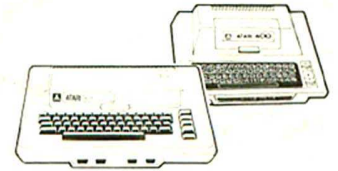


