit is een teken dat op het bandje wordt opgenomen. Voor u betekent het niets, maar voor de computer bevat het alle informatie die hij nodig heeft voor het programma. Het opnameproces duurt ongeveer 20 seconden (als het programma erg lang is meer). De recorder stopt automatisch als hij klaar is en er verschijnt READY. Als het programma op band staat kunt u het

zonder zorgen uit het computergeheugen verwijderen door de computer uit te schakelen.

Om het programma terug te laden in het geheugen moet u ongeveer hetzelfde doen. Typ

CLOAD

en druk op RETURN. De computer piept nu eenmaal, wat betekent dat u de recorder kunt starten. Spoel het bandje terug en druk op de PLAY toets. Druk opnieuw op RETURN. Het bandje begint te lopen en de computer laadt het programma. Als dit klaar is stopt de recorder en de computer meldt READY. U kunt nu het programma met LIST bekijken en controleren of het goed geladen is. Daarna kunt u het met RUN opnieuw laten lopen. Pas op want het programma dat u laadt verdringt eventueel een programma dat in het geheugen zat.

Als deze instructies te kort zijn en problemen opleveren, lees dan hoofdstuk 9. De beschrijving daar is veel uitgebreider en bevat onder andere suggesties die van dienst zouden kunnen zijn. Deze aanwijzingen kunt u zien als eenvoudig begin.

Vrije oefeningen

1. Voeg aan het programma vierhoek teken op blz. 70-71 een opdracht toe zodat de laatste lijn van de vierhoek in het blauw verschijnt. U moet een opdrachtnummer tussen de 60 en 70 kiezen.
2. Voeg een regel aan hetzelfde programma toe om een diagonaal te krijgen van de linkerbovenhoek van de vierhoek naar de rechteronderhoek.
3. In het volgende programma staan drie fouten:

```
10 PRINT "HALLO DAAR"  
20 GRAPHICS 7  
30 COLOR 1  
40 PLOT 10,734  
50 DRAWTO 0,0  
60 DRAWTO 10
```

Zoek de fouten.



Samenvatting

In dit hoofdstuk hebt u gezien hoe u uw instructies kunt samenvoegen tot een programma. U kunt zoveel instructies als u wilt, opslaan in het computergeheugen en ze dan met RUN als een groep laten uitvoeren. U kunt de LIST instructie gebruiken om het programma dat u hebt opgeslaan nog eens te bekijken of er veranderingen in aan te brengen.

Programma's hebben vele aanwendingsmogelijkheden, zoals u al hebt gezien. Omdat ze te veranderen en vaker te gebruiken zijn, kunt u met programma's veel gecompliceerdere handelingen uitvoeren dan wanneer u elke instructie afzonderlijk zou moeten intypen. Als u een cassette recorder hebt, kunt u zelfs uw programma's blijvend opslaan om ze later weer te gebruiken.

In de volgende twee hoofdstukken leert u technieken waarmee dit concept soepeler wordt. Uw mogelijkheid om de computer te besturen zullen toenemen en u zult ontdekken waartoe hij in staat is.

Hoofdstuk 6

Het gebruik van variabelen

Tot nu toe waren de instructies die u aan uw computer gaf erg bepaald. Met PRINT gaf u het afdrukken van een woord aan en met PLOT het neerzetten van een punt. Als u bij PRINT een ander woord of bij PLOT een ander punt wilde gebruiken, moest u de hele instructie opnieuw typen.

U kunt uw instructies soepeler maken door ze minder bepaald te laten zijn. U kunt getallen vervangen door *variabelen*, als het ware een etiket dat staat voor een getal dat u kiest. Als u uw programma laat uitvoeren, moet u aangeven welke variabelen welke getallen voorstellen. Later kunt u het programma met andere waarden voor de variabelen laten uitvoeren.

Variabelen maken het mogelijk uw computer voor berekeningen te gebruiken. Uw computer kan snel en goed rekenen en drukt de resultaten af op het scherm. Dit kan u uren werk besparen. Aan het einde van dit hoofdstuk leert u hoe variabelen ook letters kunnen vertegenwoordigen net als getallen, zodat u ook met woorden kunt spelen.

In dit hoofdstuk zullen we diverse variaties op het vierhoekprogramma uitproberen. Het idee van het programma uit hoofdstuk 5 blijft in tact. We zullen echter voor de coördinaten variabelen gaan gebruiken. Dit maakt het veel eenvoudiger de vorm en stand van de vierhoek te veranderen.

Het opslaan van getallen

Computers werken over het algemeen liever met getallen dan met woorden. In de meeste instructies hebben we tot nu toe een of meerdere getallen gebruikt om onze wensen kenbaar te maken. Bij de PLOT opdracht moesten we bijvoorbeeld een coördinaten-paar opgeven.

Voor een vast getal zoals b.v. 46 hoeft u slechts de cijfers te typen. Over het algemeen kunt u het getal typen zoals het geschreven wordt. Is het groter dan 1.000 dan moet u de punt weglaten die de duizendtallen van de honderdtallen scheidt. Verder moet u een punt (.) gebruiken in plaats van een komma (,) om het breukgedeelte aan te geven.

Dit zijn allemaal *constanten*. U kunt een constante in elke opdracht gebruiken, maar hij wordt direct gebruikt in plaats van opgeslagen. Als u een getal opnieuw wilt gebruiken moet u het opnieuw typen.

Er is echter een manier om een getal in uw computer op te slaan zodat u dit opnieuw kunt gebruiken zonder het opnieuw te moeten intypen. Hiervoor moet u het getal een naam geven. Deze naam mag alles zijn, als hij maar met een letter begint.

U mag bijvoorbeeld een getal FRED noemen. Om dan het getal 7 op te bergen, hoeft u alleen te typen:

```
FRED=7
```

Als u op RETURN drukt zal de computer een plaats in het geheugen voor FRED reserveren. Daarna wordt het getal 7 op deze plaats opgeslagen, zodat u het bij z'n naam kunt noemen. U kunt niet zoals bij het gewone rekenen de vergelijking omdraaien. Als u typt

```
7=FRED
```

zegt u: „Pak het getal FRED en stop dit weg onder de naam 7”. Dit is voor de computer onzin en u krijgt dus een foutmelding.

De computer ziet het 'is-gelijk-teken' als een instructie. Als hij dit teken ziet, kijkt hij wat de waarde aan de rechterkant is en bergt dan de waarde op onder de naam links. Dit noemt men een *toekenningsopdracht*.

De naam FRED vertegenwoordigt nu het getal 7. U kunt dit zien door de computer het getal te laten afdrucken:

```
PRINT FRED
```

Als u dit doet zal de computer het getal 7 tonen. (Als u 0 krijgt of iets anders, hebt u waarschijnlijk een fout gemaakt: probeer de laatste twee instructies opnieuw.)

U kunt nu de naam FRED overal gebruiken waar u gewoonlijk het getal 7 gebruikte. U kunt in de vierkleuren stand komen door te typen

```
GRAPHICS FRED
```

U kunt het punt 7,7 neerzetten met

```
PLOT FRED,FRED
```

U kunt zelfs het getal 7 in een andere variabele genaamd JAN stoppen door deze gelijk te stellen aan FRED:

```
JAN=FRED
```

Laten we eens beter naar de opdracht

```
PRINT FRED
```

kijken. Toen u op RETURN drukte liet de computer de waarde zien die u FRED had genoemd. Ziet u dat het resultaat geheel anders is dan wat er gebeurd als u typt

```
PRINT "FRED"
```

Als u dit doet typt de computer het woord FRED op het scherm, in plaats van zijn waarde. Met de aanhalingstekens leest de computer „FRED” als een boodschap. Zonder de aanhalingstekens leest hij FRED als de naam voor een opgeslagen getal.

Het opgeborgene getal is een *variabele* omdat u de waarde kunt veranderen. Hiervoor moet u slechts een nieuwe waarde geven. De computer zal deze accepteren en de oude waarde door de nieuwe vervangen.

Om de waarde van FRED in 19 te veranderen moet u typen

FRED=19

De oude waarde (7) is verloren en FRED is vanaf nu 19. U kunt dit controleren met de instructie

PRINT FRED

FRED blijft 19 totdat u er een nieuwe waarde aan toekent of de computer uitzet.

Zolang u een variabele geen waarde geeft zal de computer ervan uitgaan dat deze 0 is. U kunt dit zien door een naam af te drukken die u nog niet gebruikt heeft:

PRINT IRENE

De computer zal met 0 antwoorden, omdat er aan IRENE nog geen waarde is toegekend.

Tot nu toe hebben we voornamen gebruikt zoals FRED, JAN, IRENE. Over het algemeen is het beter een naam te gebruiken die aangeeft wat de variabele vertegenwoordigt. Als u een variabele hebt die telt hoe vaak u iets doet, kunt u deze het beste TELLER noemen. Bij grafieken bent u gewend de coördinaten X en Y te noemen, zodat dit wordt

PLOT X,Y

Als u meer dan een punt neerzet, kunt u de variabelen verschillende namen geven die met X en Y beginnen. Bijvoorbeeld als u een lijn tussen twee punten trekt, kunt u het startpunt XSTART en YSTART en het eindpunt XEIND en YEIND noemen.

Met betrekking tot de namen die u mag gebruiken zijn er slechts een paar beperkingen. De naam mag zolang zijn als u wilt, maar moet met een letter beginnen. De rest mag cijfers bevatten maar geen speciale symbolen. U mag geen kleine letters of inverse-videotekens gebruiken in de naam van een variabele. De naam van de variabele mag

ook niet beginnen met een Atari instructiewoord zoals PRINT en PLOT. De computer weet dan niet hoe de variabele zich van de instructie onderscheidt.

Binnen deze richtlijnen zijn alle namen toegestaan. De namen in de linkerkolom van figuur 6.1 laten wat mogelijkheden zien. De rechterkolom toont enkele foutieve namen. U hoeft zich echter geen zorgen te maken want de meeste namen die u wilt gebruiken zijn toegestaan.

Bij deze experimenten met opgeslagen getallen hebt u de werking van variabelen gezien. U kunt een getal in elke variabele stoppen en daarna de naam van de variabele gebruiken in plaats van het getal. Als u de waarde wilt veranderen hoeft u alleen maar een nieuwe waarde op te slaan. De variabele zal de nieuwe waarde vertegenwoordigen totdat u hem verandert.

correct	foutief	(reden)
DATUM	123	Begint niet met een letter
NUMMER	TOON'STAS	Bevat speciaal teken
ZIJDE4	Willem	Geen hoofdletters
WACHT	HET GETAL	Meer dan een woord
N	GRAPHICS	Atari instructie woord

Figuur 6.1: Goede en foutieve namen van variabelen.

Variabelen in programma's

Als u alleen gewone instructies typt lijkt het alsof variabelen niet zo handig zijn. Welke instructie u ook intypt u kunt altijd de constante gebruiken.

In programma's zijn variabelen echter erg belangrijk. Hierdoor kunt u een algemene procedure die een bepaalde taak verricht schrijven zonder dat u de specifieke getallen moet kennen die u bij een bepaald geval wilt gebruiken. Terwijl u het programma laat lopen kunt u de specifieke getallen invoeren. Als u hetzelfde later met andere getallen uit wilt voeren hoeft u alleen maar RUN te typen.

We zullen eens bekijken hoe dat in z'n werk gaat. We gaan terug naar het programma vierhoek-tekenen uit hoofdstuk 4 en 5. Om te beginnen moet u de NEW instructie geven om het programma in het geheugen van uw computer te typen zoals het hier staat:

```
4   REM *****
5   REM *PROGRAMMA VIERHOEK-TEKENEN*
6   REM *****
8   REM
9   REM -- KLAARZETTEN VIERHOEK-KLEUREN
10  GRAPHICS 7
20  COLOR 1
100 REM
101 REM -- NAAR LINKS BOVEN --
110 PLOT 10,10
120 REM
121 REM -- TEKEN DE VIER ZIJDEN --
130 DRAWTO 70,10
140 DRAWTO 70,70
150 DRAWTO 10,70
160 DRAWTO 10,10
```

Dit programma is hetzelfde als dat op figuur 5.2. In de tweede helft werden alleen een paar opdrachtnummers veranderd zodat er ruimte ontstond voor toevoegingen tussen de nummers 20 en 100. Als u niet zoveel wilt typen kunt u ook de REM opdrachten overslaan. Ze hebben op de werking van het programma geen effect.

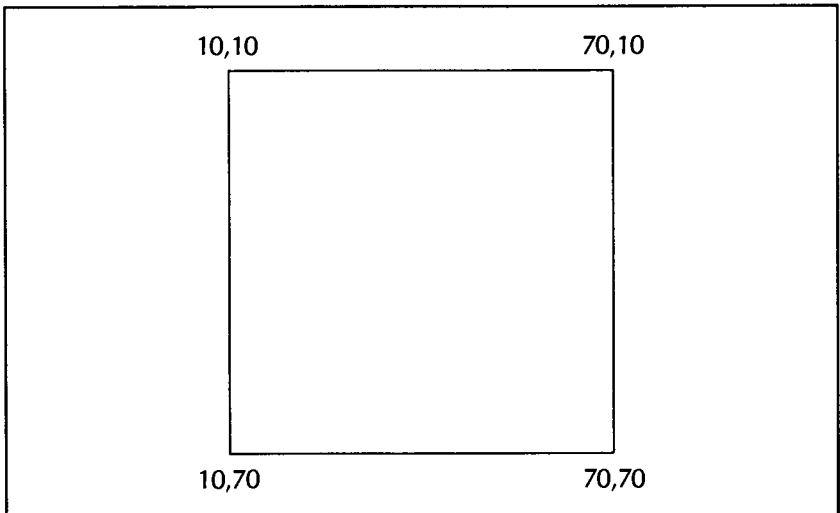
Bij deze versie van het programma was de vierhoek steeds even groot en stond altijd op dezelfde plaats. De gebruikte coördinaten van de hoeken waren constanten. De vierhoek begint bij de coördinaten 10,10 – een punt in de linker bovenhoek van het scherm. Elke zijde is naar een vast punt getrokken totdat de laatste zijde de vierhoek sluit. Figuur 6.2 toont de vaste coördinaten van de hoeken van de vierhoek.

We kunnen ons programma veelzijdiger maken zodat we er een vierhoek mee kunnen tekenen van elke grootte en vorm en op elke gewenste plaats. In plaats van vaste coördinaten voor de hoeken gaan we variabelen gebruiken.

Op figuur 6.3 ziet u de namen die we voor elk punt gaan gebruiken. Deze namen zijn zo gekozen dat ze de getallen die ze vertegenwoordigen verklaren. XLINKS bijvoorbeeld geeft de x-coördinaat aan de linkerkant van de vierhoek aan. Ziet u dat iedere variabele voor de aanduiding van twee hoekpunten wordt gebruikt: XLINKS is zowel de x-coördinaat in de hoek linksboven als linksonder.

We zullen nu eens beginnen met het veranderen van constanten in variabelen. In plaats van met NEW te werken en zodoende alles opnieuw te moeten typen, moet u nu alleen de toevoegingen intypen. Zoals u al weet wordt er een nieuwe regel in uw programma ingevoegd als u hem intypt als zich in het geheugen al een programma bevindt. Als het opdrachtnummer gelijk is aan een opdrachtnummer dat het programma al heeft, wordt de oude opdracht vervangen. Op deze manier is het mogelijk delen van het programma te veranderen zonder het hele programme opnieuw te moeten typen.

Het kan zijn dat als u een programma uit dit hoofdstuk hebt uitgevoerd het scherm in de grafische stand blijft staan. U kunt dan de opdrachten in het tekstraam aan de onderkant van het scherm intypen, maar meestal is het eenvoudiger naar de tekststand terug te keren. U kunt hiervoor zowel GRAPHICS 0 intypen als op RESET drukken.



Figuur 6.2: Vaste getallen geven u alleen een vierkant van een bepaalde maat.

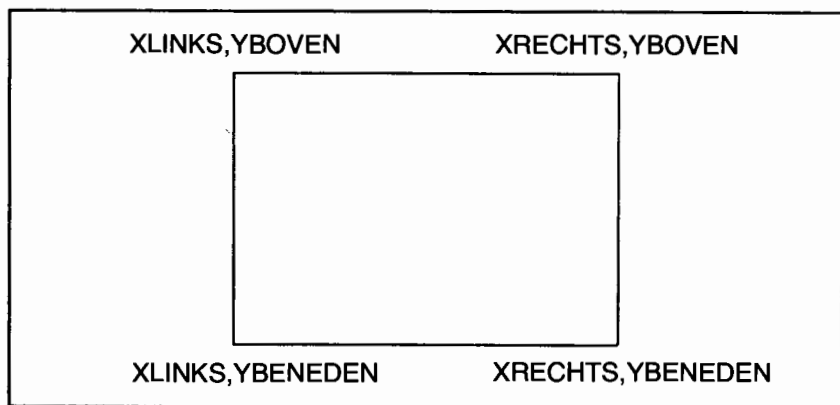
Voordat we de vier variabelen voor de hoekpunten kunnen gebruiken moeten we hun waarden opslaan. We kunnen een willekeurig getal nemen, maar we beginnen met de coördinaten die we al gebruiken: 10 en 70. Voeg de volgende vijf regels aan het programma toe:

```
30 REM – VARIABELEN KRIJGEN WAARDEN –  
40 XLINKS=10  
50 XRECHTS=70  
60 YBOVEN=10  
70 YBENEDEN=70
```

Nu we de waarden hebben toegekend kunnen we de PLOT en DRAWTO opdrachten opnieuw intypen:

```
110 PLOT XLINKS,YBOVEN  
130 DRAWTO XRECHTS,YBOVEN  
140 DRAWTO XRECHTS,YBENEDEN  
150 DRAWTO XLINKS,YBENEDEN  
160 DRAWTO XLINKS,YBOVEN
```

Deze regels zullen de PLOT en DRAWTO opdrachten in het oorspronkelijke programma vervangen.



Figuur 6.3: De hoekpunten krijgen deze variabele namen.

Maak een uitlisting (LIST) van het programma zodat u er zeker van bent dat alles goed is opgeslagen. Controleer de regels die u net hebt opgeslagen om er zeker van te zijn dat ze correct zijn. Het programma moet er precies zo uitzien als op figuur 6.4.

Als u er zeker van bent dat alles goed is kunt u het laten lopen (RUN). Het resultaat moet hetzelfde zijn als bij de oorspronkelijke versie. Dit komt omdat de variabelen die in de PLOT en DRAWTO opdrachten werden gebruikt precies dezelfde waarden (10 en 70) hebben als de vaste coördinaten uit de oorspronkelijke versie.

Met de variabele versie kunt u de vorm van de vierhoek veranderen door het wijzigen van de opdrachten die de waarden opslaan. Vervang opdracht 40 tot en met 70 door:

```
40 XLINKS=20
50 XRECHTS=140
60 YBOVEN=35
70 YBENEDEN=45
```



```
4 REM *****
5 REM * PROGRAMMA VIERHOEK-TEKENEN *
6 REM *****
8 REM
9 REM - KLAARZETTEN VIERHOEK-KLEUREN
10 GRAPHICS 7
20 COLOR 1
30 REM - VARIABELEN KRIJGEN WAARDEN -
40 XLINKS=10
50 XRECHTS=70
60 YBOVEN=10
70 YBENEDEN=70
100 REM
101 REM -- NAAR LINKS BOVEN --
110 PLOT XLINKS,YBOVEN
120 REM
121 REM -- TEKEN DE VIER ZIJDEN --
130 DRAWTO XRECHTS,YBOVEN
140 DRAWTO XRECHTS,YBENEDEN
150 DRAWTO XLINKS,YBENEDEN
160 DRAWTO XLINKS,YBOVEN
```

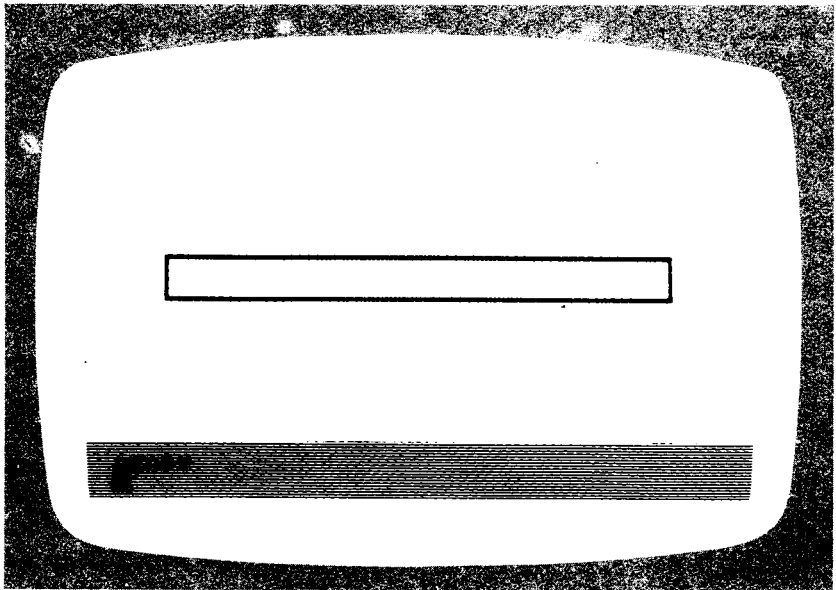
Figuur 6.4: De herziene versie van het programma vierhoek tekenen.

Als u het programma controleert op het scherm ziet u dat de oude opdrachten 40 t/m 70 door de nieuwe zijn vervangen. Als u het programma laat lopen verschijnt er nu een dunne horizontale rechthoek op het scherm zoals op figuur 6.5. Door het kiezen van andere waarden voor deze vier variabelen kunt u rechthoeken van allerlei afmetingen maken. Typ ook eens de volgende combinaties in en voer ze uit:

40	XLINKS=75	40	XLINKS=0
50	XRECHTS=85	50	XRECHTS=159
60	YBOVEN=10	60	YBOVEN=0
70	YBENEDEN=70	70	YBENEDEN=79

Met het linker voorbeeld krijgt u een dunne verticale rechthoek. Met de waarden die rechts staan tekent de computer een oranje rechthoek om het gehele grafische scherm.

Experimenteer ook eens met waarden die u zelf kiest. U kunt de gewenste variabelen veranderen en dan het programma opnieuw uit-



Figuur 6.5: Door het gebruik van variabelen kunt u aan een vierhoek elke willekeurige vorm geven.

voeren. De computer tekent de vierhoek volgens de afmetingen die u hebt aangegeven. U kunt voor de variabelen elk getal kiezen. Let erop dat u niet over de grens van de X en Y waarden gaat. X moet tussen 0 en 159 liggen en Y tussen 0 en 79. Als u er overheen gaat krijgt u een foutmelding.

Ziet u dat de variabele XLINKS drie maal verschijnt in de PLOT en DRAWTO regels van het programma terwijl er toch maar een nieuwe toekenningsoopdracht gewijzigd hoeft te worden om de waarde op alle drie de plaatsen te veranderen? Het maakt niet uit hoe vaak een variabele voorkomt. Door een nieuwe toekenning wordt de waarde overal gewijzigd.

Door variabelen te gebruiken hebben we een echte verbetering gekregen ten opzichte van eerdere versies van het programma vierhoek tekenen. We kunnen nu de computer exact de plaats van de vier hoeken en dus de grootte van de vierhoek aangeven. En wat nog belangrijker is; we kunnen een programma ontwerpen waarmee we algemene taken kunnen verrichten en dan bepaalde getallen veranderen zonder het gehele programma te moeten herschrijven. e algemene taken kunnen verrichten en dan bepaalde getallen veranderen zonder het gehele programma te moeten herschrijven.

De READ en DATA opdrachten

Als u de waarde van een variabele die u in een programma gebruikt wilt veranderen moet u teruggaan en de opdrachten daar waar die waarde is opgeslagen veranderen. Dit is geen probleem als u zelden verandert of als er maar een variable verandert moet worden. Als u echter vaak vele waarden in een programma moet veranderen is er een eenvoudigere manier. U kunt in een enkele opdracht aan verschillende variabelen een waarde toekennen en wel met de nieuwe instructies READ en DATA (lezen en gegevens).

READ en DATA worden altijd samen gebruikt en u kunt niet de ene instructie zonder de andere gebruiken. Met de READ opdracht geeft u de computer een lijst van variabelen die een waarde moeten krijgen. U typt daarna dienovereenkomstig een DATA opdracht met een dienovereenkomstige lijst van getallen die u wilt opslaan. De computer neemt de eerste waarde van de lijst uit de DATA opdracht

en kent deze toe aan de eerste variabele in de READ opdracht. Als er meer variabelen zijn zal de computer de DATA opdracht opnieuw volgen en de tweede waarde in de tweede variabele van de READ opdracht opslaan. Dit proces gaat door totdat alle variabelen gevuld zijn. U moet tenminste zoveel waarden in de DATA opdracht hebben dat alle variabelen hun waarde toegewezen kunnen krijgen. Als er te weinig getallen zijn krijgt u een foutmelding.

Om dit in de praktijk te zien werken veranderen we de regels 30 tot 70 van het programma vierhoek-tekenen met behulp van READ en DATA opdrachten. Typ de volgende regels:

```
30 REM – LEES WAARDEN IN VARIABELEN –  
40 READ XLINKS,XRECHTS,YBOVEN,YBENEDEN  
50 DATA 10,70,10,70
```

Als u dat hebt gedaan moet u nog twee lege regels toevoegen om de opdrachten uit de vorige versie die niet meer nodig zijn uit te wissen:

```
60  
70
```

Breng het opgezette programma in beeld. Het moet er uitzien als figuur 6.6. Als u het programma laat lopen krijgt u hetzelfde resultaat als een paar bladzijden terug - een grote oranje vierhoek op de linkerhelft van het scherm.

De vier variabelen in de READ opdracht komen precies overeen met de vier waarden bij de DATA. Als de computer bij de READ opdracht komt zet hij het eerste getal uit de DATA opdracht (10) op de plaats van de eerste variabele (XLINKS). Daarna komen de andere variabelen aan de beurt, 70 wordt in XRECHTS gelezen, 10 in YBOVEN en 70 in YBENEDEN. Deze vier regels hebben hetzelfde effect als de vier regels die werden vervangen:

```
40 XLINKS=10  
50 XRECHTS=70  
60 YBOVEN=10  
70 YBENEDEN=70
```



De READ en DATA opdrachten zijn echter eenvoudiger in te typen en te begrijpen.

Het grote voordeel van de READ en DATA opdrachten is dat u gemakkelijk waarden kunt wijzigen. Wilt u een smalle horizontale vierhoek tekenen, zoals op figuur 6.5 dan hoeft u alleen maar een nieuwe DATA opdracht te typen:

```
50 DATA 20,140,35,45
```

Als de computer de vierhoek tekent worden de hoeken bepaald door de nieuwe waarden.

U kunt met verschillende getallen in deze DATA opdracht experimenteren. Bij de allereerste voorbeelden moest u vier regels vervangen terwijl er bij de laatste nog maar een nieuwe DATA opdracht moet worden ingetypt. Probeer eens deze waarden:

```
50 DATA 75,85,10,70
```

en

```
50 DATA 0,159,0,79
```

U moet de oranje rechthoeken krijgen zoals bij de voorbeelden twee bladzijden terug. Experimenteer ook eens met eigen waarden.

INPUT: waarden toekennen vanaf het toetsenbord

Er is een derde mogelijkheid om getallen aan variabelen toe te kennen die zelfs nog flexibeler is dan de READ opdracht. Dit is de INPUT opdracht waardoor de computer midden in het programma stopt en u vraagt via het toetsenbord een waarde in te typen. Hiermee vervalt de noodzaak om getallen bij het schrijven van een programma te specificeren. Zo kunt u waarden geven op het moment dat u het programma laat lopen. Dit betekent ook dat u het programma met verschillende getallen kunt laten uitvoeren door alleen maar verschillende antwoorden te geven.

```

4 REM *****
5 REM * PROGRAMMA VIERHOEK-TEKENEN *
6 REM *****
8 REM
9 REM - KLAARZETTEN VIERHOEK-KLEUREN
10 GRAPHICS 7
20 COLOR 1
30 REM - LEES WAARDEN IN VARIABELEN -
40 READ XLINKS, XRECHTS, YBOVEN, YBENEDEN
50 DATA 10,70,10,70
100 REM
101 REM -- NAAR LINKS BOVEN --
110 PLOT XLINKS, YBOVEN
120 REM
121 REM -- TEKEN DE VIER ZIJDEN --
130 DRAWTO XRECHTS, YBOVEN
140 DRAWTO XRECHTS, YBENEDEN
150 DRAWTO XLINKS, YBENEDEN
160 DRAWTO XLINKS, YBOVEN

```

Figuur 6.6: Het programma vierhoek-tekenen met behulp van de READ en DATA opdrachten.

Om ervoor te zorgen dat de computer naar de eerste variabele XLINKS vraagt, kunt u deze regel in het programma vierhoek-tekenen aanbrengen:

```
40 INPUT XLINKS
```

Als de computer dit programma uitvoert stopt hij als hij bij deze opdracht aankomt en zet op het scherm een vraagteken. Dit noemt men een *prompt* en wil zeggen dat de computer van u informatie verwacht.

De meeste mensen gebruiken een PRINT opdracht direct voor de INPUT opdracht zodat ze weten op welke informatie de computer wacht. Ik stel voor dat u ongeveer het volgende doet:

```
40 PRINT "LINKER ZIJDE";INPUT XLINKS
```

Als u het programma met deze opdracht laat lopen stopt de computer en schrijft de volgende boodschap:

LINKER ZIJDE?

Nu weet u welke waarde de computer verwacht.

Zijn u de twee leestekens (:) voor het woordje INPUT op deze regel opgevallen? De betekenis van de dubbelepunt (:) hebt u al in hoofdstuk 3 geleerd. Hij scheidt twee opdrachten op dezelfde programma-regel. De puntkomma (;) heeft een ander doel. Als u een puntkomma aan het einde van een PRINT opdracht typt verhindert deze dat de computer aan het einde van de boodschap een nieuwe regel begint. Als de boodschap ten einde is zorgt de INPUT opdracht automatisch voor een vraagteken. Het resultaat is een leesbaar geheel.

Om de uitwerking van de INPUT opdracht te bekijken maken we een andere versie van het programma vierhoek-tekenen. Typ de volgende regels:

```
30 REM WAARDEN VANAF HET TOETSENBORD
40 PRINT "LINKER ZIJDE"; INPUT XLINKS
50 PRINT "RECHTER ZIJDE"; INPUT XRECHTS
60 PRINT "BOVENKANT"; INPUT YBOVEN
70 PRINT "ONDERKANT"; INPUT YBENEDEN
```

Als u de nieuwe versie van het programma in beeld brengt moet het er uitzien als op figuur 6.7. Let erop dat de variabelen in dit programma nooit een vooraf vastgelegde waarde krijgen?

Als u het programma laat lopen zult u merken dat het een beetje anders werkt dan bij het vorige. De computer springt in de grafische vierkleuren stand als gevolg van de instructies op regel 10 en 20. Daarna bij opdrachtnummer 40 verschijnt in het teksttraam de boodschap LINKER ZIJDE? en stopt hij. Dit betekent voor u dat u het eerste getal – de coördinaat van de linker zijde van de vierhoek (XLINKS) in moet typen. De computer wacht totdat u een getal in typt en op RETURN drukt. Nadat u daarmee klaar bent vraagt hij nog naar drie getallen voor de coördinaten van de andere drie zijden van de vierhoek. U kunt de antwoorden van figuur 6.8 geven.

```

4 REM *****
5 REM * PROGRAMMA VIERHOEK-TEKENEN *
6 REM *****
8 REM
9 REM - KLAARZETTEN VIERHOEK-KLEUREN
10 GRAPHICS 7
20 COLOR 1
30 REM WAARDEN VANAF HET TOETSENBORD
40 PRINT "LINKER ZIJDE";:INPUT XLINKS
50 PRINT "RECHTER ZIJDE";:INPUT XRECHTS
60 PRINT "BOVENKANT";:INPUT YBOVEN
70 PRINT "ONDERKANT";:INPUT YBENEDEN
100 REM
101 REM --- NAAR LINKS BOVEN ---
110 PLOT XLINKS,YBOVEN
120 REM
121 REM --- TEKEN DE VIER ZIJDEN ---
130 DRAWTO XRECHTS,YBOVEN
140 DRAWTO XRECHTS,YBENEDEN
150 DRAWTO XLINKS,YBENEDEN
160 DRAWTO XLINKS,YBOVEN

```

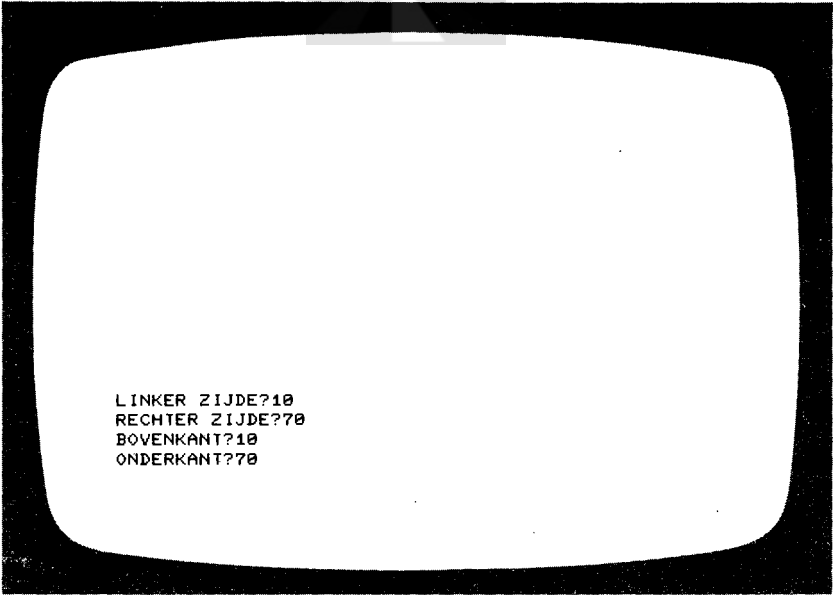
Figuur 6.7: Het programma vierhoek-tekenen met behulp van INPUT.

Als u de waarden intypt als het programma loopt kunt u het toetsenbord bedienen zoals u normaal doet. U kunt fouten corrigeren met de BACK-SPACE-toets en zoveel cijfers intypen als u maar wilt. Zodra u op de RETURN-toets drukt worden de antwoorden door de computer opgeslagen en kunt u niet meer terug. Natuurlijk kunt u het programma opnieuw laten lopen. (Als u het programma wilt stoppen voordat u het opnieuw laat lopen moet u op de BREAK-toets drukken.) Als u iets anders typt dan een getal zegt de computer:

ERROR 8 AT LINE 40

Als dit gebeurt moet u het programma van het begin af aan opnieuw laten uitvoeren.

Nadat u de vier antwoorden hebt gegeven tekent de computer de vierhoek. Als u de antwoorden uit figuur 6.8 geeft krijgt u dezelfde vierhoek als op figuur 6.2. Door met andere waarden te antwoorden kunt u de computer vierhoeken van verschillende vormen laten tekenen.



Figuur 6.8: Bij elke INPUT opdracht vraagt de computer welk getal hij moet gebruiken.

Met de INPUT versie van dit programma hebt u twee belangrijke dingen bereikt. Ten eerste hebt u een programma ontwikkeld dat tijdens de uitvoering vragen stelt. Het wacht op antwoorden en reageert dan op uw wensen.

Nog belangrijker is dat u een programma hebt dat u steeds opnieuw kunt uitvoeren. Elke keer als u het programma laat lopen zal het dezelfde algemene taak verrichten. De resultaten echter zijn afhankelijk van de getallen die u opgeeft. Dit is een erg belangrijke stap omdat de computer nu het programma gebruikt als een algemene probleemoplossing in plaats van een verzameling vaste instructies.

Berekeningen maken

U hebt nu drie verschillende manieren gezien hoe u getallen in variabelen op kunt slaan. Als u alleen een waarde aan een enkele variabele moet geven kunt u het gewone is-gelijk teken nemen. Als u diverse waarden tegelijk wilt opslaan werkt u misschien liever met de

READ en DATA opdrachten. Als u de waarden wilt geven op het moment dat het programma loopt kunt u de INPUT opdracht gebruiken.

U was tot nu toe erg beperkt in de mogelijkheden die u had om met de opgeslagen getallen te kunnen werken. U kon ze afdrukken of als coördinaten in de PLOT en DRAWTO opdrachten gebruiken. En dat was dan al alles.

Uw computer kan echter met de opgeslagen getallen ook berekeningen uitvoeren. Hij kan optellen, aftrekken, vermenigvuldigen en delen en nog een aantal gecompliceerde rekenkundige functies. Dit zit al allemaal in de machine. U hoeft alleen nog maar de gewenste formule in te geven.

U kunt de computer de formules ingeven die u uit de rekenkunde kent. Voor de standaardoperaties gebruikt u de volgende symbolen:

- + optellen
- aftrekken
- * vermenigvuldigen
- / delen
- ^ exponent (machtsverheffen)

Het sterretje (*) werd voor het vermenigvuldigen gekozen omdat de x voor de computer te verwarrend is. Om te delen gebruiken we de schuine streep (slash /) omdat hij op een deelstreepje lijkt. Deze vijf symbolen bevinden zich naast of in de buurt van de pijltoetsen rechts op het toetsenbord.

U kunt een formule intypen net zoals u dat bij het rekenen doet. Probeer eens

ANTWOORD=2+2

De computer beschouwt dit letterlijk als een opdracht. Hij neemt ANTWOORD als een variabele en kijkt naar wat er rechts van het is-gelijk teken staat om er een waarde aan toe te kennen. In dit geval ziet hij 2+2 en hij berekent dan de waarde 4. Het getal wordt dan in de variabele aan de linker kant van het is-gelijk teken opgeslagen,

dus in de variabele ANTWOORD. Om te controleren of hij het inderdaad heeft opgeslagen kunt u de instructie typen

PRINT ANTWOORD

Uw formules mogen zo gecompliceerd zijn als u maar wilt en mogen zowel variabele namen als getallen bevatten. U zou een van de volgende regels kunnen typen

$$X=15+17$$

$$KOSTEN=LOON*UREN$$

$$OPPERVLAKTE=LENGTE*HOOGTE/2$$

In het laatste voorbeeld bevat de formule meer dan twee getallen. Zoals in de wiskunde zal de computer over het algemeen de uitkomst van links naar rechts berekenen. Maar ook in dit geval gaat vermenigvuldigen voor delen. Om te zien hoe dit werkt geven we de volgende som op

$$GETAL=3*4+6/2$$

De computer zal eerst vermenigvuldigen ($3*4$) en dan delen ($6/2$). Nadat hij deze uitkomsten (12 plus 3) heeft opgeteld slaat hij de einduitkomst (15) op.

Als u deze volgorde niet wilt hebben of het te verwarrend vindt kunt u berekeningen tussen haakjes zetten. U zou de vorige opdracht bijvoorbeeld zo kunnen schrijven

$$GETAL=(3*4)+(6/2)$$

Het antwoord is hetzelfde. Als u de haakjes anders zet krijgt u ook een andere uitkomst. B.v.

$$GETAL=((3*4)+6)/2$$

Nu vermenigvuldigt de computer eerst dat wat er tussen de binnenste haakjes staat ($3*4$), telt daarna het antwoord (12) bij het getal 6 op en lost dan de buitenste haakjes op. Slechts dan kan hij het resultaat

door 2 delen en de oplossing (9) krijgen. Zoals u ziet kunt u een stel haakjes ook binnen een ander stel haakjes gebruiken.

Als u dit allemaal vrij ingewikkeld vindt hoeft u zich nog niet ongerust te maken. Ik zal me in dit boek tot eenvoudige vergelijkingen beperken zodat u alles kunt volgen. Als wiskunde voor u niets nieuws is kunt u uzelf leren met moeilijkere formules te werken.

U kunt een formule overal gebruiken waar u een getal of een variabele zou kunnen gebruiken. U kunt bijvoorbeeld typen

```
PRINT 3*5
```

Als u op RETURN drukt geeft de computer het antwoord (15). Weet u nog dat dit iets anders is dan

```
PRINT "3*5"
```

Omdat 3*5 tussen aanhalingstekens staat beschouwt de computer het als een boodschap die moet worden afgedrukt zoals u hem hebt geschreven en niet als een formule die berekend moet worden.

Om te oefenen gaan we nu een programma schrijven dat voor de variatie een blauwe driehoek op het scherm tekent (van vierhoeken hebben we genoeg, of niet?). Druk zoals altijd op RESET en typ NEW voordat u begint. Typ daarna het volgende programma:

```
10  GRAPHICS 7
20  COLOR 2:REM GROENE VERFKWAST
30  PRINT "STARTPLAATS X";:INPUT X
40  PRINT "STARTPLAATS Y";:INPUT Y
50  PLOT X,Y
60  REM GA 40 EENHEDEN NAAR RECHTS
70  DRAWTO X+40,Y
80  REM GA NAAR BENEDEN EN NAAR LINKS
90  DRAWTO X+20,Y+30
100 REM GA TERUG NAAR STARTPLAATS
110 DRAWTO X,Y
```

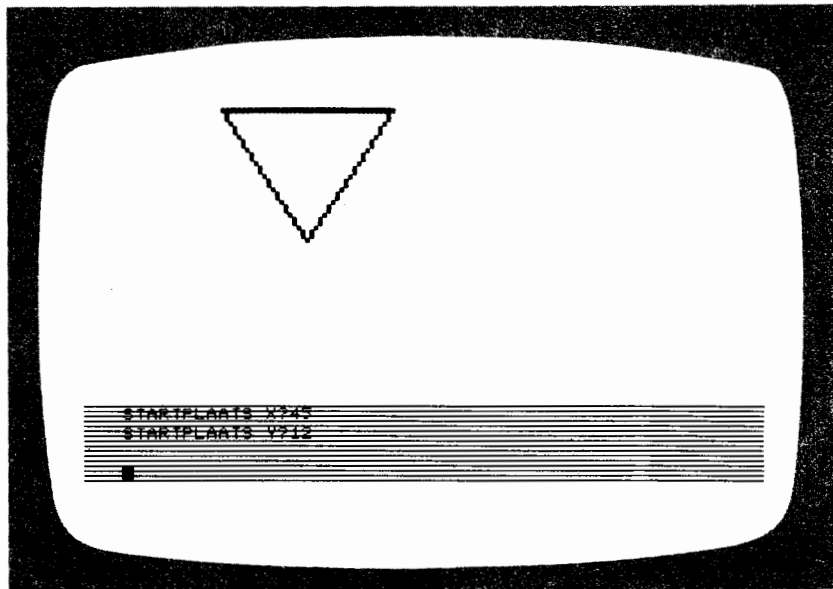
Als u nu dit programma laat lopen vraagt de computer naar de startplaatsen van de X en Y coördinaten. De getallen die u aangeeft bepa-

len de linker bovenhoek van de driehoek. Regel 50 zet dat punt neer, daarna trekt de rest van het programma lijnen naar de twee andere punten en terug naar het startpunt. Het resultaat is een driehoek zoals afgebeeld op figuur 6.9.

Laat het programma opnieuw lopen en geef de coördinaten andere waarden. In dit programma kunt u X tot 119 en Y tot 49 kiezen. Als u verschillende combinaties uitprobeert zult u merken dat de driehoek altijd dezelfde vorm en grootte heeft, maar naar verschillende posities op het grafische scherm geplaatst wordt. Dit komt omdat alle coördinaten als relatieve punten t.o.v. de linker bovenhoek zijn aangegeven. Als u dit punt verandert veranderen de andere mee.

Strings: opslaan van letters

Als laatste onderwerp in dit hoofdstuk zal ik uitleggen hoe u letters in uw computer kunt opslaan. Dit is een uitgebreid onderwerp dat op zich al een heel hoofdstuk zou beslaan. Ik kan er echter niet zo veel



Figuur 6.9: Het resultaat van het driehoek-programma .

aandacht aan besteden, maar zal het gedeeltelijk behandelen zodat u het principe begrijpt als u het in andere boeken tegenkomt.

U kunt woorden en letters in het geheugen van uw computer opslaan net zoals u getallen kunt opslaan. Hiervoor hebt u een speciaal soort variabele nodig, dat een *reeksvariabele* (string variable) genoemd wordt. Om een reeksvariabele vast te leggen moet u een dollarteken (\$) aan het einde van de naam zetten. De volgende variabelen zijn reeksvariabelen:

```
A$
NAAM$
ADRES$
TEKST$
```

Voordat u een reeksvariabele gebruikt moet u aangeven hoe lang de tekst is die opgeslagen wordt. De computer moet dit weten zodat hij voor ieder teken uit de reeks (string) in zijn geheugen een plaatsje kan reserveren. Reserveer meer plaats dan u nodig hebt en typ dan een opdracht zoals b.v.

```
10 DIM NAAM$(20)
```

DIM staat voor „dimensie”. Het zegt de computer 20 plaatsen voor de tekens uit de reeks NAAM\$ te reserveren.

U kunt nu woorden en tekens opslaan in een reeksvariabele net zoals u getallen kunt opslaan in gewone variabelen. U kunt een standaard toekenningsopdracht gebruiken door de tekst tussen aanhalingstekens te zetten:

```
20 NAAM$="ARTHUR"
```

U kunt READ en DATA opdrachten gebruiken:

```
20 READ NAAM$
30 DATA ARTHUR
```

Of een INPUT opdracht:

```
20 PRINT "WELKE NAAM";:INPUT NAAM$
```

Bij het laatste voorbeeld vraagt de computer WELKE NAAM? U kunt dan via het toetsenbord ARTHUR intypen.

Als u een boodschap in een reeksvariabele hebt opgeslagen kunt u er op verschillende manieren mee werken. U kunt hem afdrukken

40 PRINT NAAMS\$

U zou zelfs slechts een gedeelte van de reeks kunnen gebruiken door het noemen van de begin- en eindletter

60 PRINT NAAMS\$(2,4)

Als u deze opdracht laat uitvoeren drukt de computer alleen de tweede, derde en vierde letter van de naam af, dus RTH.

Met reeksen kunt u nog veel meer doen. U kunt ze splitsen en samenvoegen. Boodschappen kunnen op speciale manieren afgedrukt of opgeslagen worden zodat u ze later nog eens kunt gebruiken. Er zijn zoveel mogelijkheden mee dat ik er een boek over zou kunnen schrijven. Maar dan een boek voor gevorderden.

Vrije oefeningen

1. Voeg in de versie van het programma vierhoek-tekenen uit figuur 6.7 een INPUT opdracht toe zodat u een andere kleur dan oranje kunt kiezen. U moet een nieuwe variabele definiëren en opdracht 20 wijzigen.
2. Voeg instructies aan het einde van het driehoek programma toe (figuur 6.9) zodat de computer een David Ster tekent. U moet PLOT en DRAWTO opdrachten toevoegen om een tweede driehoek met zijn punt naar boven op de eerste te zetten.
3. Sla het getal 10 in de variabele J op. Typ hierna de volgende instructie:

J=J+1

Welke waarde zal J nu hebben? Controleer uw oplossing door de computer J te laten afdrukken (PRINT J).



Samenvatting

In dit hoofdstuk hebt u geleerd variabelen te gebruiken waarmee u een getal kunt opslaan om het later opnieuw te gebruiken. Met behulp van variabelen kunt u programma's eenvoudiger en algemener maken.

De eenvoudigste manier om een getal op te slaan is de variabele naam eraan gelijk te stellen. Het getal blijft opgeslagen totdat u het vervangt of de computer afzet.

Als u veel getallen op moet slaan of hun waarden regelmatig moet veranderen kunt u met READ de waarden uit de regel bij DATA lezen. Aan elke variabele in de READ opdracht wordt de overeenkomende waarde uit de DATA opdracht toegewezen. Om de waarden te veranderen hoeft u alleen maar de DATA opdracht te veranderen.

De eenvoudigste manier om getallen op te slaan is met INPUT. U vraagt de computer daarmee telkens te stoppen als hij die opdracht ontmoet en u kunt dan de gevraagde waarde via het toetsenbord intypen. U moet dan het getal typen en op RETURN drukken. De computer slaat de waarde op en gaat door met het programma.

Door het intypen van een formule maakt de computer berekeningen. U kunt ze zo gemakkelijk of moeilijk maken als u wilt. De uitkomst laat u afdrukken.

U kunt meer dan alleen getallen opslaan. Met de reeksvariabelen kunnen verbale boodschappen worden opgeborgen en op verschillende manieren worden afgedrukt.

In de volgende hoofdstukken leert u het gebruikmaken van variabelen en andere hulpmiddelen waardoor uw programma's meer bruikbaar en handiger worden.

Hoofdstuk 7

Het besturen van uw programma

In de vorige twee hoofdstukken hebt u geleerd hoe u een programma kunt schrijven. U kunt nu instructies in de computer typen en ze in het geheugen opslaan als een samengesteld programma. Met behulp van variabelen kunt u een programma soepeler maken zodat het een algemene taak kan uitvoeren waarbij het de waarden die u invult gebruikt.

Tot nu toe zat u echter vast aan de vanaf het begin tot het einde doorlopende structuur van het programma. De computer begon met de eerste opdracht en de rest volgde in volgorde. Als het programma af was stopte de computer en melde READY. Elke opdracht werd slechts één keer gelezen.

In dit hoofdstuk leert u hoe u de uitvoering van een programma kunt besturen. U kunt de computer vragen van het ene gedeelte van het programma naar het andere te springen of terug te keren naar opdrachten die al eens eerder werden gebruikt. U kunt ook een taak meerdere keren herhalen of de computer beslissingen laten nemen die gebaseerd zijn op informatie die u hem hebt gegeven. Tenslotte leert u nog hoe u uw programma's in kleinere *deelprogramma's* kunt splitsen die u als bouwstenen voor grotere kunt gebruiken.

GOTO: springen naar een andere opdracht

De eenvoudigste manier om een programma te besturen is met de GOTO opdracht. Hiermee krijgt de computer het signaal naar een andere opdracht te springen waar hij dan verder gaat. U wilt bijvoorbeeld een sprong maken van opdracht 20 naar 50 en alle tussenliggen-

de opdrachten overslaan. U kunt van regel 20 een GOTO opdracht maken:

```
20 GOTO 50
```

Als de computer deze regel leest gaat hij onmiddellijk naar regel 50 en gaat vandaar verder. De regels tussen 20 en 50 slaat hij over behalve als u hem er met een andere GOTO opdracht naartoe terugstuurt.

GOTO wordt meestal gebruikt om de computer naar het begin van een aantal opdrachten terug te sturen. Probeer eens

```
10 PRINT "HALLO"  
20 GOTO 10
```

Als u dit programma uitvoert zal de computer een verticale rij van HALLO's aan de linker kant van het scherm afdrukken. Eerst komt hij regel 10 tegen en drukt het woordje HALLO af. Daarna gaat hij naar regel 20. Die stuurt hem direct terug naar regel 10 zodat het woord nog een keer wordt afgedrukt. Als regel 10 voor de tweede keer klaar is leest de computer de volgende opdracht. Dit is weer regel 20 die hem weer terugstuurt naar regel 10. Dit gaat dan oneindig door.

De computer stopt niet eens als alle regels op het scherm vol zijn. Als u goed oplet ziet u dat het woordje HALLO dan op het onderste gedeelte van het scherm flinkt. De computer drukt nog steeds repeterend het woordje HALLO af en schuift de regels omhoog zodat er op het scherm plaats komt.

Als u nu op het toetsenbord een letter indrukt zult u zien dat de computer hierop helemaal niet reageert. Hij is nog steeds bezig uw programma uit te voeren en richt hier al zijn aandacht op. Totdat de computer klaar is en er READY verschijnt kunt u geen nieuwe instructie intypen. Dit programma zal echter nooit klaar zijn.

Zoiets heet een *oneindige lus*. De computer herhaalt zijn taak steeds opnieuw en gaat hiermee door totdat u de stroom uitschakelt of iets anders doet om hem te stoppen. Stoppen kunt u hem ook met RESET, maar dan wordt ook het scherm schoongemaakt.

De beste manier om een programma te stoppen is met de BREAK toets (afbreken). Op de modellen 600XL en 800XL bevindt hij zich aan de rechter bovenkant van het toetsenbord. Als u op deze toets drukt stopt de computer en meldt

STOPPED AT LINE 10

Gestopt op regel 10. READY verschijnt niet, maar de cursor is weer te voorschijn gekomen om te laten zien dat u weer letters kunt inty-
pen.

U zult ontdekken dat de BREAK-toets erg handig is om een programma te stoppen als het loopt. Het maakt niet uit wat de computer in uw programma doet, de BREAK-toets stopt hem en u kunt weer opdrachten vanaf het toetsenbord typen. Dit is erg belangrijk voor het geval dat u ooit in een oneindige lus raakt of het programma wilt afbreken. U kunt als u wilt het programma opnieuw starten op de plaats waar het stopte met de instructie

CONT

In de meeste gevallen zult u waarschijnlijk RUN gebruiken om het programma aan het begin te laten starten.

Oneindige lussen kunnen ook praktisch zijn. Als een taak steeds weer opnieuw moet worden uitgevoerd is het vaak het eenvoudigste het programma als een oneindige lus te schrijven en het met de hand te stoppen als het u goed dunkt.

Om te oefenen schrijven we nu een programma dat elk getal dat u in-
typt verdubbelt.

```
10 PRINT
20 PRINT "GEEF EEN GETAL ";:INPUT N
30 PRINT N;" MAAL 2 = ";N*2
40 GOTO 10
```

Het programma begint met het afdrukken van een blanco regel op het scherm zodat er een leesbaar geheel ontstaat. Regel 20 is een INPUT opdracht die u vraagt vanaf het toetsenbord een getal in te ty-

pen. Er wordt op uw antwoord gewacht. Daarna wordt uw getal in de variabele N gestopt. Regel 30 toont dan eerst het getal N, dan de tekst „ MAAL 2 = ” en dan de uitkomst van de berekening $2*N$. De GOTO opdracht stuurt de computer dan terug naar het begin van het programma om u naar een nieuw getal te vragen.

De PRINT opdracht op regel 30 ziet er misschien een beetje vreemd uit. Tot nu toe hebben we slechts een variabele of een boodschap na het woordje PRINT gebruikt. Hier hebben we er drie: de variabele N, de boodschap „ MAAL ” 2 = ” en het antwoord $N*2$. De computer begrijpt dit uitstekend en zet de drie informaties achter elkaar op het scherm. De puntkomma's (;) die de drie zaken scheiden zijn belangrijk daar ze de computer zeggen dat de drie informaties op het scherm op dezelfde regel moeten komen zonder extra spaties.

Als u dit programma laat lopen vraagt de computer GEEF EEN GETAL. Typ 3 en druk op RETURN. De computer reageert met

3 MAAL 2 = 6

```

READY
NEW

READY
10 PRINT
20 PRINT "GEEF EEN GETAL";:INPUT N
30 PRINT N;" MAAL 2 = ";N*2
40 GOTO 10
RUN

GEEF EEN GETAL?3
3 MAAL 2 = 6

GEEF EEN GETAL?23
23 MAAL 2 = 46

GEEF EEN GETAL?9354
9354 MAAL 2 = 18708

GEEF EEN GETAL?0
0 MAAL 2 = 0

GEEF EEN GETAL?■

```

Figuur 7.1: Met een GOTO opdracht kunt u de computer een taak laten herhalen.

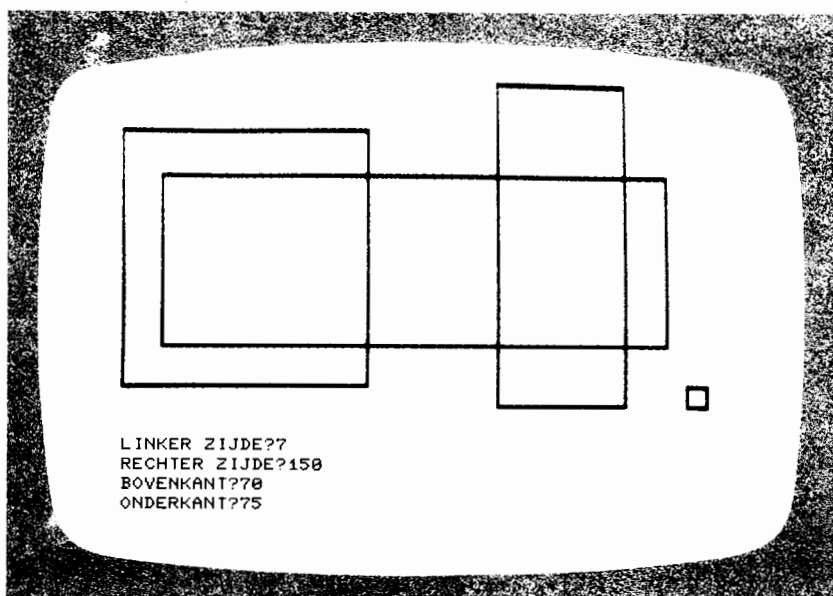
Hij gaat door en vraagt dan opnieuw naar een getal. U kunt getallen intypen en de computer verdubbelt ze zo vaak als u wilt. Figuur 7.1 toont het scherm zoals het er na een aantal antwoorden uit zou kunnen zien. Als u klaar bent moet u op **BREAK** drukken zodat u de besturing over het toetsenbord terugkrijgt.

Elk programma kan op deze manier worden herhaald. Een goed voorbeeld is het programma vierhoek-tekenen dat we in hoofdstuk 6 ontwikkelden. Overleg eens wat er zou gebeuren als u de volgende regel aan het programma op figuur 6.7. zou toevoegen

```
170 GOTO 30
```

Het programma wordt uitgevoerd zoals u dat al kent. Het vraagt naar de coördinaten voor de hoekpunten. Als u vier getallen hebt ingetypt tekent de computer onmiddellijk een oranje vierhoek.

In tegenstelling tot het programma op figuur 6.7 stopt de computer niet als hij hiermee klaar is. In plaats daarvan gaat hij terug en vraagt



Figuur 7.2: U kunt het programma vierhoek-tekenen wijzigen zodat u meer vierhoeken op het scherm kunt tekenen.

naar vier nieuwe getallen. Als u deze geeft tekent hij een tweede vierhoek zonder de eerste uit te wissen. U kunt zoveel vierhoeken tekenen als u wilt. Figuur 7.2 toont hoe het scherm er na een viertal herhalingen uit kan zien.

Als u goed naar het programma-overzicht op figuur 6.7 kijkt ziet u dat de GOTO opdracht de computer niet helemaal naar het begin van het programma terugstuurt. Dit komt omdat het niet nodig is de opdrachten GRAPHICS en COLOR in de regels 10 en 20 te herhalen. In feite zal de GRAPHICS OPDRACHT het scherm schoonmaken en het beeld dat opgebouwd werd uitwissen. Door GOTO 30 te gebruiken herhalen we alleen de regels die nodig zijn.

De GOTO opdracht is zo handig dat veel mensen hem te vaak gebruiken. Programma's die te veel heen en weer springen zijn moeilijk te volgen. U moet deze opdracht dus alleen gebruiken als het nodig is. Een programma wordt over het algemeen van boven naar beneden gelezen en te veel heen en weer is verwarrend.

FOR/NEXT lussen

U hebt nu de GOTO opdracht geleerd waarmee u uw computer een gedeelte van het programma kunt laten herhalen. Het probleem is dat de lus oneindig is. De computer herhaalt totdat u op BREAK drukt om hem te stoppen.

Waarschijnlijk wilt u de computer ook wel eens een aantal opdrachten een vast aantal keren laten herhalen en daarna met de rest van het programma doorgaan. Hiervoor zijn er twee nieuwe opdrachten: FOR en NEXT.

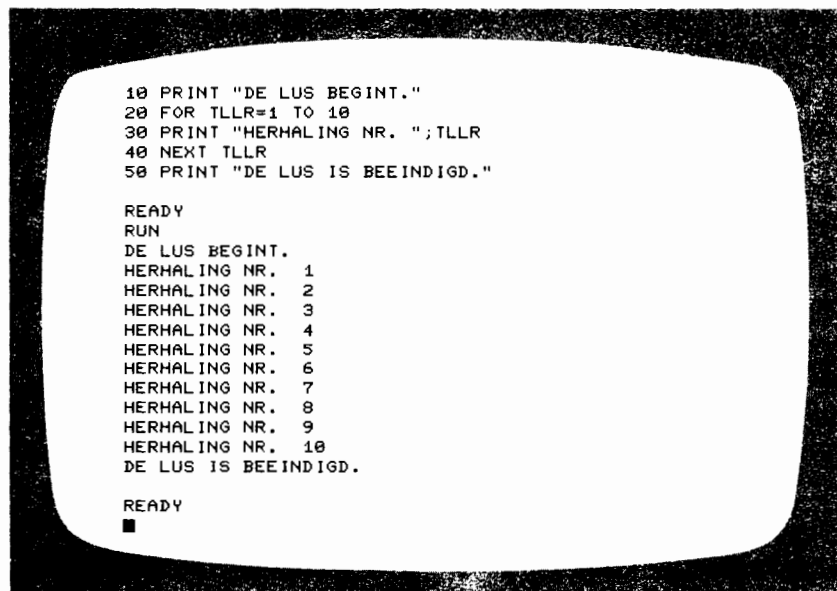
Zij werken als volgt. U plaatst een FOR opdracht aan het begin van de regel die u wilt herhalen en een NEXT opdracht aan het einde. Als de computer de FOR opdracht bereikt start hij een teller die het aantal herhalingen bijhoudt. De computer herhaalt alle opdrachten tussen FOR en NEXT totdat de teller de waarde heeft bereikt die u op gaf. Als hij NEXT bereikt heeft springt hij niet meer terug, maar loopt er langs en gaat verder met de rest van het programma.

Om te zien hoe er dit in de praktijk uitziet doen we het volgende

```
10 PRINT "DE LUS BEGINT."  
20 FOR TLLR=1 TO 10  
30 PRINT "HERHALING NR. "; TLLR  
40 NEXT TLLR  
50 PRINT "DE LUS IS BEEINDIGD."
```

Met de tekst van dit programma wil ik de werking verklaren. TLLR is de naam van een gekozen tellervariabele waarmee de computer het aantal herhalingen volgt. Als u het programma laat lopen ziet het eruit als op figuur 7.3.

De FOR opdracht in regel 20 zegt de computer de lus een aantal keren door te lopen en het aantal met de tellervariabele TLLR bij te houden. Het '1 TO 10' in de FOR opdracht legt de start- en eindwaarde van de teller vast en daarmee tegelijkertijd het aantal keer dat de lus wordt doorlopen. Zoals u uit het resultaat kunt zien start de teller-



```
10 PRINT "DE LUS BEGINT."  
20 FOR TLLR=1 TO 10  
30 PRINT "HERHALING NR. "; TLLR  
40 NEXT TLLR  
50 PRINT "DE LUS IS BEEINDIGD."  
  
READY  
RUN  
DE LUS BEGINT.  
HERHALING NR. 1  
HERHALING NR. 2  
HERHALING NR. 3  
HERHALING NR. 4  
HERHALING NR. 5  
HERHALING NR. 6  
HERHALING NR. 7  
HERHALING NR. 8  
HERHALING NR. 9  
HERHALING NR. 10  
DE LUS IS BEEINDIGD.  
  
READY  
■
```

Figuur 7.3: Een voorbeeld hoe een FOR/NEXT lus kan worden gebruikt om een PRINT opdracht te herhalen.

variabele met 1 in de eerste ronde door de lus en stijgt bij elke ronde met een punt. Als hij de bovenste grens (10) bereikt zegt de teller de computer dat de lus klaar is en de computer gaat met de opdracht achter NEXT verder.

Regel 30 toont een handige eigenschap van de FOR/NEXT lus:

```
30 PRINT "HERHALING NR. "; TLLR
```

Binnen de lus kunt u de teller als een gewone variabele gebruiken. In elke ronde heeft TLLR een andere waarde. Dit hangt af van het aantal keer dat de lus herhaald wordt. Binnen een goede opzet kunt u dit vaak in uw voordeel gebruiken.

De tellervariabele hoeft niet met 1 te beginnen. U kunt een willekeurig getal voor de begin – en eindwaarden kiezen. Zelfs negatieve getallen. Zet in regel 20 van dit programma ook eens andere waarden in zoals bijvoorbeeld

```
20 FOR TLLR=7 TO 14
20 FOR TLLR=329 TO 364
20 FOR TLLR=-3 TO 6
20 FOR TLLR=8 TO 5
```

Ziet u dat in het laatste voorbeeld de bovenste grens van de teller kleiner is dan de startwaarde. In dit geval voert de computer de lus slechts een maal uit en gaat dan achter de NEXT opdracht weer door.

U kunt de computer ook elke verhoging met een ander getal dan 1 uit laten voeren. U moet dan nog een ander getal aan de FOR opdracht toevoegen. Bijvoorbeeld

```
20 FOR TLLR=1 TO 10 STEP 2
```

Als u het programma zo laat lopen zult u merken dat er steeds twee getallen worden overgeslagen: 1, 3, 5, 7, 9. U zult tevens merken dat de teller nooit precies 10 wordt. De lus stopt bij 9 net voordat hij over het bovenste grensgetal gaat.

U kunt ook STEP (stap) gebruiken om de lusteller terug te laten tellen in plaats van vooruit. STEP moet dan een negatieve waarde hebben

```
20 FOR TLLR=10 TO 1 STEP -1
```

Als u het programma hiermee laat uitvoeren begint de teller met de waarde 10. Als hij het einde van de lus bereikt wordt er -1 bij de teller opgeteld wat betekent dat er een vanaf gaat. De lus herhaald zich met de waarde 9, 8 enz. Na 1 is de lus beëindigd en gaat het programma door.

U kunt de tellervariabele een willekeurige naam geven. U moet er echter zeker van zijn dat de naam in de FOR opdracht precies dezelfde is als die in de NEXT opdracht aan het einde van de lus. Als dit niet klopt krijgt u een foutmelding. Meestal worden eenvoudige namen voor FOR/NEXT tellers genomen b.v. I, J en K.

U kunt alle soorten opdrachten in de lus gebruiken. Zelfs een andere lus. Probeer eens het volgende programma

```
10 PRINT "DE LUSSEN BEGINNEN"  
20 FOR I=1 TO 3  
30 PRINT "BUITENSTE LUS --";I  
40 FOR J=1 TO 4  
50 PRINT "BINNENSTE LUS --";J  
60 NEXT J  
70 PRINT "BINNENSTE LUS BEEINDIGD"  
80 NEXT I  
90 PRINT "BEIDE LUSSEN ZIJN BEEINDIGD"
```

Het resultaat van dit programma is afgebeeld op figuur 7.4. Zoals u ziet wordt de buitenste lus tijdens dit programma drie maal herhaald waarbij iedere keer de waarde van zijn tellervariabele I afgedrukt wordt. In deze buitenste lus komt de computer iedere keer een binnenste lus tegen met een andere tellervariabele, J. Deze zegt hem de PRINT opdracht op regel 50 vier maal tijdens iedere ronde door de binnenste lus uit te voeren.

```

RUN
DE LUSSEN BEGINNEN
BUITENSTE LUS -- 1
BINNENSTE LUS -- 1
BINNENSTE LUS -- 2
BINNENSTE LUS -- 3
BINNENSTE LUS -- 4
BINNENSTE LUS BEEINDIGD
BUITENSTE LUS -- 2
BINNENSTE LUS -- 1
BINNENSTE LUS -- 2
BINNENSTE LUS -- 3
BINNENSTE LUS -- 4
BINNENSTE LUS BEEINDIGD
BUITENSTE LUS -- 3
BINNENSTE LUS -- 1
BINNENSTE LUS -- 2
BINNENSTE LUS -- 3
BINNENSTE LUS -- 4
BINNENSTE LUS BEEINDIGD
BEIDE LUSSEN ZIJN BEEINDIGD

READY
█

```

Figuur 7.4: Een lus binnen een lus kan een interessant resultaat geven.

Er is maar één maar bij dat handige idee van een lus in een lus. De binnenste lus moet helemaal binnen de buitenste zitten. U moet er zeker van zijn dat NEXT van de binnenste lus voor NEXT van de buitenste lus komt, zoals in het bovenstaande voorbeeld. Anders krijgt u een foutmelding.

FOR/NEXT lussen kunt u eigenlijk bij alle soorten programma's gebruiken. De meeste programma's herhalen op de een of andere manier opdrachten en bij veel herhalingen is het zelfs noodzakelijk. Ik geef u een paar voorbeelden waar u FOR/NEXT lussen kunt gebruiken.

Berekeningen Veel berekeningen worden vaak herhaald. Als u dan een lijstje met antwoorden op deze berekeningen zou hebben zou u ze niet steeds opnieuw uit hoeven te voeren.

Ga er bijvoorbeeld eens van uit dat u een onderdeel koopt voor f11,39 per stuk. U wilt er meer kopen, maar het aantal hangt af van de kosten. Dus schrijft u een programma om een tabel te maken:

```
10 REM ***KOSTEN ONDERDELEN***
20 PRIJS=11.39
30 REM DRUK KOP AF
40 PRINT "AANTAL","KOSTEN"
50 FOR I=1 TO 20
60 KOSTEN=I*PRIJS
70 PRINT I,KOSTEN
80 NEXT I
```

Als u dit programma uitvoert krijgt u een tabel zoals op figuur 7.5. U kunt nu aflezen hoeveel u voor een bepaald aantal onderdelen moet betalen. Ziet u dat er op de regels 40 en 70 een komma werd gebruikt in plaats van puntkomma. Een komma in een PRINT instructie betekent voor de computer dat hij de getallen netjes in kolommen moet plaatsen.

Grafieken Voor een groot aantal afbeeldingen zou u omstandigheden die zich herhalen op een bepaalde manier onder controle moeten hebben zodat u niet elke punt en elk lijntje opnieuw met de hand hoeft te tekenen. Bij elke figuur die u niet met een paar rechte lijnen kunt tekenen moet u gebruik maken van een FOR/NEXT lus.

U hebt de FOR/NEXT lus nodig als een groot figuur wilt tekenen. Probeer maar eens het volgende programma zodat u kunt zien hoe het werkt:

```
10 REM ***GROTE DRIEHOEK***
20 GRAPHICS 7
30 COLOR 1
40 FOR Y=0 TO 79
50 PLOT 0,Y
60 DRAWTO Y,Y
70 NEXT Y
```

Dit programma loopt iedere mogelijke Y waarde door, van 0 aan de bovenkant van het scherm tot 79 aan de onderkant. Bij elke Y zet het een punt aan de linkerkant van het scherm (coördinaat 0,Y), daarna wordt er een horizontale lijn Y eenheden lang getrokken. Het resultaat is een massieve driehoek zoals op figuur 7.6.

AANTAL	KOSTEN
1	11.39
2	22.78
3	34.17
4	45.56
5	56.95
6	68.34
7	79.73
8	91.12
9	102.51
10	113.9
11	125.29
12	136.68
13	148.07
14	159.46
15	170.85
16	182.24
17	193.63
18	205.02
19	216.41
20	227.8

READY
■

Figuur 7.5: Met een FOR/NEXT lus kunt u een tabel maken.

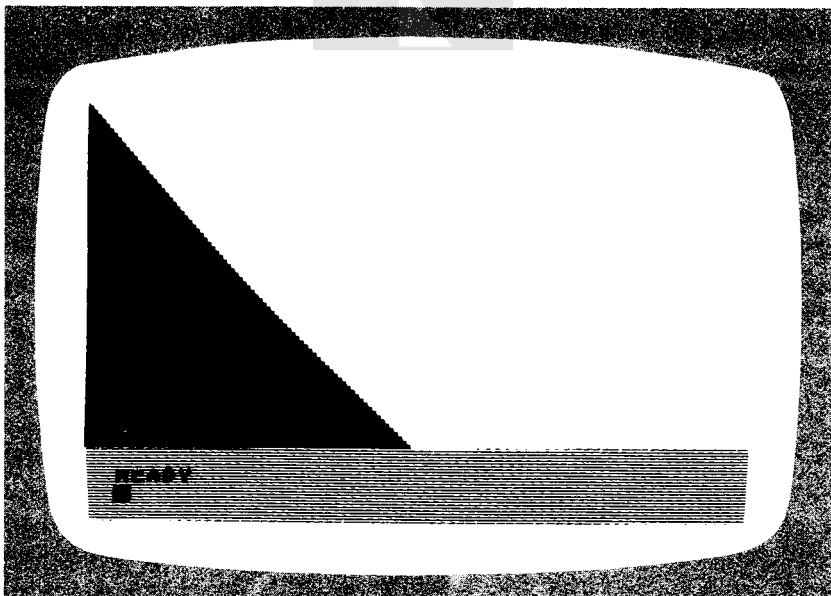
Vertragingen Iedere keer wanneer de computer een FOR/NEXT lus herhaalt kost dat een fractie van een seconde. Bij de meeste programma's zult u deze vertraging niet merken. Als u echter veel herhalingen achter elkaar wilt hebben loopt de tijd op.

Nu u dit weet kunt u dat gebruiken als u de computer langzamer wilt laten werken of een pauze in wilt lassen. Schrijf een FOR/NEXT lus met veel herhalingen op de plaats waar u de pauze wilt hebben:

```
20 FOR PAUZE=1 TO 5000
30 NEXT PAUZE
```

Er gebeurt niets als deze lus door de 5000 herhalingen gaat. Het kost alleen ongeveer tien seconden. Dit kan handig zijn in een programma als:

```
10 PRINT "GEDULDTEST"
20 FOR PAUZE=1 TO 5000
30 NEXT PAUZE
40 PRINT "GEHAALD!"
```



Figuur 7.6: Met de FOR/NEXT lussen kunt u een grote driehoek tekenen.

Als u dit programma laat lopen geeft de computer onmiddellijk het woord

GEDULDTEST

Daarna gaat hij in de „doe-niets lus” en wacht ongeveer 10 seconden voordat hij u feliciteert.

Wachtlussen zijn gebruikelijk in Atari programma's. Daar de FOR en NEXT opdracht in feite een geheel is schrijft men hem meestal op een regel, gescheiden door een dubbelepunt.

```
20 FOR PAUZE=1 TO 5000:NEXT PAUZE
```

Met deze en andere manieren kunt u met FOR/NEXT lussen uw computer op een speciale manier besturen. U kunt er een aantal opdrachten zo vaak u wilt mee herhalen en dan met de rest van het programma verdergaan. Als de computer met een lus bezig is houdt hij een teller bij die u bij u berekeningen kunt gebruiken. Hoe meer mogelijkheden u in uw computer ontdekt deste meer toepassingen zult u voor deze instructies vinden.

IF instructies: beslissingen nemen

U kunt ook de manier waarop de computer uw programma laat lopen bepalen met beslissingen die u hem laat nemen. Met de IF (als) opdracht kunt u hem laten beslissen of iets wel of niet gedaan moet worden afhankelijk van bepaalde voorwaarden.

Natuurlijk neemt de computer zelfstandig geen beslissingen. U zegt hem dat hij iets moet doen als er aan een bepaalde voorwaarde is voldaan en dat hij iets anders moet doen als aan deze voorwaarde niet is voldaan. Hij kiest zelf het antwoord naar gelang de informatie die u hem hebt gegeven. De computer kan niet denken. Hij volgt alleen de koers die aan de voorwaarde beantwoordt.

De IF opdracht bestaat uit twee delen: een voorwaarde en een actie. De twee delen van de opdracht zijn aangegeven met de woorden IF (als) en THEN (dan). De algemene vorm van de opdracht is

```
10 IF voorwaarde THEN actie
```

Als u deze instructie in een echt programma gebruikt moet u de woorden *voorwaarde* en *actie* vervangen door een in termen uitgedrukte vergelijking en de uit te voeren actie.

De voorwaarde is gewoonlijk een vergelijking tussen een variabele en een getal. U kunt bijvoorbeeld testen of de variabele NUMMER gelijk is aan 0. U zou kunnen beginnen met

```
10 IF NUMMER=0 THEN ...
```

De computer controleert de waarde van NUMMER. Als deze precies 0 is voert hij de actie uit. Indien niet dan slaat hij de actie over en voert de volgende opdracht van het programma uit.

U kunt een aantal verschillende rekenkundige vergelijkingen in uw IF voorwaarde gebruiken. De belangrijkste zijn

= gelijk aan
> groter dan
>= groter dan of gelijk aan
< kleiner dan
<= kleiner dan of gelijk aan
<> niet gelijk aan

U kunt ook twee of meer voorwaarden combineren met de woorden AND (en) en OR (of). Als u wilt controleren of een getal groter dan 5 en kleiner dan 10 is kunt u de volgende voorwaarde gebruiken

```
10 IF NUMMER>5 AND NUMMER<10 THEN ...
```

De computer voert de actie slechts uit als de variabele NUMMER een waarde tussen 5 en 10 heeft.

De actie die achter het woordje THEN volgt kan elke willekeurige Atari opdracht zijn. Als aan de voorwaarde in het IF gedeelte is voldaan zal de computer de instructie uitvoeren. Als er niet aan voldaan is negeert hij de instructie.

Hier volgen een paar voorbeelden van goede IF opdrachten:

```
10 IF I=10 THEN PRINT "NUMMER IS TIEN"  
20 IF STAND=0 THEN GRAPHICS 0  
30 IF GROOTTE >20 THEN GROOTTE=20
```

Met regel 10 verschijnt er een boodschap op het scherm als op dat punt de variabele I gelijk is aan tien. Als I een andere waarde heeft krijgt u geen boodschap.

Regel 20 gebruikt een IF opdracht om de computer in de GRAPHICS 0 stand te zetten (standaardtekst) omdat de computer bij zijn controle ervaart dat de STAND variabele voor standaardtekst 0 is.

Regel 30 toont een algemeen gebruik van de IF opdracht. Bij deze opdracht controleert de computer of de variabele GROOTTE boven de 20 is gegaan. Zo ja, dan zet de computer hem op 20. Na deze instructie kunt u er zeker van zijn dat GROOTTE niet groter is dan de maximale waarde.

Om te zien hoe een IF opdracht werkt proberen we een simpel programma dat u een getal tussen 1 en 100 laat raden. Typ dit programma in:

```
10 ANTWOORD=57  
20 PRINT  
30 REM LEGE REGEL
```

```
40 PRINT "GEEF EEN GETAL TUSSEN 1 EN 100"  
50 INPUT N  
60 IF N<ANTWOORD THEN PRINT "TE LAAG"  
70 IF N>ANTWOORD THEN PRINT "TE HOOG"  
80 IF N=ANTWOORD THEN PRINT "GERADEN!"  
90 GOTO 20
```

Regel 10 van dit programma geeft aan de computer het juiste antwoord. De rest van het programma is een oneindige lus die u vraagt een getal aan te geven dat dan met het juiste antwoord vergeleken wordt.

Als u het programma laat lopen is de eerste vraag

GEEF EEN GETAL TUSSEN 1 EN 100

U weet het antwoord al (57), maar probeer het ook eens met een paar andere getallen. Geef uw antwoord en druk dan op RETURN. De computer meldt zich met TE LAAG, TE HOOG of GERADEN, naargelang het getal dat u hebt ingetypt. De computer maakt zijn keuze overeenkomstig de IF instructies op de regels 60 tot 80 en drukt de bijbehorende tekst af. Regel 90 stuurt de computer terug naar het begin van de lus om een ander getal te vragen. Figuur 7.7 toont de resultaten van enkele gissingen. Als u klaar bent moet u op BREAK drukken om het programma te stoppen.

U kunt ook een andere uitkomst opgeven door regel 10 te veranderen. Dit kan zelfs een leuk spelletje worden. Voer een getal in door het op regel 10 in te typen en maak dan het scherm schoon. Laat nu het programma lopen. U zou een vriend of vriendin het getal kunnen laten raden dat u ingevoerd hebt.

IF opdrachten worden vaak gebruikt om de computer d.m.v. een GOTO opdracht te vragen onder bepaalde omstandigheden een gedeelte van het programma uit te voeren. In het programma „getallen raden” wilt u alleen naar regel 20 teruggaan als het antwoord fout is. Vervang de GOTO opdracht in regel 90 door de volgende IF opdracht:

```
90 IF N<>ANTWOORD THEN GOTO 20
```

De <> is een „niet-gelijk” voorwaarde en stuurt de computer naar twee verschillende gedeelten van het programma. Dit hangt ervan af of de waarden van de variabelen N en ANTWOORD gelijk zijn of niet. Als ze niet gelijk zijn springt de computer direct naar regel 20 terug en gaat vandaar verder. Als de getallen gelijk zijn is de voorwaarde ongeldig (onwaar) en hij slaat de GOTO opdracht over. Normaal zou hij met de volgende opdracht verder gaan, maar omdat het de laatste regel is stopt het programma in dit geval.

IF opdrachten met een GOTO actie maken het mogelijk uw programma in twee delen te splitsen waarbij de computer zelf de volgorde kiest afhankelijk van bepaalde omstandigheden. U zou bijvoorbeeld het volgende programma kunnen schrijven:

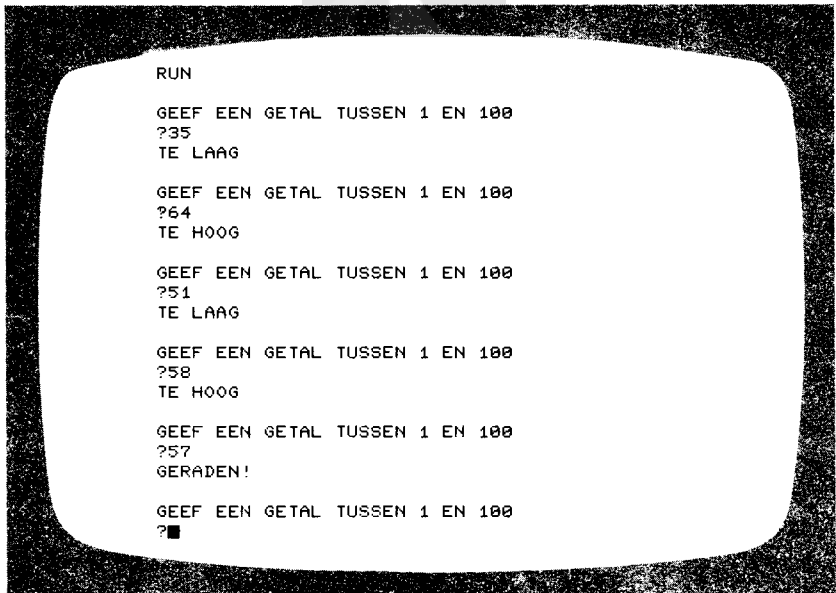
```
10 PRINT "GEEF EEN GETAL";:INPUT N
20 IF N=0 THEN GOTO 50
30 PRINT "INSTRUCTIE 30 UITGEVOERD"
40 GOTO 10
50 PRINT "INSTRUCTIE 50 UITGEVOERD"
60 GOTO 10
```

Dit programma vraagt een getal en test dan of het nul is. Als dit klopt springt de computer naar opdracht 50 en print een boodschap. Als het getal niet nul is negeert de computer de actie en gaat verder met opdracht 30. In beide gevallen komt hij terug op regel 10 na de PRINT boodschap gegeven te hebben.

Met de IF instructie kunt u gebruik maken van een bepaalde techniek die *gegevenscontrole* wordt genoemd. Als u een programma schrijft dat vanaf het toetsenbord om invoer vraagt zult u vaak een bepaald soort antwoord verwachten. U zou bijvoorbeeld een getal tussen 1 en 12 kunnen verwachten dat de maand van een jaar inhoudt. U kunt zinloze antwoorden voorkomen door een eenvoudige controle in te bouwen:

```
10 PRINT "WELKE MAAND";:INPUT MAAND
20 IF MAAND<1 OR MAAND>12 THEN GOTO 10
30 PRINT "GOED"
```

Als u dit programma laat lopen vraagt de computer naar het nummer van een maand. Als het antwoord zonder zin is stuurt de IF opdracht



Figuur 7.7: Weergave van het programma getallen raden.

de computer terug naar regel 10 om naar een andere waarde te vragen. Hij doet dit totdat u een getal tussen 1 en 12 aangeeft. In alle instructies die u na 30 plaatst is het zeker dat de variabele erin de waarde uit de in aanmerking komende maand krijgt toegewezen.

Met IF opdrachten kunt u nog veel meer doen. Als u uw computer ook moeilijkere taken uit laat voeren moet u altijd beslissingen inbouwen. Met behulp van de IF opdrachten kunt u de keuzemogelijkheden gemakkelijk sturen.

GOSUB: Subprogramma's

Er is nog een andere waardevolle manier om de ononderbroken loop van uw programma te veranderen. Dat gaat met *subprogramma's* (*subroutines, subprograms*). Het gaat daarbij om een serie van instructies die u apart van de rest kunt zetten en als een eenheid op elk gewenst tijdstip kunt gebruiken.

Om een subprogramma te gebruiken moet u het oproepen (*call*) met een GOSUB instructie. Bijvoorbeeld

```
40 GOSUB 1000
```

Dit heeft dezelfde uitwerking als een GOTO instructie; de computer wordt naar opdracht 1000 gestuurd. Met een GOSUB instructie waarschuwt u de computer echter dat u nadat het subprogramma afgewerkt is terug wilt komen.

Het subprogramma zelf is hetzelfde als elke andere groep van opdrachten, behalve dat het eindigt met de RETURN instructie. Als de computer deze laatste instructie krijgt merkt hij dat het subprogramma af is en gaat terug naar de opdracht die na GOSUB komt. Op deze manier kunt u de computer opdragen naar een bepaalde regel te gaan, daar een aantal instructies uit te voeren en daarna weer naar zijn oorspronkelijke plaats terug te keren.

Het volgende programma laat zien hoe zoiets werkt:

```
10 PRINT "HOOFDPROGRAMMA BEGINT"  
20 GOSUB 1000  
30 PRINT "HOOFDPROGRAMMA GAAT DOOR"  
40 END  
1000 REM VOORBEELD SUBPROGRAMMA  
1010 PRINT "NU IN SUBPROGRAMMA"  
1020 RETURN
```

Als u dit programma laat lopen leest u het volgende:

```
HOOFDPROGRAMMA BEGINT  
NU IN SUBPROGRAMMA  
HOOFDPROGRAMMA GAAT DOOR
```

Zo kunt u de programmastroom volgen. De computer begon met regel 10, het begin van het hoofdprogramma. Bij regel 20 sprong hij naar het subprogramma en drukte de tweede boodschap af. Bij de RETURN opdracht was het subprogramma afgelopen en de compu-

ter keerde terug naar regel 30 van het hoofdprogramma omdat deze regel direct achter de GOSUB opdracht stond die het subprogramma opriep.

De END instructie in regel 40 is erg belangrijk. Het geeft het einde van het hoofdprogramma aan en zegt dat de computer moet stoppen. Als deze regel er niet zou staan, zou de computer met regel 1000 van het subprogramma verder gaan omdat dat dan de volgende opdracht is. Dit is alleen dan de bedoeling wanneer het subprogramma nog een keer moet worden uitgevoerd.

U kunt zoveel verschillende subprogramma's maken als u wilt. U moet er alleen op letten dat ze allemaal verschillende opdrachtnummers hebben en met RETURN eindigen. Hoewel u in de manier waarop u de regels nummert vrij bent is het aan te bevelen bij de subprogramma's hoge getallen te kiezen zoals bijvoorbeeld hier 1000. Op deze manier ontstaat er in het hoofdprogramma ruimte voor het geval dat u het op een later tijdstip nog uit wilt breiden. Ook is het over het algemeen aan te raden elk subprogramma met een opmerking (REM opdracht) die de werking beschrijft, te laten beginnen.

In een subprogramma kan ieder soort instructie voorkomen. U kunt zelfs een andere GOSUB opdracht gebruiken om een tweede subprogramma op te roepen. Als u daarmee werkt zal RETURN van het tweede subprogramma de computer terugsturen naar de GOSUB opdracht in het eerste. Hij gaat dan daar door, maakt het eerste subprogramma af en keert dan terug naar het hoofdprogramma om het te beëindigen.

Het zou kunnen dat u geneigd bent om GOTO te gebruiken in plaats van RETURN om naar het hoofdprogramma terug te keren. Doe dat niet. Het is wel mogelijk, maar meestal raken de computer en u in verwarring. Zorg dat uw subprogramma's compleet zijn en let er goed op dat er aan het einde een RETURN staat.

Waarom gebruiken we subprogramma's? Er bestaan een aantal redenen waarom u uw programma's in stukjes verdeelt. Ik noem er een paar:

- *Een groep opdrachten vaker gebruiken.* In grote programma's moet een bepaalde taak vaker herhaald worden. Als u deze taak slechts op een bepaalde plaats een aantal keren wilt laten herhalen

gebruikt u een FOR/NEXT lus. Als u echter een opdracht uit een vroeger gedeelte van het programma opnieuw nodig hebt moet u de FOR/NEXT lus op iedere plaats waar u hem wilt gebruiken overtypen. In plaats hiervan vat u de taak samen in een subprogramma. En elke keer als u de taak nodig hebt kunt u hem dan met een GOSUB instructie oproepen.

- *Een programma opsplitsbaar maken.* Als u probeert lange programma's in totaal te overzien zou de moed wel eens in uw schoenen kunnen zakken. Door het programma in beter hanteerbare eenheden te splitsen maakt u het begrijpelijk. Sommige mensen schrijven hun hoofdprogramma als een serie van GOSUB opdrachten waardoor de computer op zijn beurt naar deze fragmenten wordt verwezen.
- *Het ontwikkelen van een subprogramma-bibliotheek.* Op een gegeven moment zult u er behoefte aan hebben een aantal subprogramma's samen te stellen die u als onderdelen voor verschillende programma's kunt gebruiken. Als u al deze routines op een cassette of diskette opslaat kunt u op het moment dat u ze in een programma wilt gebruiken in het computergeheugen laden.

We zijn nu aan het einde van de besturingsinstructies gekomen. Met de opdrachten die u in dit hoofdstuk hebt geleerd bent u goed op weg een vakkundig computerprogrammeur te worden.

Vrije oefeningen

1. Schrijf een programma dat een getal vraagt en dan zegt dat het positief, negatief of nul is. U hebt een INPUT en een IF opdracht nodig. Onthoud dat een negatief getal met een min-teken moet worden ingetypt.
2. Schrijf een subprogramma dat het scherm schoonmaakt door met de PRINT opdracht een ESC en CONTROL/CLEAR af te drukken. Koppel het dan aan een ander programma om het beeld op het scherm overzichtelijker te maken.
3. Schrijf een FOR/NEXT lus om de som van de getallen van 1 tot 100 te berekenen. Een wenk: gebruik een variabele SOM en tel er telkens de tweede variabele in elke ronde in de lus bij op.



Samenvatting

In dit hoofdstuk hebt u verschillende manieren geleerd hoe u uw computer kunt besturen. U hoeft zich niet langer alleen maar binnen een simpel van het begin naar het einde lopende programma te bewegen. U kunt heen en weer springen, lussen doorlopen, beslissingen nemen en subprogramma's maken.

De GOTO instructie zegt de computer naar een andere opdracht te springen. U kunt binnen het programma vooruit en achteruit springen en als u wilt oneindige lussen maken. Dit is vaak een eenvoudige manier om taken die herhaaldelijk voorkomen uit te voeren.

Met FOR en NEXT instructies kunt u de computer een groep opdrachten een bepaald aantal keren laten uitvoeren en daarna met de rest van het programma doorgaan.

Met IF instructies kunt u tijdens het programma beslissingen nemen. Op elk tijdstip kunt u de computer een voorwaarde laten testen. Als aan de voorwaarde wordt voldaan voert de computer vanwege uw IF opdracht een actie uit. Hij kan een tekst afdrukken, een waarde veranderen of naar een ander gedeelte van het programma gaan. Het kan allemaal.

De GOSUB instructie geeft u een andere manier om de volgorde van het programma te bepalen. U kunt een lang programma in stukjes verdelen en die dan daar gebruiken waar u ze nodig hebt. Hierdoor is het schrijven van programma's eenvoudiger omdat u zich slechts met een ding tegelijk bezig houdt.

We zijn nu aan het einde gekomen van de standaard programmeertechnieken die in dit boek behandeld worden. Hoofdstuk 8 behandelt de geavanceerde grafische eigenschappen van uw Atari computer in vogelvlucht. Zo zult u nog meer van zijn mogelijkheden leren kennen.

Hoofdstuk 8

Geavanceerde grafische technieken

In hoofdstuk 4 hebben we al kort over het grafisch systeem van Atari gesproken. U hebt de POSITION instructie leren kennen waarmee overal op het scherm tekst kan worden geplaatst. U ontdekte hoe er op het scherm punten kunnen worden gezet, in de vierkleuren stand lijnen kunnen worden getrokken en figuren kunnen worden gemaakt.

In dit hoofdstuk zullen we het grafisch systeem iets uitgebreider behandelen. U leert kleuren van de achtergrond en van de grafische verfkwasten bepalen. U komt erachter hoe het tekstraam aan de onderkant van het scherm verwijderd kan worden zodat u ook grotere figuren kunt tekenen. En tenslotte ontdekt u via een kennismaking met de andere grafische standen andere manieren om het scherm te gebruiken en in te delen.

Het is niet mogelijk in deze paar bladzijden het gehele grafische systeem van Atari te beschrijven. Ik wil u echter een voorproefje geven van de prachtige dingen die u met de grafische mogelijkheden kunt doen. Als u er meer over wilt weten zou u een van de technische boeken moeten raadplegen die aan Atari zijn gewijd.

Het onderwerp van dit hoofdstuk is erg moeilijk. De uitgebreide mogelijkheden van het grafisch systeem van Atari zijn erg interessant, maar kunnen vooral in het begin erg verwarrend werken. Raak echter niet in paniek als u de draad verliest. Ga met de rest van het boek door en kom hier terug als u het gevoel hebt dat u het beter kunt begrijpen.

Het veranderen van kleur met SETCOLOR

Tot nu toe hebt u slechts een paar kleuren gebruikt: witte en blauwe letters, de achtergrond en vier grafische verfkwasten (oranje, groen, blauw en zwart). U kunt echter uit een serie van 16 kleuren kiezen. Binnen elke van deze 16 kleuren kunt u uit *16 nuances (luminances)* of tinten kiezen. Er staan in totaal 256 kleuren ter beschikking. Zelfs als u een zwart/wit televisie hebt kunt u de helderheid van de grijstinten bepalen.

U heeft de COLOR instructie gebruikt om uit de vier grafische verfkwasten te kiezen, maar de afgestelde kleur (oranje, groen, blauw of zwart) kon u niet veranderen. U kon begrenst uw kleuren kiezen door van kwast te verwisselen, maar u had er geen volledige controle over.

De COLOR instructie heeft een partner die SETCOLOR heet. Die kan elke gewenste kleur aan een verfkwast geven. U kunt bijvoorbeeld de eerste verfkwast wit maken in plaats van oranje. Hoewel u niet meer dan vier kleuren tegelijkertijd kunt gebruiken, kunt u uit 256 beschikbare mogelijkheden kiezen.

Verwar COLOR niet met SETCOLOR. COLOR bepaalt welke verfkwast van de vier u gebruikt. Dus als u COLOR 2 kiest is dat in feite verfkwast 2. U zegt als het ware „Neem deze verfkwast en blijf hem gebruiken totdat ik een andere wil hebben”. SETCOLOR echter bepaalt de verfkleur van de kwasten. Het betekent in feite „Maak deze verfkwast schoon en stop hem in een nieuwe kleur”. Anders dan bij een gewone verfkwast gaat SETCOLOR ook terug en verandert alle al door hem geverfde punten in de nieuwe kleur.

De SETCOLOR instructie bestaat altijd uit drie getallen zoals bijvoorbeeld

SETCOLOR 2,9,4

Elk getal achter het woordje SETCOLOR heeft een speciale betekenis. Figuur 8.1 geeft een overzicht van het gebruik van de drie getallen in de vierkleuren stand.

Het eerste getal (A) geeft het *kleurenregister* aan dat u wilt veranderen. Een kleurenregister is een plaats waar de computer een code op kan slaan die een van de kleuren van het scherm bepaalt. Eenmaal opgeslagen verandert de nieuwe code de kleur van de verfkwast die bij dat register hoort zonder uitwerking te hebben op de andere.

De Atari computers hebben vijf kleurenregisters die van 0 tot en met 4 genummerd zijn. Jammer genoeg hebben ze een verschillende betekenis, afhankelijk van de gebruikte grafische stand. Register 2 bijvoorbeeld, stelt de kleur van een verfkwast vast in de vierkleuren grafieken, maar in de tekststand bestuurt het alleen de achtergrond. Het leren van de betekenis van al deze standen is vrij moeilijk, maar de belangrijkste zult u vrij snel onder de knie hebben.

De eerste kolom in figuur 8.1 geeft de werking aan van elk van de registers in de vierkleuren grafische stand. Is het u opgevallen dat de registers in de SETCOLOR instructie anders genummerd zijn dan de „verfkwastnummers” die u in uw COLOR opdrachten hebt gebruikt. (Dit lijkt verwarrend, maar u kunt het onthouden omdat de „wisborstel” in de COLOR opdracht altijd 0 is en bij de vier SETCOLOR registers altijd het *laatste* nummer, dus 4.) Figuur 8.2 geeft de overeenkomstige COLOR en SETCOLOR codes en de standaard afgestelde kleuren van de vijf registers. Kies de verfkwast die u wilt gebruiken uit de COLOR kolom en neem dan het overeenkomstige SETCOLOR register om hem te veranderen. Register 3 (rood) wordt in deze stand niet gebruikt, maar is in sommige andere standen wel belangrijk. De standaard afgestelde kleuren van de registers zijn in elke stand hetzelfde.

Het tweede getal in uw SETCOLOR instructie bepaalt de kleur. In deze stand kunt u uit de 16 kleuren kiezen die in kolom B van figuur 8.1 staan. U zult merken dat de kleuren als een eindeloze regenboog zijn gerangschikt. Als u door de kleuren 0 tot en met 15 loopt en dan teruggaat naar 0 zult u hierin een bijna continue cyclus zien. Dit komt in programma's vaak van pas.

Het derde getal in de SETCOLOR instructie bepaalt de *helderheid* van de kleur. Alle 16 kleuren kunt u in 8 verschillende tinten krijgen. De laagste, 0, is bijna zwart terwijl de hoogste, 14, erg licht is. Er tussenin kunt u de helderheid aan uw behoefte aanpassen. De computer

SETCOLOR A, B, C		
A: Kleurenregister	B: Kleur (tint)	C: Helderheid
0 Verfkwast #1	0 Grijs	0 Zeer donker
1 Verfkwast #2	1 Licht oranje	2 ↑
2 Verfkwast #3	2 Oranje	4 ↑
3 (niet in gebruik)	3 Rood oranje	6 ↑
4 Wisborstel (#0) en achtergrond	4 Rose	8 Middel
	5 Rose-paars	10 ↓
	6 Paars-blauw	12 ↓
	7 Blauw	14 Zeer helder
	8 Licht blauw	
	9 Blau-groen	
	10 Turquoise	
	11 Groen-blauw	
	12 Groen	
	13 Geel-groen	
	14 Oranje-groen	
	15 Licht oranje	

Figuur 8.1: De betekenis van de drie codegetallen in de SETCOLOR instructie in de vierkleuren grafische stand.

accepteert in deze stand bij het instellen van de helderheid alleen maar even getallen. Als u een oneven getal gebruikt rondt hij het naar beneden af.

Om te zien hoe dit er in de praktijk uitziet gaan we de kleuren van de verfkwasten veranderen. Typ het volgende programma:

```

10 GRAPHICS 7
20 COLOR 1
30 FOR Y=10 TO 20
40 PLOT 20,Y
50 DRAWTO 140,Y
60 NEXT Y
70 COLOR 2
80 FOR Y=30 TO 40

```

```
90 PLOT 20,Y
100 DRAWTO 140,Y
110 NEXT Y
120 COLOR 3
130 FOR Y=50 TO 60
140 PLOT 20,Y
150 DRAWTO 140,Y
160 NEXT Y
```

Als u dit programma laat lopen gaat de computer in de vierkleuren stand en tekent drie horizontale balken op het scherm. Elke balk werd met een andere verfkwast getrokken in de kleur die van te voren ingesteld was: een in oranje, een in 't groen en een in blauw. In het tekstraam verschijnt er **READY** waarmee de computer aangeeft dat hij klaar is.

Nu u met alle drie de verfkwasten iets hebt getekend kunt u de kleurinstelling veranderen. We veranderen eerst kwast 1 van oranje naar een andere kleur. Typ de volgende instructie:

```
SETCOLOR 0,5,12
```

Als u op **RETURN** drukt verandert de oranje balk in helder oplichtend rose. Het eerste nummer heeft betrekking op het te veranderen register: register 0, dat verfkwast 1 bepaalt. De andere twee getallen geven de kleur en de helderheid aan. U hebt deze registers ook verandert: kleur 5 (rose-paars) met helderheid 12.

Probeer eens de kleur of de helderheid van deze eerste balk te veranderen door **SETCOLOR 0** instructies te gebruiken. Verander ook de kleuren van de twee andere verfkwasten met **SETCOLOR 1** en **SETCOLOR 2** instructies. Sta niet verbaasd als een van deze instructies ook het tekstraam betreffen. Door de verfkwasten te veranderen verandert u ook de tekst en de achtergrondkleuren aan de onderkant van het scherm. Het is niet mogelijk het ene zonder het andere te veranderen.

U kunt de achtergrondkleur van het scherm veranderen door het **SETCOLOR 4** register te wijzigen. Probeer het eens met

```
SETCOLOR 4,5,6
```

De hele achtergrond van het scherm wordt rose. Dit maakt de kleur van de wisborstel, verfkwast 0, ook rose omdat wissen eigenlijk kleuren met de achtergrondkleur is.

Als u van de door u zelf ingestelde kleur af wilt kunt u de computer terugzetten in de standaardinstelling. Gebruik daarvoor de waarden uit de rechterkolom van figuur 8.2:

```
SETCOLOR 0,2,8
SETCOLOR 1,12,10
SETCOLOR 2,9,4
SETCOLOR 4,0,0
```

SETCOLOR REGISTER	NUMMER VAN DE VERFKWAST (KLEUR)	STANDAARD KLEUREN-INSTELLING	SETCOLOR CODE
0	COLOR 1	Oranje	0,2,8
1	COLOR 2	Groen	1,12,10
2	COLOR 3	Blauw	2,9,4
3	-	Rose of Rood	3,4,6
4	COLOR 0 (wissen)	Zwart	4,0,0

Figuur 8.2: De COLOR en SETCOLOR coderingen in de vierkleuren grafische stand.

Deze coderingen worden de *defaultwaarden* (standaardinstelling) genoemd omdat deze codes automatisch in de kleurenregisters zitten als u de computer aanzet. U kunt deze codes ook terugkrijgen door op RESET te drukken, maar daarmee maakt u ook het scherm schoon.

U kunt SETCOLOR ook gebruiken om het aanzien van het scherm in de tekststand (GRAPHICS 0) te wijzigen. De SETCOLOR registers hebben in deze stand een andere betekenis, maar het is niet moeilijker. In plaats van de verfkwasten, beïnvloeden de registers in de tekststand delen van het scherm: tekst, achtergrond en rand. Figuur 8.3 geeft een overzicht van hun betekenis.

Register 1 bestuurt alleen de helderheid van de letters, niet hun kleur. De letters hebben dezelfde kleur als de achtergrond: u kunt hem donkerder of lichter maken, echter nooit een andere kleur geven.

Om de werking hiervan te zien moet u op RESET drukken om de kleurenregisters in hun originele stand te brengen. De fabrieksinstelling van de achtergrondkleur, zie figuur 8.3, is kleur 9 met een helderheid van 4: donkerblauw. De letters van het woordje READY worden bepaald door register 1 dat op kleur 12 en helderheid 10 staat. Dit is normaal lichtgroen. De computer negeert de kleur echter en gebruikt de helderheid (10) en maakt de letters lichtblauw in plaats van groen. Als de letters minder helder zijn dan de achtergrond verschijnen ze donker tegen een lichtere achtergrond.

SETCOLOR REGISTER	DEEL VAN HET SCHERM	STANDAARD KLEUREN-INSTELLING	SETCOLOR CODE
0	–	Oranje	0,2,8
1	Tekst helderheid	Groen	1,12,10
2	Achtergrond	Blauw	2,9,4
3	–	Rose of Rood	3,4,6
4	Rand	Zwart	4,0,0

Figuur 8.3: De SETCOLOR registers in de tekststanden.

Probeer de kleur van de letters, achtergrond en rand te veranderen. Hier zijn enkele interessante combinaties:

SETCOLOR 2,0,0
 SETCOLOR 2,15,14
 SETCOLOR 2,9,4
 SETCOLOR 1,0,14
 SETCOLOR 1,0,0
 SETCOLOR 4,5,10

Om naar het blauwe scherm terug te keren kunt u altijd op RESET drukken.

Als u de letters en de achtergrond van het scherm zo instelt dat ze dezelfde helderheid hebben ziet het eruit alsof de letters verdwijnen. Ze zijn er nog wel, maar u kunt ze niet meer onderscheiden. Om ze te voorschijn te laten komen kunt u op RESET drukken of blindelings een SETCOLOR instructie geven.

Verwijderen van het tekstraam

Bij alle grafische programma's bevindt zich aan de onderkant van het scherm het blauwe tekstraam. Dit is erg belangrijk omdat u als de computer het hele scherm zou gebruiken geen plaats zou hebben om uw instructies in te typen.

U zou echter op een gegeven moment er behoefte aan kunnen hebben met een bepaalde figuur het hele scherm te benutten. Figuren over het gehele scherm zijn meestal attractiever omdat ze dan niet verstoord worden door het blauwe gedeelte. Sommige van de geavanceerde grafische standen kunnen niet gebruikt worden als het tekstraam op het scherm staat.

U kunt het tekstraam verwijderen door in de GRAPHICS opdracht 16 bij het standnummer op te tellen:

10 GRAPHICS 7+16

Omdat $7+16$ gelijk is aan 23 kunt u ook schrijven

10 GRAPHICS 23

Het is echter beter $7 + 16$ te schrijven omdat dat duidelijker aangeeft wat u doet.

Omdat de grafieken nu ook beneden de onderkant van het normale scherm kunnen komen kunt u enkele coördinaten gebruiken die groter zijn dan u tot nu toe gewend was. De grenzen van X zijn hetzelfde gebleven (0 tot 159), maar de Y-coördinaten kunnen nu tot 95 lopen (in plaats van zoals u gewend was tot 79).

Omdat u bij dit vergrootte grafische scherm geen ruimte meer hebt om instructies in te typen kunt u het alleen binnen opgestelde pro-

gramma's gebruiken. U moet in uw programma instructies geven voor alles wat u de computer wilt laten tekenen en dan steeds het programma laten uitvoeren (RUN). U kunt dus als de computer eenmaal in zo'n grafische stand staat geen directe instructies meer geven.

Als u uw programma normaal laat beëindigen zal de computer uw afbeelding niet zoals gewoonlijk op het scherm laten staan. Hij zal het gehele beeld onmiddellijk wegwissen en het woordje **READY** verschijnt. Dit kunt u voorkomen door de computer met de laatste programmaopdracht in een oneindige lus te sturen zodat hij de afbeelding tekent en daarna vastloopt. Dit kan met een opdracht als

```
999 GOTO 999
```

Als de computer bij deze laatste opdracht komt blijft de afbeelding op het scherm staan. Hij reageert ook niet op uw toetsenaanslag. Hij is immers bezig regel 999 te herhalen. Als u klaar bent om weer instructies te geven drukt u om het programma te stoppen op de **BREAK**-toets. Hierdoor wordt de afbeelding van het scherm gewist en u kunt weer instructies geven.

Probeer het volgende programma om de werking van grafieken op een geheel scherm te zien:

```
10  GRAPHICS 7+16
20  COLOR 1
30  PLOT 0,0
40  DRAWTO 159,95
50  PLOT 0,95
60  DRAWTO 159,0
999 GOTO 999
```

Als u het programma laat lopen tekent de computer een oranje X over het gehele scherm. Regel 999 plaatst het programma in een oneindige lus zodat de afbeelding op het scherm blijft. Druk op **BREAK** zodat u de besturing over het scherm terugkrijgt.

De grafische standen

Tot nu toe hebt u slechts twee van de 16 grafische standen van uw computer gezien. Deze twee zijn ook de belangrijkste, maar de overige veertien zijn ook handig. Door de stand te gebruiken die voor uw doeleinden het beste is kunt u uw grafische figuren volgens uw wensen samenstellen.

Figuur 8.4 is een vergelijkingstabel van de verschillende standen. Bekijk de verschillen tussen de standen goed.

De kolom waarin de rasterdichtheid (resolution) wordt gegeven is erg belangrijk. Deze getallen geven aan uit hoeveel kolommen en regels het beeld is opgebouwd. Hoe fijner het raster des te dichter opeen liggen de getekende punten en lijnen. Als u erg gedetailleerde afbeeldingen wilt tekenen moet u het fijnste mogelijke raster gebruiken. Als u afbeeldingen met dikke lijnen wilt tekenen kunt u standen met een grover raster gebruiken.

Het gekozen raster heeft invloed op de hoeveelheid geheugen dat de computer nodig heeft. Bij een fijn raster heeft de computer heel wat ruimte in het geheugen nodig om de beeldinformatie in op te slaan. Dit is erg belangrijk wanneer u een Atari 600 XL hebt zonder geheugenuitbreiding, omdat sommige standen bijna de volledig beschikbare geheugenhoeveelheid in beslag nemen. Dit kan problemen opleveren. U kunt in die stand n.l. geen programma's schrijven die uit meer dan tien tot vijftien regels bestaan. U kunt alle grafische standen op uw computer gebruiken. U moet echter voorzichtig zijn met de standen die meer dan 8000 bytes geheugen nodig hebben (zie rechterkolom figuur 8.4). Het geheugen is geen probleem indien u een geheugenuitbreiding of een van de duurdere Atari computers hebt, b.v. model 800XL.

Als u een model uit de nieuwe XL-serie hebt kunt u over alle zestien standen beschikken. Als u een van de oudere modellen 400 of 800 hebt, mist u enkele standen tussen 9 en 15. Alle oudere modellen hebben de standen tussen 0 en 8 en enkele ook 9, 10 en 11. Maar maak u geen zorgen: zelfs als u enkele speciale standen mist, het meest wezenlijke zit er altijd in. Sla gewoon de beschrijving en uitleg over van de standen die u niet hebt.

STAND NUMMER	SOORT STAND	RASTER- DICHT- HEID	AANTAL KLEUREN	GE- BRUIKTE GE- HEUGEN (BYTES)
0	Normale tekst	40 x 24	1	992
1	Grote tekst	20 x 20	5	674
2	Grote tekst	20 x 10	5	424
3	4-kleuren grafieken	40 x 20	4	434
4	2-kleuren grafieken	80 x 40	2	694
5	4-kleuren grafieken	80 x 40	4	1174
6	2-kleuren grafieken	160 x 80	2	2174
7	4-kleuren grafieken	160 x 80	4	4190
8	fijn raster grafieken	320 x 160	1	8112
9	GTIA grafieken	80 x 192	1	8138
10	GTIA grafieken	80 x 192	9	8138
11	GTIA grafieken	80 x 192	16	8138
12	4-kleuren tekst	40 x 20	5	1154
13	4-kleuren tekst	40 x 10	5	664
14	2-kleuren grafieken	160 x 160	2	4270
15	4-kleuren grafieken	160 x 160	4	8112

Figuur 8.4: De 16 grafische standen

Na deze opmerkingen kunnen we doorgaan met onze ontdekkingsreis van de grafische mogelijkheden. In deze enkele bladzijden kan ik jammer genoeg slechts een kort overzicht geven. Als u er meer over wilt weten kan ik u aanraden het boek van Bill Carris *Inside ATARI BASIC* (Reston, 1983) of *Atari Sound and Graphics* van Moore, Lower en Albrecht (Wiley, 1982) te lezen. Die zijn uitgebreider dan dit boek.

Vierkleuren grafische standen. U hebt er al een gebruikt: grafische stand 7. De andere drie werken op precies dezelfde manier, maar hebben een ander raster. Stand 3 en 5 hebben een grof raster, dus daar liggen regels en kolommen verder uiteen en hun coördinaten zijn kleiner.

Stand 15 heeft zelfs een fijner raster dan stand 7, met vierkanten die half zo hoog zijn. Jammer genoeg neemt hij tweemaal zoveel geheugen in beslag - bijna de gehele beschikbare ruimte op Atari 600XL. Als u model 800XL of een geheugenuitbreiding hebt zult u stand 15 erg handig vinden.

De verfkwasten en de SETCOLOR registers werken allemaal precies hetzelfde in de vier standen. U kunt een programma van de ene stand naar de andere veranderen door slechts de coördinaten van de gewenste schaal aan te geven.

Tweekleuren grafische standen. De standen 4,6 en 14 zijn gelijk aan de vierkleuren standen 5, 7 en 15 behalve dat zij twee van de vier verfkwasten missen. U kunt nog steeds verfkwast nummer 1, met oranje als standaardkleur en wisborstel (COLOR 0) gebruiken; 2 en 3 kunt u echter niet gebruiken. Wat is het nut van tweekleuren standen als zij niets anders doen? Zij beslaan slechts half zoveel ruimte in het computergeheugen. Dit kan erg belangrijk zijn als u op een Atari 600XL lange programma's schrijft. Als u slechts met één kleur wilt werken bespaart u bij gebruik van deze standen ruimte.

Fijn raster grafieken. Uw Atari computer heeft ook een speciale fijn raster grafische stand. Dit is nummer 8 en er zijn nogal wat verschillen in vergelijking met de andere.

GRAPHICS 8 geeft u tweemaal de fijnheid van zelfs de fijnste twee en vierkleuren standen. De punten die er neergezet worden zijn de kleinste die mogelijk zijn op een gewoon televisiescherm. Dit betekent dat de computer diagonalen kan tekenen die nagenoeg recht zijn. Dus zonder het „trapeffect” dat u vaak in de andere standen ziet. U kunt deze fijn rasterstand gebruiken om ingewikkelde figuren met fijne details te tekenen.

Hiervoor betaalt u uw prijs. Om deze hoge nauwkeurigheid te krijgen moet u kleur opofferen. U kunt alleen lijnen trekken in dezelfde tint als de achtergrond. Over het algemeen betekent dit dat u uzelf tot wit moet beperken.

De fijn raster grafieken nemen ook een groot gedeelte van het computergeheugen in beslag. Om deze stand op een Atari 600XL te kunnen gebruiken moet u of hele kleine programma's schrijven of een geheugenuitbreiding kopen.

U hebt in de fijne rasterstand enige controle over de kleuren van het scherm, maar het systeem om de kleuren te veranderen is niet gelijk aan dat van de andere standen. Het lijkt op de standaard-tekststand. SETCOLOR 2 kiest de achtergrondkleur, SETCOLOR 1 de helderheid van de getekende punten en lijnen en SETCOLOR 4 bepaalt de rand. Zoals in de tekststand kunt u de punten alleen de kleur van de achtergrond geven. De helderheid ervan kunt u wel bepalen. De SETCOLOR instructies hebben hun uitwerking op zowel het tekstraam als op het grafische scherm.

U hebt slechts een grafische verfkwast in de fijne rasterstand, maar u kunt hem aan- en uitzetten met de COLOR instructie. COLOR 1 zorgt ervoor dat de verfkwast de lijnen trekt, terwijl COLOR 0 hem laat wissen door de achtergrondkleur te gebruiken.

Als u fijn raster grafieken wilt gebruiken, adviseer ik met de volgende instructies te beginnen:

```
GRAPHICS 8  
SETCOLOR 2,0,0  
COLOR 1
```

Deze selecteren de stand en veranderen de achtergrondkleur in zwart zodat de grafieken naar voren komen. U kunt natuurlijk voor het scherm een andere kleur kiezen, maar de combinatie van wit op zwart werkt meestal het prettigst.

Speciale grafische stand (GTIA). De XL series en sommige nieuwere Atari 400 en 800 modellen hebben drie andere speciale grafische standen die genummerd zijn met 9, 10 en 11. Ze worden soms *GTIA standen* genoemd naar de naam van de elektronische chip die Atari toevoegde om ze mogelijk te maken.

Deze drie standen zijn handig omdat ze een breed scala aan kleuren bieden. Met nummer 11 kunt u zestien verschillende kleuren tegelijk op het scherm krijgen. U kunt hiermee regenboogachtige figuren tekenen. In stand 9 kunt u zestien verschillende helderheden tegelijk afbeelden.

De GTIA standen zetten punten neer die een beetje groter zijn dan in de normale grafische standen. Hoewel dit geen ernstige beperking is wordt het moeilijk dunne diagonale lijnen realistisch te tekenen.

Deze standen zijn daarom beter voor afbeeldingen die hele gedeeltes op het scherm met kleur vullen.

De GTIA standen zijn jammer genoeg moeilijk te gebruiken omdat ze heel anders werken dan de andere. Door het grote scala aan kleuren wordt de SETCOLOR instructie gecompliceerder en geeft aan de verfkwastrnummers in de COLOR opdracht een vreemde betekenis. Daar deze standen geen tekstraam aan de onderkant van het scherm hebben kunt u ze alleen binnen programma's gebruiken.

Stand 11 is waarschijnlijk de indrukwekkendste van de drie. Alle punten moeten van dezelfde helderheid zijn, maar u kunt zestien kleuren tegelijkertijd gebruiken. U bepaalt de helderheid door in het SETCOLOR register 4 een waarde te zetten. Daarna kunt u met de COLOR opdracht zestien verschillende verfkwasten kiezen. De verfkwasten hebben de nummers 0 tot en met 15 en komen overeen met de kleurnummers in de middelste kolom uit figuur 8.1.

Stand nummer 9 is een eenkleur versie met hetzelfde idee. In deze stand hebben alle punten dezelfde kleur, maar u kunt zestien helderheidswaarden gebruiken. U kiest een kleur door een waarde in SETCOLOR register 4 te stoppen of u gebruikt de standaardkleur, grijs. Het nummer in de COLOR instructie bepaalt de helderheid van deze stand, dus niet de kleur. Deze kleur is vooral interessant omdat u zestien helderheidswaarden kunt gebruiken en niet alleen de even nummers.

Stand nummer 10 is waarschijnlijk de flexibelste, maar het is ook de meest gecompliceerde. Met deze stand kunt u over negen verschillende verfkwasten tegelijk beschikken en u kunt de kleur en helderheid kiezen op uw eigen manier. Om hiervan effectief gebruik te maken hebt u gevorderde programmeertechnieken nodig die niet in het concept van dit boek passen. Als u hierin geïnteresseerd bent kunt u het boek *Inside ATARI BASIC* van Bill Carris lezen.

Om een indruk van de mogelijkheden van deze speciale standen te krijgen zou u het volgende programma in moeten typen:

```
10 PRINT "WELKE STAND";:INPUT STAND
20 GRAPHICS STAND
30 FOR Y=0 TO 79
```



```
40 COLOR Y
50 PLOT 0,Y
60 DRAWTO Y,Y
70 NEXT Y
80 GOTO 80
```

Dit is een variatie op het programma grote driehoek aan het einde van hoofdstuk 7. Het verschil hierbij is echter dat de computer iedere horizontale lijn met een andere kleur tekent.

Als u het programma laat lopen vraagt de computer:

WELKE STAND?

U kunt elke grafische stand kiezen alhoewel enkele van de grove rasterstanden foutmeldingen veroorzaken die u kunt negeren. Als u klaar bent drukt u om de computer te stoppen op **BREAK**. Daarna kunt u het programma met een ander nummer opnieuw laten lopen. Stand 11 geeft een mooi 16-kleurig regenboogpatroon. Stand 9 geeft vijf grijze balken die door alle grijstonen van zwart naar wit gaan (dit is het patroon uit figuur 8.5). Stand 10 geeft een serie van kleurrijke banden ondanks het feit dat dit programma niet alle negen kleuren gebruikt. Ook de standen 6, 7, 8, 14 en 15 werken in dit programma.

De tekststanden

In aanvulling op de 11 grafische standen heeft uw computer ook nog vijf *tekststanden*. Een kent u er al, de standaard wit-op-blauw letters die u steeds hebt gebruikt. Dit is **GRAPHICS** stand 0 en verreweg het eenvoudigste in het gebruik. De andere hebben enkele handige eigenschappen zoals bijvoorbeeld zeggingschap over de kleur van de letters.

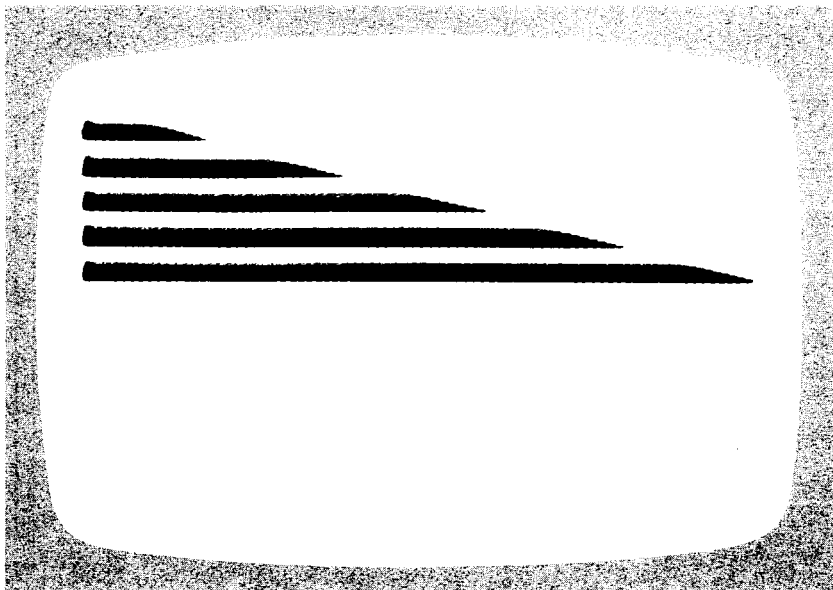
De andere vier tekststanden lijken in enige opzichten op de grafische standen hoewel u in deze standen geen punten en lijnen kunt tekenen. Als u ze met een **GRAPHICS** opdracht uitkiest geeft de computer een zwart grafisch scherm. U kunt tegen deze achtergrond letters in verschillende kleuren neerzetten.

U gebruikt een variatie van de standaard PRINT instructie om met deze speciale standen tekst af te drukken. Typ meteen achter het woordje PRINT #6 en puntkomma (;). Daarna, tussen aanhalingstekens, de boodschap. Een typische opdracht in een van deze standen zou er zo uit kunnen zien:

PRINT #6;"HALLO"

#6 is voor de computer een teken dat u de tekst op het grafisch scherm wilt hebben. Anders zou hij met normale witte letters in het blauwe tekstraam afgedrukt worden.

Grote tekst. Met stand 1 en 2 kunt u grotere letters neerzetten dan met de normale tekststand. U kunt er alleen maar hoofdletters mee schrijven, maar de kleur kunt u kiezen. Deze standen zijn daarom handig voor grote, belangrijke boodschappen en als u een tekst uit meer kleuren af wilt drukken. Omdat de letters breder zijn dan de normale passen er maar 20 tekens op een regel. Tussen de beide standen is slechts een klein verschil; de letters in stand 2 zijn tweemaal zo hoog als in stand 1. Voor de rest zijn ze hetzelfde.

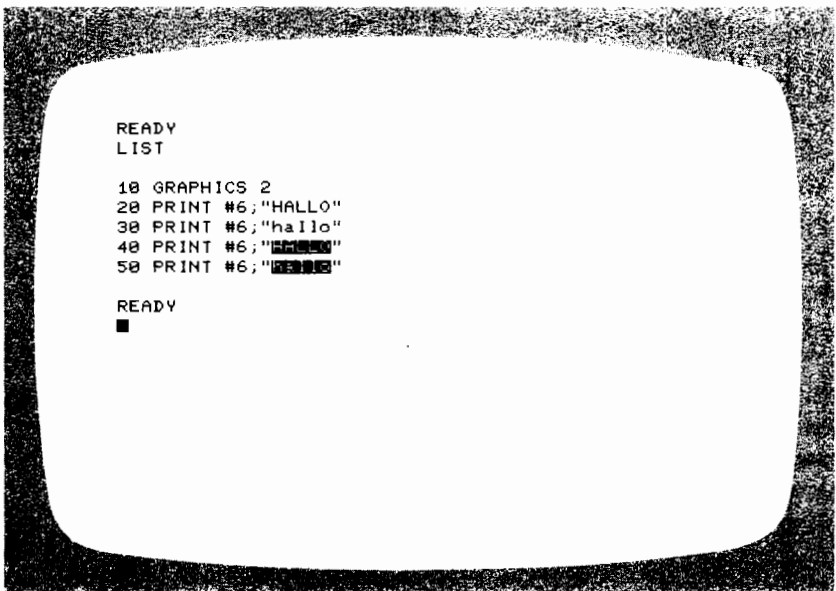


Figuur 8.5: Met de speciale stand 9 hebt u een volledige controle over de helderheden.

De functie van de CAPS en inverse-video toetsen wijkt in deze standen af. Als u met PRINT kleine letters of inverse-videotekst afdruckt krijgt u een enigzins verbazingwekkend resultaat. Het worden hoofdletters, maar in andere kleuren. Om te zien hoe dat functioneert moet u het programma uit figuur 8.6 eens laten lopen en er goed op letten dat u de kleine letters en inverse-videotekst typt zoals ze op het voorbeeld staan.

Figuur 8.7 laat u zien hoe u de SETCOLOR registers kunt gebruiken om de kleuren van elk van de vier soorten tekst die u in stand 1 en 2 kunt typen, kunt veranderen. Let erop dat deze kleurinstelling ook de kleur van het tekstream aan de onderkant van het scherm bepaalt.

Vierkleuren tekst. Stand 12 en 13 hebben meer speciale functies en zijn moeilijker te gebruiken. Zij bieden echter interessante keuzemogelijkheden. Bij deze standen die alleen op de nieuwe XL series aangebracht zijn is elke letter opgebouwd uit punten van vier verschillende kleuren. Als u deze kleuren goed kiest kunt u interessante



Figuur 8.6: Een programma waarmee u blokletters in vier kleuren af kunt beelden.

drie-dimensionale effecten krijgen. Ook deze standen zijn, met uitzondering van de hoogte der letters hetzelfde.

Om deze stand te proberen kunt u de eerste regel uit het programma op figuur 8.6 veranderen in

10 GRAPHICS 13

Laat dan het programma lopen. In het begin zult u waarschijnlijk het woordje HALLO niet herkennen. De computer drukt het vier keer aan de bovenkant van het scherm af. Dit komt omdat het binnenste van de letters met de standaardkleuren gevuld wordt. Om het scherm leesbaarder te maken moet u de volgende instructies in het tekststream typen:

```
SETCOLOR 0,2,2
SETCOLOR 2,9,12
SETCOLOR 3,4,10
```

U zult nu de tekst kunnen lezen en tegelijkertijd zien dat elk deeltje van de letters uit aparte kleuren bestaat.

Deze stand kan zowel hoofdletters als kleine letters afdrucken. Inverse-video bestuurt echter het kleurenschema zoals u uit de derde en vierde afdruk van het woordje HALLO op het scherm kunt zien. De blauwe delen zijn in rood verandert, terwijl de andere kleuren hetzelfde gebleven zijn. Figuur 8.8 toont de betekenis van de SETCOLOR registers in deze twee standen.

SETCOLOR REGISTER	GEKOZEN LETTERTYPE	STANDAARD KLEUREN-INSTELLING	SETCOLOR CODE
0	Normale hoofdletters	Oranje	0,2,8
1	Normale kleine letters	Groen	1,12,10
2	Inverse hoofdletters	Blauw	2,9,4
3	Inverse kleine letters	Rose of Rood	3,4,6
4	Achtergrond	Zwart	4,0,0

Figuur 8.7: De besturing van de kleuren in de tekststanden 1 en 2.

SETCOLOR REGISTER	DEEL VAN HET GEKOZEN LETTERTYPE	STANDAARD KLEUREN-INSTELLING	SETCOLOR CODE
0	Linkerkant	Oranje	0,2,8
1	Midden	Groen	1,12,10
2	Rechterkant van normale letters	Blauw	2,9,4
3	Rechterkant van inverse letters	Rose of Rood	3,4,6
4	Achtergrond	Zwart	4,0,0

Figuur 8.8: De SETCOLOR registers in de vierkleuren tekststanden 12 en 13.

Met de speciale tekststanden zijn we aan het einde gekomen van onze uitleg van de 16 Atari grafische standen. Het was slechts een korte kennismaking met een van de aantrekkelijkste kanten van uw computer. Ik hoop dat u met uw studie verder gaat.

Samenvatting

Uw Atari computer heeft een attractief en veelzijdig grafisch systeem. Bij gebruik van de geavanceerde technieken die in dit hoofdstuk beschreven werden beschikt u over de grondbeginselen van dit systeem.

In de grafische stand 7 kunt u met SETCOLOR de standaardkleuren van uw verfkwasten veranderen. U kunt elke combinatie kiezen van 16 kleuren en acht intensiteitswaarden voor de punten en lijnen die u op het scherm wilt tekenen. In de tekststanden regelt de SETCOLOR instructie de kleuren van de letters, achtergrond en rand.

Door het getal 16 bij het nummer van de grafische stand op te tellen kunt u het tekstraam verwijderen en het grafische scherm gaat dan tot de onderkant van het scherm. U kunt dit alleen binnen een programma doen.

Van de 16 grafische standen van Atari krijgt u er met 11 een echt grafisch scherm waarop u punten en lijnen kunt tekenen. De 11 standen verschillen onderling in het aantal kleuren en in de nauwkeurigheid

door fijn of grof raster. Als u de speciale GTIA standen beheerst kunt u verschillende kleuren op het scherm tegelijkertijd gebruiken waardoor u mooie regenboogeffecten kunt krijgen.

De vijf andere standen zijn niet voor grafieken, maar voor tekstweergave. Door de grote tekst en de vierkleuren stand te gebruiken kunt u teksten in overgrote letters en in kleur afbeelden.

Wij zijn nu aan het einde van deel twee gekomen. U weet nu hoe u programma's moet schrijven en variabelen kunt gebruiken. Tevens kunt u de volgorde waarin de computer uw instructies opvolgt regelen. U hebt een klein deel van de mogelijkheden gezien van het grafische systeem. U bent nu zover dat u zelfstandig met uw computer verder op ontdekking kunt gaan. Nu wordt het pas interessant.

Deel 3

Het opslaan

Hoofdstuk 9

Het opslaan op cassette

In de twee eerste delen van dit boek hebben we het over de Atari computer zelf gehad. Alles wat u tot nu toe heeft gedaan lukt met computer en televisie; een extra apparaat werd niet vereist.

In het derde gedeelte behandelen we enkele dingen die u met aanvullende apparatuur die u aan uw computer kunt aansluiten kunt doen. In dit hoofdstuk staat de Atari Programma Recorder centraal. Hij gebruikt een cassettebandje om gegevens en programma's op te slaan. Hoofdstuk 10 gaat over de disk drive die een soortgelijke functie heeft. Deze is sneller en handiger, maar iets duurder dan een programma recorder. In hoofdstuk 11 zullen we het nog over een paar andere apparaten hebben die u kunt toevoegen zodat het systeem nog bruikbaar wordt.

Er zijn twee redenen om een programma recorder of een disk drive te gebruiken. Allereerst worden veel interessante programma's voor uw Atari op cassettebandje of diskette aangeboden. Als u hiervan gebruik wilt maken hebt u een van de twee nodig om het programma in het geheugen van uw computer te laden.

De tweede reden is de permanente opslag. Als u in uw computer een programma typt zit het erin totdat u een NEW instructie geeft of de stroom uitzet. Als u het programma slechts een paar minuten wilt gebruiken is dat niet erg. Veel mensen schrijven echter hun programma's met de bedoeling ze later ook nog eens te gebruiken. Met een cassetterecorder of een disk drive kunt u het programma opslaan, de computer uitzetten en het programma daarna opnieuw laden. Op deze manier is het mogelijk dat u een gecompliceerd programma te allen tijde ter beschikking houdt.

In dit hoofdstuk behandelen we hoe u de Atari Programma Recorder aan moet sluiten en gebruiken. U zult erachter komen hoe u commerciële programmatuur vanaf een cassettebandje in uw computer kunt laden en hoe u uw eigen programma's kunt opslaan.

Het klaarzetten van de programma recorder

Er zijn twee versies van de Atari Programma Recorder. Het nieuwste model is model 1010. Het is compact, werkt goed en past qua stijl bij de 600XL en 800XL computers.

Als uw systeem al iets ouder is kan het zijn dat u model 410 hebt, een oudere versie. Het werkt precies als model 1010 zodat u beide modellen kunt gebruiken.

Om de programma recorder aan te sluiten moet u twee verbindingen maken. Eerst de dikke zwarte kabel uit de achterkant van de recorder halen en in de opening aan de achterkant waarbij PERIPHERAL staat steken. De stekker past er maar op een manier in. Probeer zolang totdat hij past. Vervolgens moet u de stekker van de recorder in het stopcontact steken.

De programma recorder is hetzelfde als een gewone cassetterecorder. Het enige verschil is dat u hem d.m.v. de speciale aansluiting via de computer automatisch aan en uit kunt zetten. De RECORD, PLAY, REWIND en ADVANCE knoppen werken hetzelfde als op een normaal cassettedeck; OPNAME, AFSPELEN, TERUGSPOELEN en VOORUIT. U kunt normale cassettebandjes gebruiken. Een normale cassetterecorder kunt u echter niet nemen. U hebt de Atari recorder met de speciale kabel nodig.

Als u alleen kant en klare programma's gebruikt zult u waarschijnlijk geen lege bandjes kopen. Normalerwijze hoeft u alleen maar het programma van het gekochte bandje in het geheugen van uw computer te laden.

Als u uw eigen programma's wilt opslaan of een speciaal programma wilt gebruiken dat ervan uitgaat dat u informatie op band wilt zetten moet u een paar lege bandjes kopen. Gebruik normale cassettes. Korte bandjes zoals b.v. C-30 zijn meestal beter omdat langere bandjes uitrekken en daardoor onleesbaar kunnen worden. Het is niet no-

dig high-fidelity of speciale computerbandjes te kopen. De goedkoopsten zijn net zo goed.

Wees zuinig op uw bandjes! Omdat de bandjes een duur programma of kostbare gegevens kunnen bevatten moet u er voorzichtig mee omgaan. Iedere keer als u het programma wilt gebruiken moet u het van het bandje lezen en zelfs een heel klein barstje of scheurtje kan het onleesbaar maken.

Spoel de bandjes altijd terug naar het begin voordat u ze uit de recorder neemt. De meeste bandjes hebben aan het begin een plastic *aanloopstrookje*. Als u niet tot daartoe terugspoelt kan het zijn dat u het bandje bij het uitnemen uit de cassette beschadigt.

Houd de bandjes uit de buurt van magnetische velden. Deze zouden de opgenomen informatie kunnen wissen. Zelfs uw televisie of telefoon kunnen een magnetisch veld veroorzaken dat sterk genoeg is om uw opname te beschadigen. Houd uw programma recorder ten minste 60 cm van uw televisie verwijderd en laat nooit bandjes op uw bureau rondslingeren.

Het laden van kant en klare cassette programma's

Een groot gedeelte van de commerciële beschikbare software voor de Atari computers staat ter beschikking op cassettebandjes. Cassettes zijn eenvoudig in gebruik en overal te krijgen. Als u zo'n programma koopt krijgt u een gewone cassette waarop het programma staat. Om met het programma te kunnen werken moet u het in het geheugen van uw computer laden.

Omdat veel firma's kant en klare cassettes op de markt brengen is het onmogelijk algemene laadinstructies te geven. U moet de bijgeleverde instructies lezen om erachter te komen welke specifieke instructies voor het laden van dat speciale programma nodig zijn.

Hoewel ik geen procedure kan geven die voor alle cassettes geldt kan ik wel de meest gebruikelijke vertellen. Die is

1. Typ:

CLOAD

2. Druk op de RETURN toets en u hoort een piep.

3. Leg het bandje in de recorder en spoel hem terug totaan het begin van de kant waar het programma is opgeslagen.
4. Druk op PLAY van de recorder. Het bandje start nog niet.
5. Druk nog een keer op RETURN. Het bandje begint te lopen en u hoort door uw televisie een piepend geluid.
6. Wacht totdat het bandje stopt. Als de computer READY meldt is het programma correct geladen.
7. Druk op STOP van uw programma recorder en spoel het bandje terug. Het programma zit nu in het computergeheugen en u kunt ermee werken zoals met ieder ander programma.

Zolang het bandje wordt geladen kunt u via het toetsenbord geen andere instructies geven. Om de procedure te stoppen en het programma niet verder te laden moet u op BREAK drukken. De computer meldt READY en u kunt weer instructies geven.

Als er een foutmelding volgt zijn er bij het laden van het programma problemen opgetreden. Spoel het bandje terug en herhaal de CLOAD procedure. Als u weer een foutmelding krijgt kan het zijn dat u een bandje hebt dat op een andere manier moet worden geladen. Controleer de meegeleverde instructies.

Het kan ook zijn dat het bandje defect is of in de loop van tijd onleesbaar is geworden. Cassettebandjes raken erg gauw beschadigd. Als u een bandje niet kunt laden dat op een eerder tijdstip wel geladen kon worden betekent dit waarschijnlijk dat de codering onleesbaar is geworden.

U kunt nu drie dingen doen. Allereerst het bandje omdraaien naar de achterkant, terugspoelen en de procedure herhalen. De meeste software leveranciers zetten op de achterkant van het bandje een copie van het programma voor het geval dat u met de eerste copie problemen heeft.

Als dat niet lukt kunt u proberen het programma ondanks de foutmelding toch te laten lopen. Dit werkt zelden, maar het is het proberen waard.

Tenslotte kunt u nog met uw leverancier contact opnemen. Omdat er regelmatig problemen zijn met cassettes zullen de meeste producenten een onleesbare cassette vervangen.

Het opslaan van uw eigen programma's

Als u een van uw eigen programma's op cassette wilt opslaan kunt u een procedure die CSAVE wordt genoemd gebruiken. Deze lijkt erg op de CLOAD instructie. Typ uw programma in het computergeheugen. Over de lengte hoeft u zich geen zorgen te maken. Er gaan honderden programma's op een kleine cassette. Als uw programma klaar is zoals u het wilt hebben typ dan

CSAVE

De computer piept twee keer. Dit betekent dat u de cassette in de recorder moet leggen, hem terug moet spoelen en op de RECORD en PLAY knoppen moet drukken. Zoals bij een normale cassetterecorder moet u allebei de knoppen tegelijkertijd indrukken. Vervolgens doet u precies hetzelfde als onder punt 3 tot en met 7 van de CLOAD instructie beschreven. U drukt op RETURN om het bandje te starten, wacht dan tot de computer stopt en READY meldt. Het programma is nu op het bandje opgeslagen.

Als het een belangrijk programma is of als het uren kost om het opnieuw in te typen is het bandje erg waardevol. Spoel het terug en kenmerk het. Doe het in een opbergcassette als u er een hebt of berg het op een plaats op waar het niet per ongeluk gewist kan worden.

U kunt uw cassettes ook tegen uitwisongelukjes beschermen door een van de kleine plastic nopjes aan de bovenkant van de cassette te verwijderen. Zoals bij geluidscassettes zal de Atari programma recorder niet op bandjes opnemen die deze nopjes niet meer hebben.

Om uw cassette op deze manier te beschermen moet u de opnamekant naar u toe houden. Dan met een pen het nopje zoals u op figuur 9.1 kunt zien, uit de linkerbovenkant drukken. Het andere nopje beschermt de achterkant van het bandje.

Als u later op de beschermde kant toch nog iets op wilt nemen is dit mogelijk. Dek het gaatje met een stukje plakband af en u kunt uw programma op de bekende wijze opnemen. Voor de recorder is het nu zo alsof het plastic nopje er nog inzit. U moet er voordat u op RECORD drukt zeker van zijn dat u op deze cassette op wilt nemen.

Als u er zeker van wilt zijn dat u geen programma kwijt raakt kunt u op een ander bandje een reserve-opname (backup) bewaren. Stop een andere lege cassette in de recorder en geef nogmaals de instructie CSAVE. Wanneer de computer met het opslaan van het programma klaar is hebt u een copie op de reservecassette. Bewaar deze reserve-opname op een andere plaats. Op deze manier kunt u als er iets met het origineel gebeurt nog altijd het programma vanaf de tweede cassette laden.

U kunt een opgeslagen programma terugladen in de computer door middel van een CLOAD instructie, precies hetzelfde als bij een kant en klaar cassetteprogramma. Volg de CLOAD procedure zoals hierboven beschreven en zet de programmalijs (LIST) op het scherm om er zeker van te zijn dat het het juiste programma is.

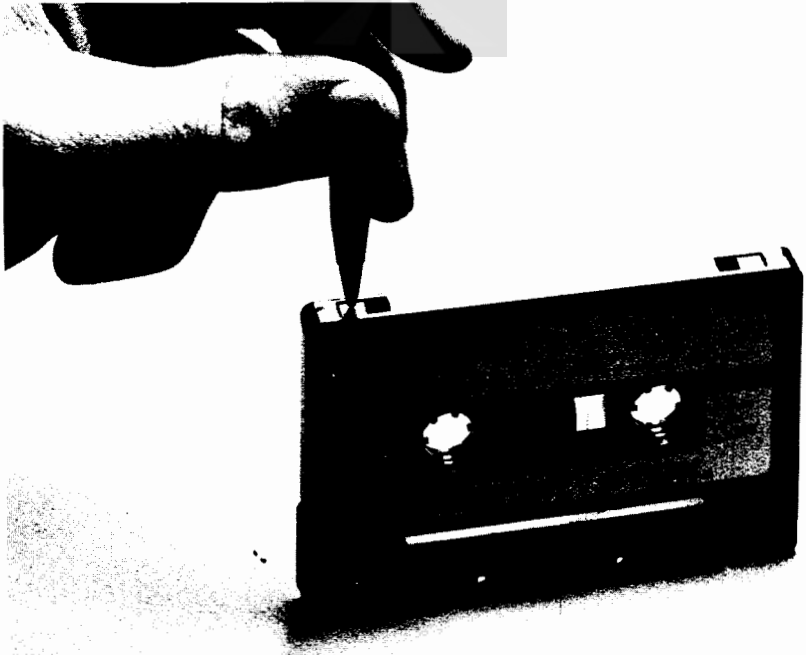
Als u de CLOAD instructie geeft wist de computer het programma dat zich in het geheugen bevindt zodat eventuele overgebleven regels niet tussen het nieuwe programma komen. Het oude programma wordt gewist zoals dat met de NEW instructie zou gebeuren. Als u dus aan een belangrijk programma hebt gewerkt moet u dit eerst opslaan voordat u een ander laadt.

Omdat er tussen de CLOAD en CSAVE procedure bijna geen verschil is kunt u ze gemakkelijk verwisselen. Als u per ongeluk op de RECORD knop drukt wanneer u een programma in het geheugen wilt laden kunt u een waardevol programma wissen.

Het aantal piepjes is een ezelsbruggetje. Als u alleen PLAY moet indrukken piept de computer één keer. Als u op PLAY en RECORD moet drukken piept hij twee keer. Dus: één piep, één knop; twee piepen, twee knoppen.

Het is mogelijk meer dan één programma op iedere kant van de cassette op te slaan. Maar wees voorzichtig. U kunt de lengte van de verschillende programma's in de gaten houden met behulp van de teller die zich aan de bovenkant van de recorder bevindt. Als u zeker weet dat u niet over een ander programma heen opneemt kunt u proberen meer programma's op een kant onder te brengen.

Als u op een bandje meerdere programma's op wilt nemen gebruik dan het resetknopje dat zich naast de teller op de programma recorder bevindt. Zet hem op 000. Spoel het bandje een stukje voorbij het



Figuur 9.1: Het beschermen van bandjes tegen het per ongeluk wissen.

eerste programma, lees de tellerstand af en neem dan het tweede programma op. Als u het tweede programma in uw computer wilt laden kunt u die plaats met behulp van de teller terugvinden.

Gebruik deze techniek slechts wanneer u erg veel programma's opslaat en het niet erg vindt als u er eens eentje verliest. U kunt bandjes sparen door programma's samen op te nemen, maar u kunt er ook gemakkelijk een wissen als u niet voorzichtig bent. Daarom moet u, behalve als u echt weet wat u doet, slechts één programma per kant maken.

Andere manieren om programma's op te slaan

De CSAVE en CLOAD instructies zijn het eenvoudigst om programma's van en naar de cassette op te slaan en te laden. Er zijn echter nog twee andere instructies die ongeveer hetzelfde werken, maar

een iets ander resultaat geven. Deze andere instructies zijn ook iets algemener omdat zij ook bij programma's werken die op andere apparatuur, b.v. disk drives, worden opgeslagen.

Bij alle instructies gebruikt u de zeven-stappen procedure om de programma recorder te besturen. U typt eerst de instructie en drukt dan op RETURN. De computer piept één of twee keer, dat hangt ervan af of u alleen op PLAY of op PLAY en RECORD moet drukken. Dan legt u het bandje in de recorder en drukt op RETURN om het te starten. Zodra de computer klaar is met het opslaan van het programma of laden van het bandje meldt hij READY.

De eerste instructies zijn SAVE en LOAD. Ze doen precies hetzelfde als CSAVE en CLOAD. Ze zijn alleen algemener en kunnen ook met andere apparaten werken. De instructies zien er zo uit:

SAVE "C:"

en

LOAD "C:"

„C:” zegt de computer dat hij de cassetterecorder moet gebruiken om het programma op te slaan of te laden. De letter C die altijd gevolgd moet worden door een dubbele punt, is de standaardnaam voor cassetterecorder. U moet hem bij alle cassette-instructies aangeven, uitgezonderd bij CSAVE en CLOAD.

De SAVE en CSAVE instructies gebruiken verschillende coderingswijzen om het programma op te slaan. U kunt niet LOAD gebruiken voor een programma dat met CSAVE opgeslagen is en omgekeerd. Evenals de CLOAD instructie wist de LOAD instructie een eventueel programma dat in het computergeheugen zit.

Waarom deze uitgebreide instructies? Omdat u door middel van SAVE een programma in één stap kunt laden en laten uitvoeren:

RUN "C:"

Dit draagt de computer op het cassetteprogramma in het geheugen te laden en direct daarna met de uitvoering te beginnen. Bij een bandje dat met CSAVE is opgeslagen zou u twee instructies moeten geven:

CLOAD
RUN

LIST en ENTER zijn de andere twee instructies. De LIST instructie is een variatie op de instructie die u al vanaf hoofdstuk 5 gebruikt om het hele of een gedeelte van het programma op het scherm weer te geven. Om de programmaregels met deze instructie op cassette op te slaan moet u typen:

LIST "C:"

„C:” draagt de computer op de programmaregels in plaats van op het scherm te laten zien op cassette op te slaan. U kunt ze weer inladen door te typen:

ENTER "C:"

U kunt nu ook alleen met ENTER programma's die met LIST zijn opgeslagen opnieuw laden. Programma's die met CSAVE of SAVE „C:” zijn opgeslagen moeten met hun eigen instructies worden geladen.

Het grote voordeel van LIST en ENTER is dat u hiermee gedeeltes van een programma apart kunt zetten. Als u alleen de programmaregels van bijvoorbeeld 40 tot en met 70 op wilt slaan kunt u typen:

LIST "C:",40,70

De computer neemt dan de regels 40 tot en met 70 op. Als u later met ENTER dit programma teruglaadt zijn dit de enige die terugkomen.

Dit kan goed van pas komen als u een handig deelprogramma uit een totaalprogramma wilt opslaan. Dit deelprogramma kunt u met LIST opslaan en later met ENTER in andere programma's terughalen.

De ENTER instructie wist niet zoals bij CLOAD en LOAD het geheugen voordat het nieuwe programma geladen wordt. Het voegt aan het programma dat al aanwezig is slechts de regels toe. Als u het geheugen leeg wilt hebben moet u NEW typen voordat u ENTER typt.

U kunt van deze instructie zelfs gebruik maken als u een groot aantal regels wilt verwijderen. Normalerweise zou u elke opdrachtnummer van de regels die u wilt verwijderen een voor een in moeten typen. Met de LIST „C:” instructie kunt u de regels die u wilt bewaren op band opslaan, met NEW het geheugen wissen en het opgeslagen gedeelte met ENTER „C:” terugladen.

Samenvatting

Met een Atari Programma Recorder kunt u met commerciële programma's die op cassettebandjes worden verkocht werken en kunt u uw eigen programma's opslaan zodat u ze niet iedere keer opnieuw hoeft in te typen.

CSAVE en CLOAD zijn de twee belangrijkste instructies om programma's op een cassettebandje op te slaan of van een cassettebandje in de computer te laden. Als u van deze instructies gebruik maakt zegt de computer dat u de cassette in de recorder moet leggen en dat u op bepaalde knoppen van de recorder moet drukken. De computer schakelt de recorder in en uit als hij de informatie moet lezen of schrijven.

SAVE „C:” en LOAD „C:” zijn alternatieve instructies. Ze zijn algemener dan CSAVE en CLOAD omdat ze ook met andere apparaten zoals bijvoorbeeld de disk drive kunnen worden gebruikt. U kunt programma's die op deze manier zijn opgeslagen laden en laten lopen met slechts een instructie en wel RUN „C:”.

U kunt ook LIST „C:” en ENTER „C:” gebruiken om programma's op te slaan. Ze werken iets anders omdat ze het programma regel voor regel opslaan. Deze instructies kunnen goed van pas komen als u slechts een gedeelte van een programma wilt bewaren.

Hoofdstuk 10

De disk drive

Met een Atari Disk Drive kunt u uw informatie gemakkelijker opslaan dan met een cassetterecorder. U kunt de kant en klare software praktisch meteen gebruiken zonder dat u hoeft te wachten totdat de programma recorder het bandje gelezen heeft. Met een disk drive kunt u veel verschillende programma's op dezelfde diskette gemakkelijk en betrouwbaar opslaan.

Een disk drive is een mechanisch apparaat dat informatie kan opslaan op en lezen vanaf een plat magnetisch oppervlak. Hij werkt ongeveer zoals een platenspeler: een ronde diskette draait heel snel en de informatie wordt in dunne sporen op het oppervlak opgenomen. In tegenstelling tot een grammofoonplaat kan de informatie die op een diskette staat gewijzigd en verwijderd worden. In dit opzicht werkt de disk drive meer zoals een bandrecorder.

In figuur 10.1 is een diskette afgebeeld. Het is een 5 1/4 inch (13 cm) vinylen schijf bedekt met een magnetisch oppervlak. Om hem te beschermen zit hij in een plastic hoes met slechts enkele openingen. Tijdens het gebruik draait de drive de disk in het hoesje en hij neemt de informatie die de computer nodig heeft van het oppervlak.

Disk betekent magnetische schijf, disk drive is het apparaat waarmee de disks (diskettes) kunnen worden gelezen of beschreven.

Omdat diskettes speciaal zijn ontworpen om computergegevens op op te slaan zijn ze in ieder geval beter dan cassettes. Ze draaien veel sneller zodat de computer meer informatie binnen een bepaalde tijd kan opslaan en terughalen. Op het platte oppervlak is de informatie efficiënter opgeslagen zodat een diskette meer kan bevatten dan een cassette.

Het grootste voordeel van het opslaan op diskette is de flexibiliteit. Met een bandje kunt u alleen aan het begin starten en tot het einde doorlezen. Bij diskettes kan de computer naar elke gewenste plaats gaan die u aangeeft om de informatie die u wilt hebben te lezen. Het opslaan van verschillende programma's en het teruglezen op een diskette is daarom eenvoudig. Door deze flexibiliteit kunt u met de disk drive dingen doen die met een cassette recorder nauwelijks mogelijk zijn.

U kunt maximaal vier disk drives op uw computer aansluiten, maar meestal is een voldoende. In dit hoofdstuk ga ik er van uit dat over het algemeen slechts één disk drive is aangesloten. Het gebruik van meerdere drives wordt aan het einde in het kort behandeld.

Het installeren van de disk drive

Atari heeft twee verschillende versies in de handel. De originele drive is model 810, rechts op figuur 10.2. Het nieuwere model, links op figuur 10.2, is een verbeterde versie die sneller werkt en op iedere diskette meer informatie kan opbergen. De verschillen zijn echter gering en beide modellen werken op alle Atari computers.

Hoewel technisch gezien de uitrusting met een disk drive beter is is hij nauwelijks moeilijker te installeren dan de programma recorder. Het principe is hetzelfde en de verschillen hebt u gauw onder de knie.

Schakel voordat u begint de computer en de disk drive uit. Nooit een disk drive aan de computer aansluiten als hij aanstaat.

Verbind de lange zwarte kabel van de disk drive met de computer. Deze kabel lijkt op die van de programma recorder en heeft aan beide kanten een zwart stekkertje. Steek het dikke einde in één van de grote I/O CONNECTOR openingen aan de achterkant van de disk drive. In welke opening maakt niet uit. Stop daarna de andere kant in de opening aan de achterkant van de computer waar PERIPHERAL staat.

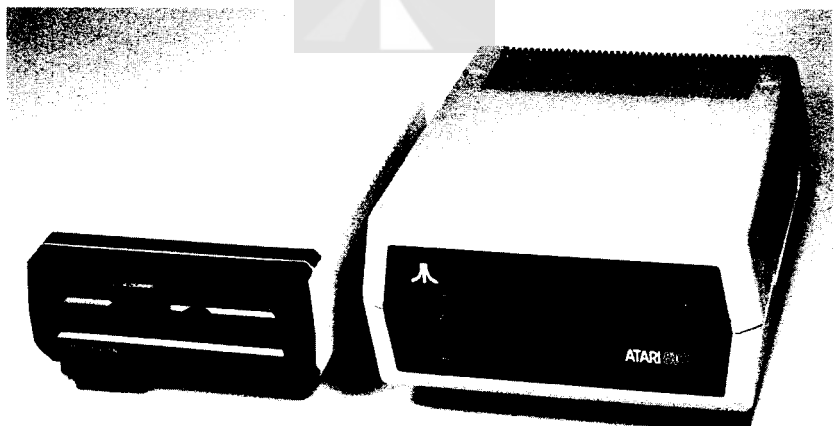
Als u de disk drive samen met een programma recorder of een ander randapparaat wilt gebruiken kunt u het kabeltje in de andere I/O CONNECTOR opening aan de achterkant van de drive steken.



Figuur 10.1: Een diskette.

Door de toestellen op deze manier door te lussen kunt u er zo veel aansluiten als u wilt. De programma recorder moet altijd aan het einde zitten omdat hij niet met twee aansluitpunten is uitgerust.

We kijken nu naar de kleine opening op de achterkant van de disk drive waarbij DRIVE SELECT of DRIVE CODE staat. Hierin zitten een paar kleine schakelaars die de computer aangeven wat voor een disk drive dit is. De tweede schakelaar ziet u misschien op het eerste moment niet, maar hij zit direct achter de eerste en er tegen-



Figuur 10.2: Het oude en het nieuwe Atari disk drive model.

aan. Als u slechts één drive gebruikt moeten allebei de schakelaars naar links staan om disk nummer 1 in te stellen (zie figuur 10.3). Voor de computer is dit disk drive D1.

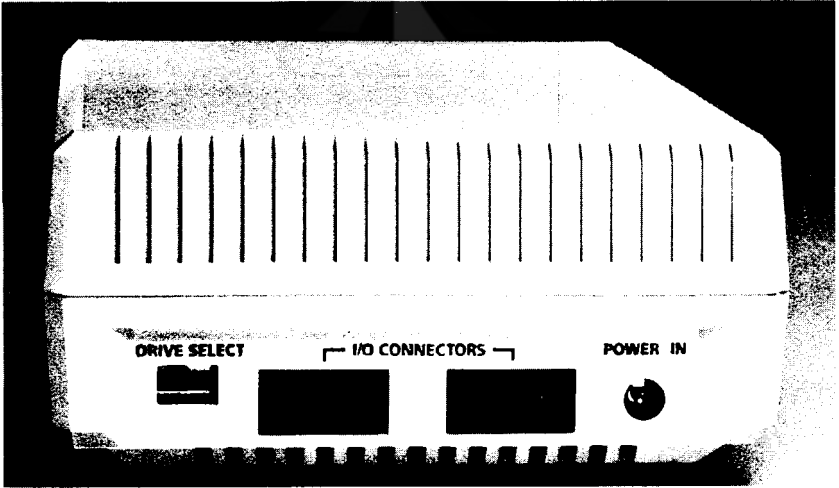
De disk drive heeft een speciale stekker. Deze wordt meegeleverd. Verbind het kleine uiteinde met de opening aan de achterkant van de drive waarbij PWR of POWER IN staat. Steek daarna de stekker in het stopcontact. Dit is alles wat het aansluiten betreft.

De omgang met diskettes

Diskettes kunnen erg gauw beschadigen. Als u er voorzichtig mee omgaat zijn ze betrouwbaar, maar als u ze niet goed behandelt beschadigt u ze voortdurend.

Buig ze nooit. U kunt ze een beetje bewegen zonder dat dat kwaad kan, maar als u ze vouwt maakt u ze onleesbaar. Pas op dat u ze niet te veel buigt als u ze in de drive steekt.

Houd vuil en stof weg van de diskettes en raak nooit het openliggende gedeelte van het opname-oppervlak aan. Door het kleinste vuiltje kan de magnetische laag op de diskette al beschadigd worden. Ga vooral voorzichtig om met de langwerpige gleuf van het midden naar buiten. Hier hebben de koppen van de disk drive contact met de diskette om de informatie op te slaan en terug te lezen.



Figuur 10.3: Behalve als u meer dan één drive gebruikt, moeten allebei de schakelaars naar links staan.

Berg de diskette na gebruik altijd in het papieren beschermhoesje op. Bewaar ze op kamertemperatuur op een plaats waar ze niet beschadigen. Er bestaan diskettenboxen waarin u de diskettes veilig op kunt bergen.

Tijdens het gebruik van een diskette in de disk drive mag u nooit een diskette plaatsen of verwijderen als de drive draait. Wacht altijd totdat de drive stopt en het rode lampje uitgaat. *Neem altijd de diskettes uit de drive voordat u hem of de computer uitzet.* Als u deze regels niet naleeft kan het gebeuren dat op de plaats waar de koppen op de diskette staan informatie uitgewist of het magnetisch oppervlak beschadigd wordt.

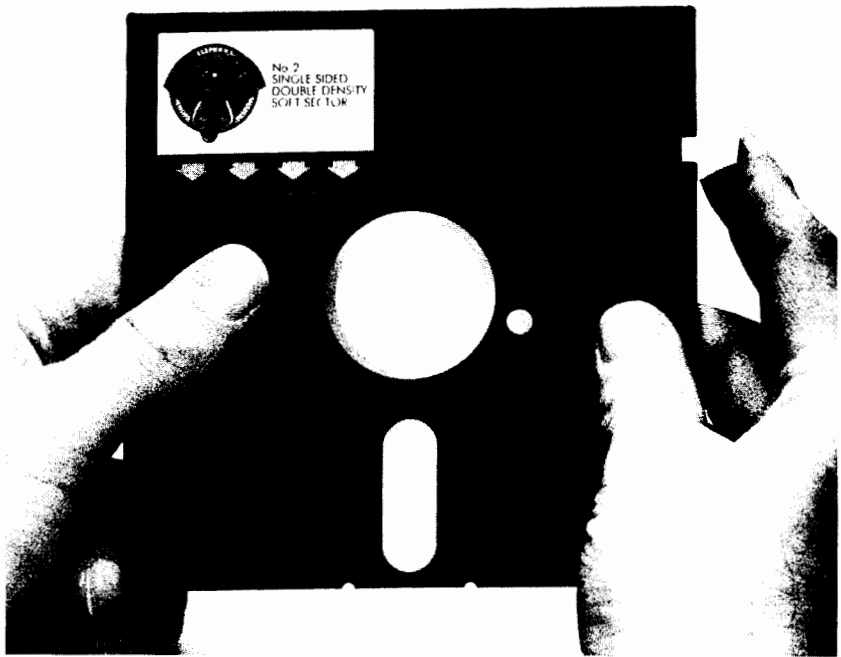
Er is een manier om het diskettesysteem te beveiligen en wel met de kleine *inkeping ter schrijfbeveiliging* aan de zijkant van het plastic hoesje. Normalerweise kunt u met de diskette lezen en schrijven. Als u nu deze inkeping bedekt zoals op figuur 10.4 beveiligt u de diskette tegen opname. De disk drive kan nog steeds informatie van de diskette lezen, maar kan er niets meer opzetten of van afwissen. Als u later toch weer op de disk wilt opnemen hoeft u alleen maar het plakertje te verwijderen. De meeste commerciële programma's worden

verkocht met de inkeping bedekt of zonder inkeping zodat u ze niet kunt wissen. Natuurlijk heeft deze afdekking geen invloed op andere beschadigingen.

Ik hoop niet dat ik u angst heb aangejaagd met zoveel doe-niet-instructies. Ik heb ervaren dat diskettes verbazendwekkend veerkrachtig zijn en vaak een buiging overleven al zou men denken dat ze stuk zouden zijn. U moet echter erg zuinig zijn op de waardevolle informatie die erop zit. Dus ben er voorzichtig mee. Als u uw diskettes goed behandelt zijn ze erg betrouwbaar.

Het aanzetten van de disk drive

Ongeacht hoe u uw disk drive wilt gaan gebruiken, u moet hem met een kant en klare diskette opstarten. Dit kan met een Atari *Disk Besturingssysteem* (*Disk Operating System: DOS*) die bij de disk drive



Figuur 10.4: Door het bedekken van de inkeping die dient om de diskette tegen ongewenste beschrijving te beschermen kunt u voorkomen dat u met de disk drive informatie van de diskette wist.

meegeleverd wordt of met een commerciële software pakket dat voor direct gebruik in de disk drive ontworpen is. Als het de eerste keer is raad ik aan de Atari DOS diskette te gebruiken.

Aan de voorkant van de drive bevindt zich een smal gleufje of deurtje waarin u de diskette moet steken. Het deurtje sluit op verschillende manieren. Dit hangt ervan af of u een oud of nieuw model hebt. Maar het principe is hetzelfde.

Bij model 1050 moet u de diskette in een erg smal gleufje schuiven. Aan de linkerkant is een hefboompje dat u naar beneden kunt drukken. Als u dat doet drukt u het drivemechanisme tegen de diskette. Wacht totdat het rode lampje uitgaat en zet het hefboompje omhoog. U kunt de disk er nu weer uit nemen.

Bij het oudere model 810 schuift u de diskette in een vrij brede gleuf. Om de diskette op z'n plaats vast te zetten moet u het zwarte plastic deurtje omlaag drukken totdat het klikt. Druk dan op het knopje iets onder het deurtje, links van de woorden Atari 810 om het deurtje te openen en de diskette eruit te nemen.

Bij allebei de modellen gebruikt u dezelfde procedure om computer en disk drive aan te zetten. Volg als u computer en disk drive gebruikt altijd de volgende instructies in de aangegeven volgorde:

1. Controleer altijd voordat u de drive aanzet of het deurtje open staat en er geen diskette inzit. U kunt een diskette beschadigen als u de disk drive aan en uitzet als hij in het gleufje zit. Uw computer moet nu nog uitstaan.
2. Schakel de lege drive in. Het rode PWR ON lampje en het „in bedrijf lampje” moeten gaan branden en de drive begint te zoemen. Schrik niet als het lijkt alsof er papier wordt kapot gescheurd. Dit geluid is normaal. Na een paar seconden gaat het „in bedrijf lampje” uit en de drive stopt. *Doe niets totdat de drive stopt.*
3. Als de drive stilstaat en het „in bedrijf lampje” uit is kunt u de gewenste diskette in het gleufje steken. Dit moet een kant en klaar softwarepakket of een Atari DOS diskette zijn. U kunt nu nog geen lege diskette gebruiken. U steekt de diskette met het etiket naar boven en de langwerpige opening naar de achterkant van de machine gericht in het gleufje van de drive. Let goed op bij het

erin schuiven. Omdat de diskette vierkant is past hij op verschillende manieren in het gleufje. Hij werkt echter alleen maar als u hem er zo inschuift als op figuur 10.5 wordt aangegeven.

4. Druk de diskette helemaal in het gleufje totdat u voelt dat hij klikt. Bij model 1050 is er rechts vanuit het midden een kleine inkeping zodat het mogelijk is de diskette ook het laatste stukje erin te duwen. Sluit het deurtje. Het drivemechanisme draait even om de diskette te centreren en stopt dan. De diskette zit nu mechanisch vast en de disk drive kan de informatie die erop staat lezen.
5. Schakel de computer en televisie in. De drive zoemt weer ongeveer tien tot twintig seconden. In de tussentijd is het scherm blauw en leeg. Als de drive stopt meldt de computer READY en u kunt uw instructies intypen.

Zo gaat het als alles goed verloopt. Het kan ook gebeuren dat uw drive blijft draaien en dat de computer meldt

BOOT ERROR

Dit wil zeggen dat u tijdens het opvolgen van de instructies een fout hebt gemaakt. Over het algemeen zal het daaraan liggen dat u een diskette gebruikt die uw computer niet kan lezen of dat u de diskette er niet juist ingezet hebt. Controleer e.e.a. aan de hand van figuur 10.5.

In dit geval blijft de drive zoemen en er verschijnt op het scherm een lange rij van BOOT ERROR meldingen. Ofschoon het rode lampje blijft branden moet u toch het deurtje van de drive opendoen om de diskette eruit te kunnen halen. Daarna de computer uitzetten en wachten totdat de drive stopt. Dan de disk drive uitzetten en opnieuw beginnen.

Waarom moet er een disk in de drive zitten als u de computer aanzet? Als u een Atari computer met een disk drive gebruikt moet de computer enkele bijkomende instructies in zijn geheugen laden zodat hij op de disk-instructies kan reageren. De Atari computer is zo geprogrammeerd dat hij deze bijkomende instructies op de diskette die u gebruikt op kan zoeken en ze dan automatisch in zijn geheugen laadt



Figuur 10.5: Het insteken van de diskette in de drive.

als hij opstart. Daarom moet u een Atari DOS of een andere kant en klare diskette gebruiken. Als de computer de instructies niet kan vinden kan hij niet goed opstarten.

Het gebruik van kant en klare disketteprogramma's

Net zoals bij kant en klare cassettes zijn er geen algemene instructies voor het laden van commerciële diskette-software. Het is het beste dat u de instructies die bij het programma zitten goed doorleest. Hierin vindt u vast en zeker gedetailleerde informatie met betrekking tot het laden van het programma.

Sommige disketteprogramma's laden en starten uit zichzelf meteen als u de computer met de diskette in de drive inschakelt. Andere star-

ten de computer op en melden **READY**. In dat geval moet u meestal de volgende instructie geven:

RUN "D:PROGRAMMA"

waarbij **PROGRAMMA** de naam van het gebruikte pakket is. Raadpleeg het instructieblad voor de details.

Als u een commerciële disketteprogramma gebruikt moet u er goed op letten dat u de boven aangegeven vijf stappen nauwkeurig opvolgt. Disketteprogramma's kunnen vrij duur zijn en door een klein foutje kunt u ze onleesbaar maken. De meeste softwareproducenten vervangen een beschadigde diskette voor een gering bedrag, maar er komt wel het e.e.a. voor kijken eer het zover is.

Meestal is het mogelijk de diskette op een andere lege diskette te kopiëren. Zie hiervoor de instructies onder „diskettes kopiëren” verderop in dit hoofdstuk. U kunt dan de copie gebruiken en het origineel opbergen.

Veel programma's zijn echter zo ontworpen dat u ze niet kunt kopiëren. Software-firma's doen dit om te vermijden dat u onbevoegd kopiëren aan een vriend geeft. Evenals bij boeken hebben de meeste programma's kopiërrechten en zonder toestemming van de auteur mag er geen copie worden gemaakt. Vooral bij zulke programma's moet u er altijd op letten dat u de originele disk niet beschadigt.

De DOS disk

We zijn de Atari DOS diskette al bij de opstartproceure van de disk drive tegengekomen. Deze diskette, soms met het etiket **MASTER DISKETTE**, bevat de instructies die de computer nodig heeft om met de disk drive te kunnen werken. Als u deze diskette verliest zonder er een copie van te hebben kunt u uw disk drive niet meer met uw computer gebruiken. DOS betekent Disk Operating System (disk besturingssysteem) en is in feite een programma dat zonder dat u het merkt door de computer wordt uitgevoerd.

Atari heeft drie verschillende versies van dit besturingssysteem: DOS 1.0, DOS 2.0S en DOS 3.0. DOS 1.0 was de originele versie die

vrij snel door DOS 2.0S werd vervangen. Beide versies werken nagenoeg hetzelfde en kunnen zowel op de nieuwe als op de oude disk drive gebruikt worden.

Tegelijkertijd met de introductie van het nieuwe model 1050 heeft Atari een nieuwe versie van de DOS ontworpen. Deze nieuwe versie, DOS 3.0, is speciaal ontworpen om gebruik te maken van de hogere snelheid en opslagdichtheid van de 1050 drive. Er zijn ook nog veel andere verbeteringen ten opzichte van de oudere versies in aangebracht. DOS 3.0 werkt ook op de oude drives.

Toen dit boek uitkwam was DOS 3.0 nog niet verkrijgbaar. Daarom zijn alle beschrijvingen in dit boek gericht op DOS 2.0S. Het is de versie die het meest wordt gebruikt. Als DOS 3.0 op de markt komt zal hij vrijwel gelijk zijn in gebruik. Het maakt dus geen verschil als u over de oude versie leest.

Op het moment dat u de computer met de DOS disk in de drive aanzet wordt een gedeelte van het DOS programma automatisch in het computergeheugen geladen. Hierdoor kan de computer op disk-instructies reageren die hij anders niet zou herkennen. U kunt uw computer nu gebruiken zoals u tot nu toe geleerd hebt, maar u kunt ook de nieuwe instructies gebruiken die nu aan zijn woordenschat zijn toegevoegd. Deze nieuwe instructies waaronder ook SAVE, LOAD en RUN worden verderop in dit hoofdstuk behandeld.

Ook kunt u het DOS programma op een meer zichtbare manier gebruiken door de volgende instructie te typen

DOS

Hierdoor schakelt de computer de normale stand uit waarin hij BASIC instructies en programmaregels accepteert en leest een ander gedeelte van het DOS programma in. Dit is een verzameling van *disk hulpinstructies (disk utilities)* waarmee het u mogelijk wordt speciale diskoperaties uit te voeren. Pas op want het gebruik van de DOS instructie wist een eventueel BASIC programma uit. Sla dat eerst op.

Typ de DOS instructie en druk op RETURN. De disk drive begint te zoemen, het beeld verandert en ziet eruit als op figuur 10.6. Dit is het *DOS menu*, omdat het alle keuzemogelijkheden die u hebt laat zien.

Over het algemeen geeft u de instructie door een letter uit dit menu in te typen en op RETURN te drukken. Als de computer meer informatie nodig heeft om de instructie uit te voeren meldt hij zich via het scherm.

Typ instructie A terwijl de DOS disk nog in de drive zit. Wij vragen nu de DISK DIRECTORY (diskinhoud) op. De *directory* is de lijst met namen van alle programma- en gegevensbestanden (file) die op de disk staan. Als u op RETURN drukt reageert de computer met de cryptische vraag:

DIRECTOR-SEARCH SPEC,LIST FILE?

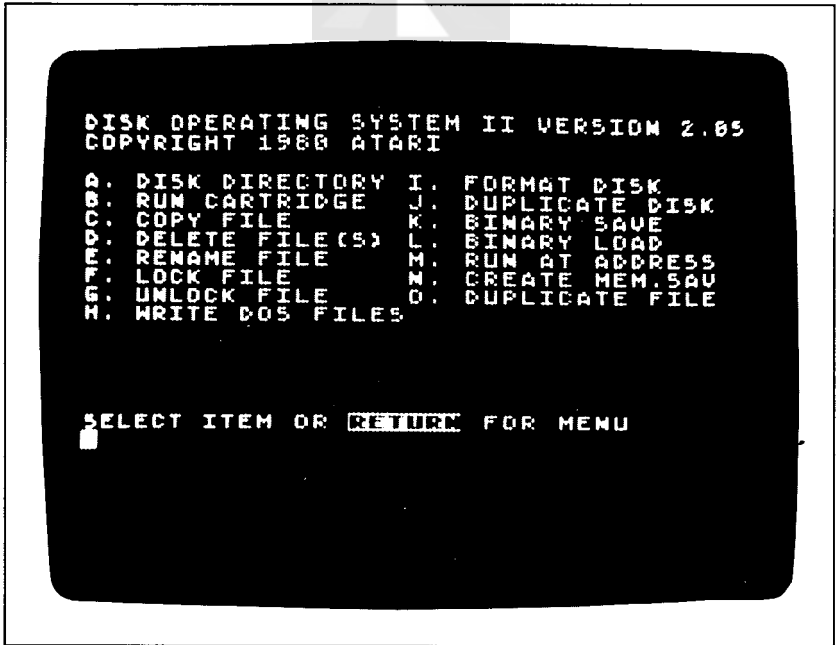
Inhoud-zoek spec,lijst bestand? Dit betekent: wilt u een bepaalde naam opzoeken of een lijst hebben van alle bestanden die er op de disk staan? Druk op RETURN om de inhoud van de gehele disk te zien. Als u versie 2.0S van het DOS programma hebt geeft de computer u waarschijnlijk drie namen, ongeveer zo:

```
DOS          SYS 039
DUP          SYS 042
AUTORUN     SYS 001
625 FREE SECTORS
```

De namen van de programma's en het aantal FREE SECTORS (vrije sectoren) kunnen op uw diskette anders zijn. Dit aantal geeft aan hoeveel ruimte er nog op de disk ter beschikking staat (625 is erg veel).

Als er veel verschillende bestanden op de disk zijn opgeslagen kan de directory een heel scherm vullen en het kan zelfs zijn dat enkele namen er aan de bovenkant afrollen. Om dit te voorkomen kunt u het afrollen van de lijst stoppen door op CONTROL en het cijfer 1 te drukken. Handel als volgt: geef zoals altijd de directory instructie en druk dan op CONTROL-1 als het overzicht op het scherm verschijnt. Het stopt op het punt waar het is zodat u kunt lezen wat er staat. Als u zover bent opnieuw op CONTROL-1 drukken en de lijst rolt af tot het einde.

Omdat u slechts één DOS diskette hebt is het verstandiger om in plaats van deze een lege te gebruiken. U krijgt lege diskettes in een



Figuur 10.6: Het DOS menu.

kantoorboekhandel, computerzaak of bij uw Atari-verkoper. Atari verkoopt lege diskettes onder zijn eigen naam, maar u kunt ook andere goede merken nemen. Meestal worden ze in verpakkingen van vijf of tien stuks verkocht.

Voordat u een lege diskette kunt gebruiken moet u hem eerst voor uw computer in sectoren indelen (format). Zonder dat kan de computer hem niet lezen. De computer zet met de disk drive magnetische tekens op de disk zodat hij een bepaalde plaats kan terugvinden als de disk ronddraait. Het is niet nodig dat u weet hoe dit werkt. De computer zorgt ervoor dat het gebeurt.

Neem de DOS diskette uit de disk drive en stop er een lege in. Typ dan I voor FORMAT DISK. De computer vraagt:

WHICH DRIVE TO FORMAT?

Welke drive moet er worden ingedeeld. Behalve als u meer dan één drive hebt moet u antwoorden met 1 of D1. De computer vraagt dan

TYPE "Y" TO FORMAT DISK 1

Typ „Y” om disk 1 in te delen. Hij vraagt of u er zeker van bent dat u deze diskette in wilt delen. Overtuig u dat de diskette leeg is omdat bij deze opdracht alle informatie die zich op de diskette bevindt gewist wordt. Als u er zeker van bent typ dan Y en druk op RETURN. U hoort ongeveer 45 seconden lang klikkende geluidjes. Daarna stopt de drive. U kunt pas doorgaan als het „in bedrijf lampje” uit is.

Voordat u de lege diskette nu als DOS diskette kunt gebruiken moet u nog een ding doen. Typ de H instructie, WRITE DOS FILES (DOS bestanden schrijven). Dit betekent voor de computer dat het DOS programma uit het geheugen op de lege diskette gecopiëerd moet worden. Antwoord op de twee vragen met D1 en Y en wacht dan op de melding

WRITING NEW DOS FILES

(„Schrijven van nieuwe DOS bestanden”)

Als de drive stopt bevat de diskette alle informatie die de computer nodig heeft om de disk drive te besturen. U kunt deze diskette gebruiken alsof het de originele DOS disk zou zijn.

Ik stel voor deze procedure iedere keer te volgen als u een nieuwe diskette gebruikt. Start de disk drive altijd met een diskette die het DOS programma bevat (dit kan zowel de originele DOS disk zijn als een copie). Deel de nieuwe lege diskette in en zet er de DOS bestanden op. U kunt deze disk als „copiëer-disk” voor andere gebruiken en de originele DOS diskette op een veilig plekje bewaren.

Bestanden

De rest van de nieuwe diskette kunt u verder gebruiken. U kunt er programma's, gegevens of andere informatie op opslaan.

De computer verdeelt de disk in *bestanden* (*files*). Dit zijn logische blokken met informatie die u op een gegeven moment wilt gebruiken. Een opgeslagen programma is normalerwijze een bestand. U slaat het als een eenheid op en haalt het als een eenheid terug. Andere bestanden die veel gebruikt worden bestaan uit gegevensblokken (bijvoorbeeld een verzendlijst) en *systeem hulpprogramma's* (*system utilities*) zoals het DOS programma.

Ieder bestand heeft zijn eigen naam. De computer houdt automatisch bij waar ieder bestand is opgeslagen zodat u alleen maar de naam hoeft aan te geven. De disk directory (kies A uit het DOS menu) is een lijst met namen van alle bestanden die op de diskette staan. U kunt maximaal 64 bestanden op een Atari diskette opslaan.

De naam van een bestand mag 8 letters bevatten. Hij mag ook cijfers bevatten. Alleen het eerste teken moet een letter zijn. Spaties en leestekens zijn niet toegestaan. Zoals bij de variabelen is het verstandig een naam te kiezen die de inhoud van het bestand weergeeft. Anders zou u gemakkelijk kunnen vergeten onder welke naam u de informatie opgeborgen heeft.

U kunt drie extra letters aan de bestandsnaam toevoegen (*extension*). Dit wordt vaak gebruikt om het soort van informatie aan te geven (BAS betekent BASIC, DAT betekent DATA = gegevens, SYS staat voor systeemprogramma zoals ook DOS). U kunt de toevoeging ook gebruiken om het verschil van verschillende versies van een programma aan te geven.

De volledige naam van een bestand bestaat uit drie delen: het drive-nummer (D1, behalve als u meerdere drives gebruikt), de bestandsnaam en de toevoeging. Typ een dubbelepunt (:) tussen drivenummer en bestandsnaam en een punt (.) tussen filenaam en toevoeging. Gewoonlijk kunt u het disknummer weglaten als u maar één drive gebruikt en dat geldt ook voor de toevoeging als er niets op staat. Hier volgen een aantal complete geldige bestandsnamen:

D1:DOS.SYS

D1:BESTAND.EXT

D1:PROG137.BAS

D1:ALLEENNAAM (geen toevoeging)

Nu u weet hoe u bestanden namen kunt geven kunt u enkele andere mogelijkheden van het DOS menu gebruiken. Met C, D en E kunt u een bestand op disk kopiëren, verwijderen en een nieuwe naam geven. Als voorbeeld maken we nu een copie van het DOS.SYS bestand en noemen die NIEUWBES. Typ een C en druk op RETURN. De computer reageert met

COPIE-FROM,TO?

Copie-van,naar? Hij vraagt u op te geven welk bestand gecopiëerd moet worden en welke naam het nieuwe bestand moet krijgen. Typ nu het volgende

DOS.SYS,NIEUWBES

Ziet u dat we het drivenummer D1 uit de bestandsnamen hebben weggelaten (als u niets anders aangeeft gaat hij van D1 uit). De bestandsnamen worden in deze regel door een komma gescheiden.

Als u op RETURN drukt gaat de drive zoemen en maakt een tweede exemplaar met alle informatie uit DOS.SYS. Als hij klaar is kunt u controleren of de nieuwe copie op de disk gezet is, door de directory op het scherm te vragen (instructie A). Als u met een nieuwe ingedeelde disk werkt zal er waarschijnlijk het volgende verschijnen

```
DOS          SYS 039
DUP          SYS 042
NIEUWBES    039
587 FREE SECTORS
```

Probeer de E-nieuwe naam en D-verwijderen instructies eens op dit NIEUWBES bestand uit. Ze werken allebei op dezelfde manier als de C-copiëer instructie. Bij „verwijderen” echter stelt de computer nog de vraag

```
TYPE "Y" TO DELETE ...
D1:NIEUWBES.?
```

(Typ "Y" om te verwijderen
Wilt u D1:NIEUWBES.?)

De computer vraagt dit om er zeker van te zijn dat u het bestand ook echt wilt verwijderen. Als u op Y en RETURN drukt is het bestand voor altijd weg.

Het opslaan van eigen programma's

U kunt de programma's die u zelf hebt geschreven nu gemakkelijk als bestand op disk zetten. U kunt een groot aantal verschillende programma's op een diskette zetten en ze gebruiken als u ze nodig hebt.

U kunt van uw programma's een bibliotheek opbouwen zodat u nooit een oud programma opnieuw hoeft in te typen. U zoekt alleen het bestand op de diskette op en laat het opgeslagen programma uitvoeren.

Om een programma te typen en op te slaan moet u het DOS menu verlaten en terugkeren naar de normale BASIC programmeertaal. Dit doet u door middel van de B-instructie, aangegeven met RUN CARTRIDGE in het DOS menu. Als u een van de nieuwe XL computers hebt waarbij BASIC ingebouwd is is dit misschien wat verwarrend. Deze procedure stamt nog uit de tijd dat de Atari BASIC cartridge ingestoken moest worden. Bij de nieuwe XL computers stuurt deze instructie de computer terug naar de bekende READY melding om aan te geven dat u nu normale instructies en programmaregels in kunt geven. In de toekomstige versie van het DOS programma zal Atari de woorden RUN CARTRIDGE waarschijnlijk wijzigen.

Als u het woordje READY op het scherm hebt kunt u het programma precies zoals u het wilt hebben intypen. Laat het eens lopen om er zeker van te zijn dat het werkt en gebruik dan de volgende instructie om het op diskette op te slaan

SAVE "D:BESTNAAM.EXT"

Voor BESTNAAM en EXT kunt u iedere naam en toevoeging gebruiken die u aan uw programma wilt geven. De bestandsnamen moeten altijd worden voorafgaan door een D: (of D1: als u het drive-nummer erbij moet geven) en alles moet tussen aanhalingstekens worden gezet. Als u D: niet aangeeft weet de computer niet of u de disk drive of een ander apparaat zoals bijvoorbeeld de programma recorder, bedoelt.

Elk programma dat u op diskette opslaat moet u een naam geven zodat de computer het kan vinden als u er weer mee wilt werken. Bij een programma recorder hoeft dat niet omdat de computer altijd het eerste programma dat hij op het bandje tegenkomt laadt.

Als u met SAVE een bestand met een naam op wilt slaan die op die diskette al werd gebruikt schrijft de computer er overheen en is het oude bestand weg. Dit is handig als u kleine veranderingen in een programma hebt gemaakt dat u had opgeslagen en waarbij de nieuwe versie nu de oude moet vervangen. U hoeft alleen maar over de oude

versie heen te schrijven. U moet natuurlijk oppassen dat u niet per ongeluk over een programma heenschrijft dat u niet kwijt wilt. Het is daarom belangrijk uw programma's namen te geven die duidelijk genoeg zijn zodat u ze niet per ongeluk opnieuw gebruikt.

Als u later een bepaald programma wilt terugladen typ dan

```
LOAD "D:BESTNAAM.EXT"
```

De computer vindt het aangegeven bestand en laadt het in zijn geheugen. U kunt LIST en RUN gebruiken alsof u het programma net hebt ingetypt. Zoals bij cassettes kunt u de LOAD en RUN instructies tegelijkertijd in een stap laten uitvoeren met de instructie

```
RUN "D:BESTNAAM.EXT"
```

De SAVE, LOAD en RUN instructies om op diskette op te slaan zijn hetzelfde als de cassette-instructies SAVE „C:”, LOAD „C:” en RUN „C:” die aan het einde van hoofdstuk 9 behandeld werden. C: en D: tussen aanhalingstekens gevat zijn voor de computer de aanwijzing dat òf de cassetterecorder òf de disk drive gebruikt moet worden gebruikt. Overigens zijn de opdrachten hetzelfde. De LIST en ENTER instructies gelden ook voor diskette. Om er achter te komen hoe die werken kunt u in hoofdstuk 9 onder „andere manieren om programma's op te slaan” lezen. Er zijn geen instructies die overeenkomen met CSAVE en CLOAD.

Diskettes kopiëren

In het DOS menu zit een handige instructie waarmee u een reservekopie (backup copie) van een hele diskette kunt maken: J voor DUPLICATE DISK.

Als het DOS menu nog niet op het scherm staat typ dan DOS, daarna typt u J en drukt dan op RETURN. De computer reageert met

```
DUP DISK-SOURCE,DEST DRIVES?
```

Copiëer disk-oorsprong, bestemmingsdrive?

Deze regel vraagt *van* welke drive u de informatie *naar* welke drive wilt kopiëren. Als u zoals meestal het geval zal zijn slechts één drive hebt moet u op beide vragen met 1 of D1 antwoorden. Typ het volgende en druk daarna op RETURN

D1,D1

Als u slechts over één disk drive beschikt moet u de originele diskette verwisselen tegen een tweede waarop de copie komt te staan. De computer gebruikt zijn geheugen als een opslagbuffer. Hij leest eerst de informatie van de originele disk, stopt dan zodat u de originele disk eruit kunt nemen en de tweede diskette erin kunt stoppen. Daarna maakt hij een copie van de gelezen informatie op de tweede disk. Wanneer het geheugen zoals meestal het geval is niet alle informatie in één keer kan bevatten, vraagt de computer u opnieuw de originele disk te plaatsen en daarna weer de copie zodat hij weer een gedeelte van de informatie kan kopiëren. Hij vraagt u zolang te wisselen totdat alles van de originele disk op de tweede overgenomen is.

Als u tenslotte klaar bent bevat de „gebruiksdisk” een exacte copie van de bestanden van het origineel. Deze mogelijkheid met DUPLICATE DISK kunt u zo vaak gebruiken als u wilt om kopiën van uw belangrijkste diskettes te maken. Door met kopiën te werken verliest u niets van de belangrijke informatie als er eens iets mee zou gebeuren.

Als u meer dan één disk drive hebt hoeft u niet steeds de diskettes te verwisselen. Stop de originele disk in één van de drives en de disk waarop de copie komt te staan in de andere. De computer zet dan direct alle informatie van de ene disk op de andere zonder dat hij vraagt de diskettes te verwisselen.

Dit is erg handig en daarom één van de belangrijkste redenen waarom sommige mensen een tweede drive kopen. Hoewel u met twee drives niet meer mogelijkheden hebt als met één drive, worden met twee drives veel diskhandelingen wel veel eenvoudiger.

U sluit de tweede drive op dezelfde manier aan als de eerste. U moet nu alleen het kabeltje in de achterkant van de eerste drive steken in plaats van in de PERIPHERAL opening op de computer. Schakel met een pen het zwarte schakelaartje in de opening aan de achterkant van de tweede drive om.

De tweede drive kunt u op dezelfde manier als de eerste gebruiken. U moet er alleen met D2 naar verwijzen. U kunt SAVE, LOAD en RUN gebruiken op programma's door D2 als een deel van de bestandsnaam te gebruiken:

SAVE "D2:BESTNAAM.EXT"

Het enige wat u met de tweede drive niet kunt doen is het systeem opstarten. De computer ontleent namelijk altijd zijn DOS instructies aan drive nummer 1.

Samenvatting

Voor permanente opslag is een disk drive sneller, flexibeler en betrouwbaarder dan een programma recorder. U kunt veel verschillende programma's op een enkele diskette opslaan en ze met behulp van hun naam terughalen. U kunt ook commerciële diskprogramma's laden en in enkele seconden laten lopen.

Sommige commerciële programma's laden zichzelf zonder dat u hiervoor een speciale instructie geeft, terwijl andere een LOAD en RUN opdracht nodig hebben.

Om uw programma op te slaan of te laden moet u een Atari Disk Operating System (DOS) disk gebruiken. Als u de computer met deze disk in de drive aanzet zal hij automatisch enkele extra instructies laden die het opslaan het terughalen van programma's mogelijk maken. U kunt ook de DOS instructie geven voor meer specifieke disk handelingen zoals bijvoorbeeld het indelen van een lege diskette, verwijderen van bestanden of het kopiëren van hele disks.

Hoofdstuk 11

Nawoord

Tot zover hebt u in de loop van dit boek met veel dingen kennis gemaakt die u met uw computer kunt doen. U leerde hoe u hem moest installeren en hoe u kant en klare software kunt gebruiken. U leerde eigen instructies te geven en programma's te schrijven waarmee u de computer op verschillende manieren kunt besturen. In het derde deel bent u vertrouwd geraakt met de mogelijkheden van de Atari Programma Recorder en de Disk Drive die u op uw systeem kunt aansluiten om uw programma's blijvend op te slaan.

Dit alles is echter pas het begin. Met deze basiskennis kunt u uw ontdekkingsreis voortzetten. U kunt een printer (drukker) of andere apparatuur aansluiten en complexe programma's schrijven die gecompliceerde problemen oplossen.

In de laatste bladzijden wil ik u een indruk geven van de vele mogelijkheden die er nog zijn. Ik kan echter niet alles laten zien. Als u gedetailleerde informatie wilt hebben zou u één van de boeken uit te literatuurlijst kunnen raadplegen of zelf een beetje kunnen experimenteren.

Andere uitbreidingsmogelijkheden

De drie *printers* die Atari in zijn leveringsprogramma heeft zijn al aan het einde van hoofdstuk 1 besproken. Zij verschillen in vele opzichten, maar hebben alle drie dezelfde functie: zij nemen de informatie die in de computer zit over en brengen deze op papier.

Een printer is om een heleboel redenen zinvol. Bijvoorbeeld voor het geval dat u een duurzaam overzicht van een programma wilt hebben zodat u zeker weet dat u het correct intypt als het verloren zou

zijn gegaan. Of wanneer u een lang programma schrijft; u kunt het dan met behulp van de computer in één keer lezen in plaats van in stukjes op het scherm. En natuurlijk hebt u een printer nodig als u de computer als tekstverwerker gebruikt.

U sluit de printer op de computer aan via de contactopening waarbij PERIPHERAL staat. Dus op dezelfde manier alsof u een programma recorder of een disk drive aansluit. Als u meerdere randapparaten tegelijk wilt gebruiken kunt u ze als een ketting verbinden zoals in het hoofdstuk over de disk drive beschreven.

Als u de printer hebt aangesloten en aangezet kunt u hem zonder problemen gebruiken. Veel financiële en tekstverwerkende programma's drukken met gebruikmaking van een eenvoudige opdracht tabellen af.

Als uw eigen programma in het computergeheugen zit kunt u hiervan een copie op papier zetten. Daartoe moet u de volgende variatie op de standaard LIST instructie gebruiken:

LIST "P:"

De computer typt het programma dan regel voor regel uit zoals het op het beeldscherm zou verschijnen. „P:” in de opdracht zegt de computer dat het programma naar de printer moet in plaats van naar het scherm.

U kunt de printer ook voor andere dingen gebruiken dan voorprogrammaweergave. U kunt boodschappen of lijsten metgetallen op papier afdrukken in plaats van op het scherm. Bij model 1025 kunt u op een speciale manier afdrukken of de regelspatiëring veranderen. Met model 1020 printer/plotter (plotter = tekenapparaat) kunt u zelfs bepaalde grafische tekens afdrukken.

Atari is van plan met een hele lijn randapparatuur te komen die bij de nieuwe XL computers aansluiten. Sommige apparaten zullen misschien nooit verwezelijkt worden, maar het is in ieder geval de moeite waard om er naar uit te kijken.

Een interessant product is de grafische lei. Dit is een speciaal elektronisch veld waarop u direct grafische tekeningen kunt maken zonder te programmeren. U tekent met een speciale pen op de lei en de com-

puter voelt precies uw bewegingen. Met een denkbeeldige verfkwast tekent hij -precies uw bewegingen volgend- dat wat u tekent na.

Nog interessanter is waarschijnlijk de *lichtpen*. Hij werd al enkele jaren geleden door Atari ontwikkeld. Dit is een elektronisch stokje waarmee u iets op het scherm aan kunt wijzen. Met een speciale schakeling kan de computer exact waarnemen naar welk punt u wijst en kan de cursor naar dat punt bewegen of hem een bepaalde kleur geven. Als u dit doet lijkt het erop alsof de pen op uw scherm lijnen trekt.

Atari heeft ook een *uitbreidingssysteem* gepland. Dit is een grote doos waarmee het mogelijk wordt diverse uitbreidingen op uw computer aan te sluiten waaronder ook apparatuur die niet door Atari is gemaakt.

Op het ogenblik zijn deze bijzetapparaten nog niet zeker. Welzijn ze allemaal officieel door Atari aangekondigd, maar het kan zijn dat ze nooit in de handel gebracht worden. Uw leverancier kan u hierover gedetailleerde informatie geven.

Andere wegen

Als u uit dit boek het gedeelte over het programmeren hebt gelezen hebt u verschillende manieren geleerd waarop u uw computer kunt besturen. U kunt een programma ontwerpen, variabelen gebruiken om berekeningen te maken en de programmaloop met lussen en beslissingen besturen. Met behulp van de speciale grafische instructies kunt u op verschillende manieren tekeningen op het beeld brengen.

Bedenk echter dat alleen de grondslag is gelegd en dat u nog veel meer kunt doen. Dit blijft slechts het begin.

De BASIC programmeertaal heeft specifieke eigenschappen waarmee u de computer op andere wijze kunt besturen. U kunt bijvoorbeeld *formaties (arrays)* gebruiken waarmee u een hele tabel met gegevens in het geheugen kunt opslaan in plaats van ze aparte namen te geven. U kunt woorden en getallen als *gegevens (data)* op cassette of diskette opslaan. U kunt deze informatie later terughalen en in andere programma's gebruiken.

U kunt in uw Atari computer ook enkele andere programmeertalen gebruiken. Een van de populairste is Atari PILOT, een grafische taal voor kinderen. In deze taal geeft u instructies om een klein *schildpad* over het scherm te bewegen. Terwijl hij zich beweegt laat de schildpad een spoor achter waarmee kinderen creatieve beelden kunnen maken.

Bij elke programmeertaal moet de computer de Engelse instructies vertalen in de code die hij zelf gebruikt. Dit is een langzaam en inefficiënt proces. Sommige mensen slaan daarom dit proces bij complexe programma's die vlug moeten gaan over. Ze doen dit met de *machinetaal* die de computer direct bestuurt.

Met de machinetaal geeft u de instructies op het diepste niveau van de computer en u kunt hiermee zijn werking direct besturen. Enkele van de geavanceerde grafische mogelijkheden van de Atari computer kunnen alleen worden bestuurd via de directe machinetaal-instructies. De grappige figuurtjes en voorwerpen uit videospelen zijn hiervan een voorbeeld. Met machinetaal wordt de computer ook sneller omdat daarmee het uitvoeren van inefficiënte handelingen teruggedrongen wordt.

Helaas is machinetaal erg technisch en moeilijk te gebruiken omdat u gedwongen wordt elke handeling van de computer op de voet te volgen. U moet een heleboel over de werking van de computer kennen en bereid zijn om saaie programma's te maken. Zelfs beroepsprogrammeurs vermijden machinetaal als het niet perse nodig is deze te gebruiken.

Machinetaal heeft veel weg van *assembleertaal*, die iets beter te begrijpen is. Hoewel u nog steeds pietlutterige instructies moet geven voor iedere handeling die de computer moet uitvoeren kunt u labels gebruiken en verbale instructies in plaats van genummerde codes. De computer kan de assembleertal direct in machinetaal vertalen.

Sommige boeken en tijdschriften gebruiken een tussenvorm tussen BASIC en machinetaal door twee specifieke BASIC instructies te gebruiken: PEEK en POKE. Hiermee kunt u variabelen direct in het computergeheugen lezen en schrijven zonder ze een variabele naam te geven. Op deze wijze kunt u bepaalde vormen van directe besturing over de interne handelingen van de computer toepassen wat u

anders met machinetaal zou moeten doen. Dit proces is echter nogal omslachtig en moeilijk te volgen.

Wat u ook doet, ik hoop dat u als u dit boek uithebt uw ontdekkings-
tocht voortzet. Er zijn nog veel andere boeken waaruit u veel kunt le-
ren, meer nog dan ik u op de voorafgaande bladzijden heb kunnen
vertellen. Uw belangrijkste bron bent u echter zelf. Wees vooral cre-
atief en speel met uw computer. Welke weg u ook inslaat, met uw
computer kunt u een heleboel leuke dingen doen.

Verdere Literatuur

Er zijn heel wat boeken geschreven over de Atari computers. Ik kan ze niet allemaal vermelden. Hier volgen er een paar waarin u nuttige informatie kunt vinden.

Inside ATARI BASIC *Bill Carris (Reston, 1983)*

Dit is een van de aantrekkelijkste boeken dat geschreven is met betrekking tot de inleiding op het programmeren op de Atari in BASIC. Het is voorzien van aantrekkelijke afbeeldingen en nuttige programmeertips. Vooral op grafisch gebied is het erg goed omdat de GTIA standen er duidelijk en uitvoerig in behandeld worden. Het is erg prettig geschreven.

Atari Sound and Graphics *Herb Moore, Judy Lower en Bob Albrecht (Wiley, 1982)*

Dit is een inleiding op BASIC met speciale aandacht voor de geluidsen grafische mogelijkheden van Atari. Ofschoon men er meer uitgebreid over had kunnen schrijven is het boek interessant en één manier om uw computer te leren kennen.

Your First Atari Program *Rodney Zaks (Sybex, 1984)*

Een goede inleiding van de fundamentele technieken van de BASIC programmering. Het behandelt hoofdzakelijk de manier waarop u moet denken wanneer u een programma schijft. De goede illustraties van Daniel Le Noury verduidelijken een en ander.

Your Atari Computer *Lon Poole, Martin McNiff en Steven Cook*
(Osborne/McGraw-Hill, 1982)

In dit boek dat 450 bladzijden beslaat vindt u alle informatie die u ooit nodig zult hebben met betrekking tot de Atari computer en zijn randapparatuur. Iedereen die zich serieus met het schrijven van programma's bezig houdt moet het gelezen hebben en het is tevens een geweldig naslagwerk. Maar het is niet gemakkelijk. De beschrijvingen zijn nogal technisch en gaan ervan uit dat u een bepaalde voorkennis hebt. Zolang er nog geen herziene druk is zijn de verbeteringen die in de XL serie zijn ingebouwd niet behandeld.

Atari BASIC Programs in Minutes *Stanley R. Trost* (Sybex, 1984)

Een verzameling korte programma's die u direct in uw Atari computer kunt typen en gebruiken zonder BASIC programmeerkennis. Het bevat financiële, privé en studie programma's.

The Book of Atari Software 1983 *Bewerking door Jeffrey Stanton, Robert P. Wells en Sandra Rochowansky* (Addison-Wesley, 1983)

Een catalogus met alle programma's die begin 1983 voor de Atari computers verkrijgbaar waren. Elk programma heeft een speciale code gekregen en wordt kort besproken. Ofschoon u met een halve bladzijde uitleg niet alles over het programma kunt te weten komen geeft het toch een overzicht van wat er te koop is. Herziene drukken zullen waarschijnlijk volgen.

Bijlage B

BASIC woordenlijst

Dit is geen volledige woordenlijst van de BASIC taal, maar een samenvatting van de instructies die in dit boek werden behandeld. Voor een complete lijst verwijzen wij naar hoofdstuk 11 van het boek *Your Atari Computer* geschreven door Poole, McNiff en Cook (Osborne/McGraw-Hill, 1982).

CLOAD De computer laadt een programma vanaf de cassette in zijn geheugen. Hij piept één keer om aan te geven dat u op PLAY van uw programma recorder moet drukken en laadt dan het programma. Een eventueel programma dat zich nog in het geheugen bevindt wordt gewist.

COLOR *n* Kiest de verfkwasst die bij de PLOT en DRAWTO opdrachten gebruikt wordt. De computer gebruikt de aangegeven verfkwasst totdat u een nieuwe COLOR opdracht geeft. In speciale grafische standen heeft de COLOR opdracht andere betekenissen (zie hoofdstuk 8).

CSAVE Slaat een programma op cassette op. De computer piept twee keer om aan te geven dat u op PLAY en RECORD van de programma recorder moet drukken en neemt dan het programma op. Het programma wordt niet tegelijkertijd uit het computergeheugen gehaald.

DATA Wordt samen met de READ opdracht gebruikt om getallen aan variabelen te geven. De getallen die in de lijst achter het woordje DATA staan worden in volgorde aan de corresponderende variabelen in de READ opdracht toegekend. Als niet alle getallen

uit de DATA opdracht worden toegekend zullen ze door volgende READ opdrachten worden gebruikt, beginnend met het getal dat de laatste READ opdracht liet staan.

DIM Reserveert een ruimte voor een tekstvariabele of een rij getallen.

DIM A\$(12)

maakt ruimte voor 12 letters in de tekstvariabele A\$.

DOS Betekent voor de computer dat u het Disk Operation System programma wilt invoeren zodat u bepaalde diskhandelingen uit kunt voeren zoals bijvoorbeeld het indelen van een diskette voor uw computer, kopiëren van een gedeelte of van een hele diskette. U kunt deze instructie alleen dan gebruiken als u de computer met de Atari DOS diskette in de drive volgens de procedure beschreven in hoofdstuk 10 hebt aangezet.

DRAWTO x,y Trekt in de grafische standen een lijn naar de coördinaten x,y vanaf de plaats waarop de verfkvast dan staat. De verfkvast bevindt zich in de laatste stand die u met de COLOR opdracht hebt gegeven. De coördinaten moeten binnen het gegeven bereik van de door u gebruikte grafische stand liggen.

END Geeft het einde van het programma aan. U kunt het weglaten als u een normaal programma hebt dat van het begin tot het einde loopt. Als u echter deelprogramma's gebruikt of andere opdrachten die op de laatste programmaopdracht volgen moet u deze instructie gebruiken om de computer te stoppen.

ENTER Wordt bij de programma recorder of de disk drive gebruikt om een programma te laden dat met de LIST instructie is opgeslagen. Bij een cassette moet u aangeven:

ENTER "C:"



Bij een disk drive moet u de naam van het terug te halen bestand opgeven:

ENTER "D:BESTNAAM.EXT"

De ENTER instructie wist het computergeheugen niet schoon. Het voegt slechts de regels toe aan het reeds aanwezige programma.

FOR/NEXT De FOR opdracht geeft het begin van een lus aan en zegt de computer hoe vaak hij deze moet herhalen. De algemene vorm is:

FOR I=a TO b STEP c

U moet de tellervariabele (I), zijn startwaarde (a) en zijn eindwaarde (b) aangeven. U kunt tevens de stapgrootte (c) aangeven die de teller bij elke herhaling van de lus moet maken.

Als u die niet aangeeft gaat hij van 1 uit. FOR moet altijd gevolgd worden door een NEXT opdracht waarmee het einde van de lus aangegeven wordt.

GOSUB *n* Hiermee voert de computer het deelprogramma uit dat bij opdrachtnummer *n* begint. Hij loopt het deelprogramma door totdat hij de RETURN opdracht bereikt die het einde aangeeft. Daarna gaat de computer met het hoofdprogramma na de GOSUB opdracht door.

GOTO *n* Geeft de computer aan dat de uitvoering bij opdrachtnummer *n* verdergaat.

GRAPHICS *n* Maakt het scherm schoon en wisselt naar de nieuwe grafische stand. Het getal *n* geeft de nieuwe stand aan. Het meest gebruikt worden 0 (tekst), 7 (vierkleuren grafieken) en 8 (grafieken met zeer fijn raster). Als u de stand aangeeft waarin u zich bevindt verandert de stand niet, maar het scherm wordt wel schoongemaakt. Door 16 aan het nummer toe te voegen kunt u het tekstraam aan de onderkant van het scherm verwijderen.

IF/THEN Geeft aan dat er een beslissing genomen moet worden. Als aan de voorwaarde achter IF wordt voldaan zal de computer de opdracht die achter THEN staat uitvoeren. Als niet aan de voorwaarde wordt voldaan gaat de computer met de volgende opdracht verder.

INPUT Zorgt ervoor dat de computer bij elke keer dat hij de opdracht tegenkomt het programma stopt en naar een getal of stuk tekst vraagt dat aan een variabele moet worden toegekend. Hij drukt een vraagteken af en wacht op het antwoord dat door de RETURN instructie moet worden gevolgd. Dan wordt het getal aan de variabele die in de INPUT opdracht is aangeduid toegekend.

LIST Zonder toevoeging, brengt de computer bij het woordje LIST het programma dat is opgeslagen op het scherm. Als LIST door een opdrachtnummer wordt gevolgt verschijnt alleen die opdracht. Als u twee getallen aangeeft toont de computer de opdrachten die zich tussen deze getallen bevinden.

LIST kan ook worden gebruikt om een programma op te slaan.

LIST "C:"

geeft aan dat de computer de programma-lijst naar de recorder moet sturen in plaats van naar het scherm. Als u een disk drive hebt moet u de bestandsnaam aangeven:

LIST "D:BESTNAAM.EXT"

Een programma dat u met LIST hebt opgeslagen kunt u alleen met ENTER terughalen.

LOAD Laadt een programma in dat op cassette of diskette is opgeslagen. Typ om van cassette te laden:

LOAD "C:"

Als u met diskettes werkt moet u ook de naam van het bestand aangeven:

LOAD "D:BESTNAAM.EXT"

De computer wist het programma dat zich in het geheugen bevindt voordat hij het nieuwe opslaat. **LOAD** werkt alleen op bestanden die met **SAVE** opgeslagen zijn.

NEW Maakt het programmeergeheugen van de computer schoon zodat u een nieuw programma kunt intypen. Als u deze instructie vergeet te gebruiken kunnen er oude regels van een vorig programma tussen het nieuwe programma komen.

NEXT Geeft het einde van een **FOR/NEXT** lus aan. Aan deze opdracht moet altijd een **FOR** opdracht vooraf gaan. De naam van de variabele achter **NEXT** moet corresponderen met de tellervariabele in de **FOR** opdracht.

PLOT x,y In de grafische stand zet deze instructie een punt op de coördinaten x,y . Wanneer zich de verfkwast op een andere plaats op het scherm bevindt springt hij naar de nieuwe, aangegeven plaats zonder een lijn op het scherm te trekken. Als een lijn wordt getrokken begint deze bij het nieuwe punt. De coördinaten moeten binnen het bereik van de gebruikte grafische stand liggen.

POSITION x,y In de tekststand verplaatst u met **POSITION** de cursor naar de coördinaten x,y . Een volgende **PRINT** opdracht laat de tekst op dat punt beginnen.

PRINT Drukt op het scherm een boodschap af. Als u een woord tussen aanhalingstekens hebt gezet, drukt de computer het af zoals u het hebt ingetypt. Als u geen aanhalingstekens gebruikt behandelt de computer het woordje als de naam van een variabele en drukt de daarvoor geldende waarde af. U kunt verschillende boodschappen in een enkele **PRINT** opdracht stoppen en ze met behulp van een puntkomma (;) scheiden. Een puntkomma aan het einde voorkomt dat de computer de volgende boodschap op de volgende regel neerzet.

In de speciale tekststanden 1, 2, 12 en 13 kunt u teksten op het grafische beeld krijgen met de instructie

PRINT #6;"boodschap"

READ Werkt in combinatie met de DATA opdracht om aan een serie variabelen een reeks van getallen toe te kennen. Aan elke variabele in de READ opdracht wordt het corresponderende getal in de DATA opdracht toegekend. De eerste READ opdracht in een programma laadt de getallen in beginnend bij de eerste DATA opdracht. Latere READ opdrachten beginnen met het toekennen van het eerste nog niet gebruikte getal in de DATA opdrachtenlijst.

RUN Geeft aan dat de computer het programma dat zich in het geheugen bevindt uit moet voeren. Programmaregels worden in de volgorde van hun opdrachtnummers gelezen, behalve wanneer GOTO, GOSUB of FOR/NEXT opdrachten de loop veranderen. Aan het einde stopt de computer met de melding READY. Het programma blijft in het geheugen.

SAVE Neemt het programma dat zich in het computergeheugen bevindt op cassette of diskette op. Als u cassettes gebruikt typt u

SAVE "C:"

Bij disk drives moet u ook de bestaandsnaam aangeven waaronder u het programma op wilt slaan:

SAVE "D:BESTNAAM.EXT"

Met LOAD laadt u een programma in dat u met SAVE hebt opgeslagen.

SETCOLOR Verandert de kleuren van de verfkwasten met behulp van de codes uit figuur 8.1. De computer voert eerst eventuele berekeningen uit rechts van het is-gelijk teken en slaat daarna het resultaat in de variabele aan de linkerkant op. Bijvoorbeeld de opdracht

J=2+2

geeft aan de variabele J de waarde 4.

Bijlage C

Atari foutcodes

Dit is een overzicht van de Atari foutcodes die het meest voorkomen met een verklaring ervan en de meest waarschijnlijke oorzaak. Niet alle codes zijn beschreven. In ieder geval wel die welke u in dit boek bent tegengekomen. Als u grafieken tekent en zeker als u een disk drive gebruikt kan het voorkomen dat u nog meer onwaarschijnlijke fouten tegenkomt. U vindt een complete lijst in de *Atari BASIC Reference Guide* die bij uw computer meegeleverd werd. Voor een uitvoerigere uitleg van foutcodes die niet veel voorkomen kunt u de bijlage van het boek *Your Atari Computer* van Poole, McNiff en Cook (Osborne/McGraw-Hill, 1982) raadplegen.

2 Onvoldoende geheugen

Afhankelijk van het Atari-model beschikt uw computergeheugen over 16 of 64 K bytes. Programmaregels, variabelen en het grafisch beeld nemen allemaal een gedeelte van het geheugen in beslag. Op de Atari 600XL komt deze foutmelding vaak voor bij grafieken met een heel fijn raster.

3 Waarde ongeldig

U hebt een getal gebruikt dat voor de instructie of te groot of te klein is. Dit is een fout die vaak voorkomt. Zij kan ontstaan als u een te hoog opdrachtnummer gebruikt (het maximale kan zijn 32767) of als u met PLOT of DRAWTO coördinaten buiten het beeldbereik aangeeft. Ook bij veel andere opdrachten kan deze fout voorkomen.

4 Te veel variabelen

U kunt maximaal 128 variabelen in een programma gebruiken. Zelfs als u een variabele gebruikt en daarna weer verwijdert telt hij mee. Een NEW instructie veegt de hele lijst van variabelen weg.

- 6 Te weinig Data-waarden**

Er bevonden zich in de DATA opdracht niet genoeg waarden om alle variabelen in een READ opdracht ermee te kunnen voorzien.
- 7 Getal groter dan 32767**

Opdrachtnummer of diverse andere waarden moeten kleiner zijn dan 32767.
- 8 INPUT opdracht fout**

Het getal of de tekst welke u als antwoord op een INPUT vraag intypte kon niet in de variabele worden opgeslagen. Dit komt vaak voor als u een letter of een leesteken in het midden van een getal typt. Het gebeurt ook als u op RETURN drukt zonder een antwoord te hebben geven.
- 9 Dimensie bij formatie of reeks fout**

Voordat u een formatie of een lus kunt gebruiken moet u er met een DIM opdracht plaats voor reserveren.
- 11 Reëel getal te klein/te groot**

De computer kan met zeer hoge getallen rekenen (tot 10^{98}). Het kan echter gebeuren dat u daar boven uitkomt, bijvoorbeeld als u door nul deelt.
- 12 Regel niet gevonden**

U hebt geprobeerd naar een regelnummer te springen dat niet bestaat. Controleer uw GOTO en GOSUB opdrachten.
- 13 FOR opdracht ontbreekt**

Bij een NEXT opdracht hoort een FOR opdracht met dezelfde tellervariabele. Het kan zijn dat u een FOR of NEXT opdracht bent vergeten of een lus niet volledig binnen een andere hebt geschreven.
- 14 Regel te lang**

Instructies kunnen maximaal 114 tekens of 3 regels van het scherm bevatten.

15 GOSUB of FOR regel verwijderd

Een RETURN of NEXT opdracht is niet op de juiste wijze gebruikt in combinatie met een GOSUB of FOR opdracht.

16 RETURN fout

De computer is een RETURN opdracht tegengekomen zonder naar een deelprogramma te zijn verwezen. Er ontbreekt een GOSUB opdracht of u hebt END aan het einde van het hoofdprogramma vergeten.

19 LOAD programma te lang

U hebt geprobeerd een programma van diskette of bandje te laden dat niet in het computergeheugen past.

21 LOAD bestand fout

Met LOAD kunt u alleen een bestand inladen dat met SAVE is opgenomen. Programma's die met CSAVE of LIST zijn opgeslagen moeten met resp. CLOAD en ENTER worden geladen.

130 Onbekend randapparaat

U hebt geprobeerd een randapparaat te gebruiken dat niet op uw systeem is aangesloten of u hebt vergeten D: (toestelaanduiding) aan de bestandsnaam van de diskette vooraf te laten gaan.

133 Bestand of apparaat niet toegankelijk

Kan bij het gebruik van PLOT of DRAWTO in een tekststand voorkomen. Nadat een grafisch programma beëindigd is moet het scherm voordat u PLOT of DRAWTO kunt gebruiken met een GRAPHICS opdracht schoon en toegankelijk worden gemaakt.

138 Wachtijd te lang

De computer heeft binnen de daartoe vastgestelde tijd van een randapparaat geen reactie gekregen. Het kan zijn dat het randapparaat niet juist is aangesloten.

141 Cursor buiten beeldbereik

U hebt geprobeerd om met POSITION de cursor buiten het beeldbereik van de door u gebruikte tekststand te plaatsen.

144 Apparaatuitvoering fout

U hebt geprobeerd om een nieuw programma op te slaan op een diskette die tegen het opslaan van een ander programma was beveiligd.

147 Onvoldoende scherm RAM

U hebt geprobeerd een grafische stand te gebruiken die de grenzen van uw computergeheugen overschreed. Deze fout kan vaak voorkomen bij grafieken met een fijn raster op Atari 600XL.

162 Disk vol

De ruimte die er op een diskette ter beschikking staat is groot, maar begrensd. Als u een aantal erg lange programma's op dezelfde diskette opslaat kan het zijn dat u deze foutmelding krijgt.

165 Bestandsnaam fout

De bestandsnamen op een diskette moeten met een letter beginnen en kunnen alleen hoofdletters en cijfers bevatten.

169 Directory vol Er passen op een diskette maximaal 64 bestanden, zelfs wanneer er nog ruimte over is om informatie in op te slaan. Misschien kunt u enkele bestanden wissen zodat u toch weer plaats krijgt.

170 Bestand niet gevonden

U hebt geprobeerd een bestand te laden dat zich niet op de diskette bevindt. U hebt of een verkeerde naam aangegeven of de verkeerde diskette gebruikt.

Oplossingen van de geselecteerde oefeningen

3.2: Geef de instructie

```
PRINT "EEN":PRINT "TWEETWEETWEET":PRINT "DRIE"
```

Om hetzelfde in slechts één opdracht te doen moet u de volgende regel typen en daarbij ESC, CONTROL en de omlaag toets op de plaatsen waar het ↓ symbool staat, gebruiken

```
PRINT "EEN ↓TWEETWEET ↓DRIE"
```

3.3: De computer begint met het afdrukken van de tekst direct op de regel onder de instructie. Voordat hij de kans kreeg het woordje HOGER neer te zetten, plaatste de eerste pijl de cursor op de P van PRINT. De computer drukt het woordje HOGER over het woordje PRINT af en gaat twee regels verder omhoog en drukt het woordje LAGER af. Dat is dan het einde van de tekst dus gaat hij weer twee regels omlaag en meldt zoals altijd READY. Maar weer wordt het woordje door een ander woordje - en wel HOGER dat net werd afgedrukt heen geschreven. Dat ziet er dan zo uit:

```
LAGER
```

```
READY " ↑HOGER ↑↑ LAGER"
```

4.1: U kunt voor de drie namen veel posities kiezen. Dit zou er een kunnen zijn:

```
POSITION 10,7:PRINT "TOM":POSITION 27,  
11:PRINT "TOM":POSITION 0,20:PRINT "TO  
M"
```

4.2: Typ de zeven instructies uit hoofdstuk 4 onder „vierkleuren tekening: Graphics 7” die de vierhoek van figuur 4.6 tekenen en voeg er nog zes aan toe. Bijvoorbeeld:

```
COLOR 3
PLOT 20,20
DRAWTO 60,20
DRAWTO 60,60
DRAWTO 20,60
DRAWTO 20,20
```

5.1: Voeg deze regel aan het programma toe:

```
65 COLOR 3
```

5.2: Voeg deze regel toe:

```
80 DRAWTO 70,70
```

Als u net oefening 1 hebt gedaan is de diagonale lijn blauw.

5.3: De drie fouten zijn:

1. Het woordje PRINT in regel 10 is niet juist gespeld.
2. De tweede coördinaat (734) ligt buiten het toegestane bereik.
3. De DRAWTO opdracht in regel 60 bevat slechts een coördinaat.

6.1: Voeg opdrachtnummer 15 toe en vervang opdracht 20 als volgt:

```
15 PRINT "KLEURNUMMER";INPUT N
20 COLOR N
```

6.2: Voeg deze opdrachten aan het programma dat de driehoek van figuur 6.9 tekent toe:

```
120 REM OMGEKEERDE DRIEHOEK
130 PLOT X,Y+20
140 DRAWTO X+20,Y-10
150 DRAWTO X+40,Y+20
160 DRAWTO X,Y+20
```

6.3: De nieuwe waarde van J zal gelijk zijn aan de oude waarde (10) plus 1 of 11.

7.1:

```
10 PRINT "WELK GETAL";:INPUT GETAL
20 IF GETAL>0 THEN PRINT "POSITIEF"
30 IF GETAL<0 THEN PRINT "NEGATIEF"
40 IF GETAL=0 THEN PRINT "NUL"
```

7.2:

```
1000 REM DEELPROGRAMMA OM SCHERM SCHOON TE
MAKEN
1010 PRINT "(ESC-CONTROL-CLEAR)"
1020 RETURN
```

7.3: Probeer eens het volgende programma:

```
10 TOTAAL=0
20 FOR I=1 TO 100
30 TOTAAL=TOTAAL+1
40 NEXT I
50 PRINT TOTAAL
```

Het antwoord moet 5050 zijn.

Trefwoordenlijst

- Aan/uit schakelaar** 26
Aansluiting 28
 van disk drive 179
Adaptor *zie* Aansluiting
Aftreksymbool (–) 118
APX (Atari Programm Exchange) 33
Arrays *zie* Formaties
Assembleertaal 202
Atari computers 20
Atari composer 39
AtariWriter 39
Automatische herhaling 44
- BACK SPACE toets** 45, 50
Backup cassettes 174
**BASIC programmeer-
 taal** 201–202, 207–212
 cartridge voor 21–22
 machinetaal en 201–202
 OPTION uitschakeltoets 40
Beeldomkeertoets *zie* Inverse-videotoets
Benoemen van bestanden 192–194
Berekeningen 117–121
Beslissingen 138–142
Bestanden 138–142
Blanco regels 58, 59
BOOT ERROR 186
BREAK toets 45, 47
 stoppen van programma met 116, 127
Bytes 18
- CAPS toets** 45, 46
Cartridge programma's 32, 38
Cassetterecorder *zie* Programma recorder
Cassettebandjes 32, 169–175
 nadelen van 28, 32, 178
 programma laden van 39, 40, 196
 programma opslaan
 op 96–98, 169, 173–175, 195, 198
CLEAR toets 50
 deelprogramma met 145
 in instructies 58–59
 voor **POSITION** instructie 69
CLOAD instructie 98, 169–175, 196
COLOR instructie 149, 159
 in grafische stand 66
 in GTIA standen 159–160
 in grafieken met een fijn raster 158
 in tegenstelling tot **SETCOLOR** 152
 in programma vierhoektekenen 109
CONT instructie 127
CONTROL toets 48, 49
 binnen **PRINT** opdracht 55
 ESC toets uitgesteld effect van 59
 om directory te stoppen 190
Coördinaten 69–71, 74
 variabelen als 101–104
Copiëren van bestanden 193–194
Copiëren van diskettes 187
CSAVE instructie 97, 173–174,
 176, 207, 215
Cursor 44, 47–50, 216
- DATA opdracht** *zie* **READ** en **DATA**
 opdrachten
Deelprogramma's 125, 142–145, 209
Deelteken (/) 118
DELETE/BACK SPACE toets 50
DELETE instructie 194
Diapositief beeld *zie* Inverse-videotoets
DIM opdracht 122, 208, 214
Directory 190–191, 193, 216
Disk drive 12, 29, 32, 179–182
 installeren 30, 180–182
 meerdere 180, 197–198
 opstarten 182–186
Diskettes 32, 182–186
 copiëren 196–198
 indelen voor uw computer 191–192
 in drive stoppen 185–186
 lege 191–192
 omgang met 182–184
 programma laden
 van 39, 40, 196, 210–211

- programma opslaan
 op 96, 194–196, 211
- Disk Operation System *zie* DOS diskette
- Disk hulpprogramma's 188–190
- Dollar tekens (\$) voor
 tekstvariabelen 121–123
- DOS instructie 189, 196, 208
- DOS diskette 184, 188, 191
 kopiëren 192
 versies van 188
- DOS menu 196–198
- DRAWTO instructie 75–78, 208
- Driehoektekenen-
 programma 120–122, 123
- Dubbele punt 57, 115
- DUPLICATE DISK
 instructie 196–198
- END instructie 144, 208
- ENTER instructie
 177–178, 196, 208–209, 215
- ERROR (fout) meldingen
 51–53, 93–96, 213–216
 BOOT ERROR 186
- ESC toets 58–60, 67
- Exponenttekens (^) 118
- Extensie in bestandsnaam 193
- FOR/NEXT lussen
 130–137, 145, 146, 209, 214
- Formaties 201, 214
- Formules 118–120, 123
- Fouten opsporen 93–96, 99
- Foutmeldingen *zie* ERROR
- Functietoetsen 60–62
- Gegevenscontrole 141, 144
- Geheugen 22
 grafische stand 66, 156–158
 schoonmaken, NEW instructie
 voor 211
- Geheugenuitbreiding 22
- Getallen-raadprogramma 139–140
- GOSUB opdracht 142–146, 209, 215
- GOTO opdracht 125–130, 209
 in tegenstelling tot GOSUB
 opdracht 144–145
 IF opdrachten met 140–142
- Grafieken 65–79, 150–154
 FOR/NEXT lussen voor 136, 137
 vol scherm 155
- Grafische lei 200
- Grafische standen 65–66, 79, 156–161
 fijn raster 157, 158–159, 216
 speciaal (GTIA) 157, 159–160
 tekst 67, 154–155, 157, 159–160
 tweekleuren 158, 159
 veranderen 67, 209
 vierkleuren 66, 72–79, 158, 159
- Grafische tekens 48
- Grafisch scherm 72, 73
- GRAPHICS instructie 67, 72, 79, 209
- Haakjes 119–120
- HELP toets 60–61
- Helderheden 150, 151, 152, 153
- Hoofdletters *zie* Tekststanden
- IF opdracht 138–142, 146, 210
- Inhoudsopgave *zie* Directory
- INPUT opdracht
 113–117, 120, 123, 124, 210
- INSERT toets 50, 90
- Installeren 23, 25–27
 van disk drive 180–182
- Instructies 51–57, 79–80, 215
 samengestelde 58
- Inverse-videotoets 46, 53, 56, 163
- Invoegen van programmaregels 87–89
- Is-gelijk teken (=) 102, 123, 213
- Joystick 28, 34
- Kabels 25, 26, 27
- Keuzetoets *zie* Optietoets
- Kleine letters *zie* Tekststanden
- Kleur
 helderheden 150
 standaardinstelling 150–153, 213
 veranderen 153–154
- Kleurenregister 150, 152, 153
- Lichtpen 201
- Lijnen trekken 74–79

- LIST** instructie 86, 89
 voor cassette 177, 178, 210
 voor diskette 197, 210
 voor afdrukken 200
- LOAD** instructie 39, 210–211
 van cassette
 98, 171, 173–174, 210–211
 van diskette 196, 211
- Lussen** 130–135, 146, 209, 211
- Machinetaal** 202–203
- Massief figuurprogramma** 137
- Minteken** (–) 118
- Modem** 22
- Monitor in tegenstelling tot**
 televisie 27
- NEW** instructie 85, 91, 211, 213
- NEXT** opdracht *zie* FOR/NEXT lussen
- Oneindige lus** 127, 141, 146
- Opdrachtnummers** 85–91
- Opmaakinstructies** 47–51, 57, 63–64
- Opmaak van programmaregels** 88–89
- Opslaan**
 van getallen 101–105, 123–124
 van instructies 84–86
 van letters 121–123
 van programma's 84–90, 169
- Optelteken** (+) 118
- OPTION** toets 40, 60
- Pac-Man** 34
- PEEK** 202
- PILOT** programmeertaal 38, 202
- PLOT** instructie 74–79, 91–92,
 106, 211
- Plusteken** (+) 118
- POKE** 202
- POSITION** opdracht 68–71, 211
- POWER** schakelaar 26
- PRINT # 6** 162, 211
- PRINT** opdracht 53–57, 64,
 83–91, 211
INPUT opdracht met 115
 met speciale tekststand 162, 211
- Printer** 29, 199, 200
- Programma recorder** 28, 29
 installeren 28
 programma laden vanaf 171–172
 programma opslaan op 173–174,
 175, 176
- Programma's** 83–92
 fout zoeken in 93–96
 opstellen van 88–89
 stoppen van 127, 136–137
- Programmeregels toevoegen** 87–89
- Programmatuur** 32–33
 catalogus 206
 gebruiken van 38–39, 188
- Programmeertalen** 84, 202
- Puntkomma** 115, 127, 162
- Pijltoetsen** 48, 59
- Raadprogramma** *zie* Getallenraad-
 programma
- Randapparatuur** 27–30
 aansluiten van 28, 180, 200
- Raster** 66, 79, 156–157
- READ en DATA** opdrachten
 112–113, 122, 124, 207, 212
- Reeksen** 122
- Reeksvariabelen** 122
- Rekenkundige handelingen**
 101, 118–121
- REM** opdracht 93, 95
 in subprogramma's 143
- RENAME** instructie 194
- RESET** toets 53, 62, 85
- RETURN** instructie 47, 63, 215
- RETURN** toets 47, 51, 55, 56
- RUN** instructie 86, 87, 212
 met disketteprogramma 188, 196
- RUN CARTRIDGE** instructie 195
- SAVE** instructie 212
 met cassettes 97, 175–176, 178, 212
 met disk drive 196, 212
- Scherms** 43, 67–68
- Schrijfbescherming** 183, 184, 216
- Schuine streep (/)** als deeltteken 118
- SELECT** toets 61

- SETCOLOR instructie 148–154, 212
 in fijn raster grafische stand 159
 in GTIA standen 159–160
 in tekststand 153–154, 164, 165
 in vierkleuren-stand 165
- SHIFT toets 45, 50
- Sleutelwoord 54
- Software *zie* Programma's en
 Programmatuur
- Spraak synthesizer 22
- Standen *zie* Grafische standen,
 Tekststanden
- START toets 61
- STEP 132–133, 209
- Sterretje (*) 118
- String *zie* Reeks
- Stroomtoevoer *zie* Aansluiting
- Subprogramma's *zie* Deelprogramma's
- Systeem hulpprogramma's 192
- Tekststanden 65, 66–71, 161–165, 166
 hoofdletters 162–164, 165
 kleine letters 45, 56
 SETCOLOR registers in 152–153
 vierkleuren 157, 165–166
- Tekststream 70, 71
 verwijderen 154–155, 165, 209
- Tekstverwerking 39
- Telefoon modem 22
- Televisie *zie* Scherm
 aansluiten van computer aan 26
- Tellervariabele 131–133
- Toekenningsopdracht 101, 122
- Toetsenbord 43–47, 63–64
- Toevoegen van programmaregels
 87–89, 90
- Toevoegen in bestandsnaam *zie* Extensie
- Variabelen 101, 103
 namen voor 102–105
 tekst 121–123, 124, 213
- Vaste getallen 102, 105
- Vergelijkingen 138–142
- Vermenigvuldigingsteken (*) 118
- Verkwasten 73–74, 148–154
- Verplaatsen van programmaregels
 89–90, 91
- Verwijderen
 bestanden 195
 programmaregels 89, 91, 178
 tekens 49–50
- Vierhoektekenen programma
 91–92, 95, 129
- Vierkleuren grafische standen
 66, 70–79, 157, 158
 in programma vierhoektekenen
 91–92
 SETCOLOR instructies codes in
 150–154
- Vierkleuren tekststanden
 157, 162–163, 164–165
- Vuurknop 28
- Wachtlussen 136–137
- Wisborstel 73, 77–78, 152

**VRAAG EEN
 COMPLETE
 CATALOGUS
 AAN BIJ:**

Duitsland
 SYBEX-VERLAG GmbH
 Vogelsanger Weg 111
 4000 Düsseldorf 30
 Tel.: (02 11) 62 64 41
 Telex: 8 588 163



USA
 SYBEX INC
 2344 Sixth Street
 Berkeley, CA 94710, USA
 Tel.: (415) 848-8233
 Telex: 336 311

Engeland
 SYBEX Ltd.
 Unit 4, Bourne Industrial Estate
 Crayford, Kent DA14BU
 Tel.: Crayford (03 22) 5 77 17
 Telex: 896 939

Frankrijk
 SYBEX
 6-8, Impasse du Curé
 75018 Paris
 Tel.: 1/203-95-95
 Telex: 211.801f



ATARI
MUSEUM.NL

Praktische gids ATARI 600XL/800XL

Alles over de nieuwe ATARI XL Serie!

Met behulp van dit boek leert u in zeer korte tijd hoe u de Atari 600XL/800XL gebruikt. Beetje voor beetje raakt u vertrouwd met de installatie, het toetsenbord en het scherm.

Door de stap voor stap beschrijving van elk ATARI model wordt het voor u heel eenvoudig om uw eigen BASIC programma's te schrijven. U kunt ook het programmeren overslaan en direct leren hoe u met commerciële software aan de slag kunt gaan.

U leert in een handomdraai

- het gebruik van de nieuwe XL grafische mogelijkheden
- het aansluiten van uw disk drive en het gebruik ervan
- het op diskette opslaan en laden van programma's
- hoe u uw systeem met handige accessoires uit kunt breiden
- en nog heel veel meer!

Alles wat u als beginner moet weten wordt u op een prettige en begrijpelijke manier verteld.

Over de Auteur

Thomas Blackadar is een beroepsauteur. Hij werkt al meer dan tien jaar met computers. Hij haalde aan de Princeton University een B.A. graad en studeerde aan de Universiteit van Freiburg in Zwitserland.

Hij heeft nog verschillende andere computerboeken voor SYBEX geschreven die echter nog niet vertaald zijn: *The Best of TI 99/4A Cartridges*, *The Best of VIC-20 Software* and *The Easy Guide to Your Coleco Adam*.

ISBN 3-88745-109-0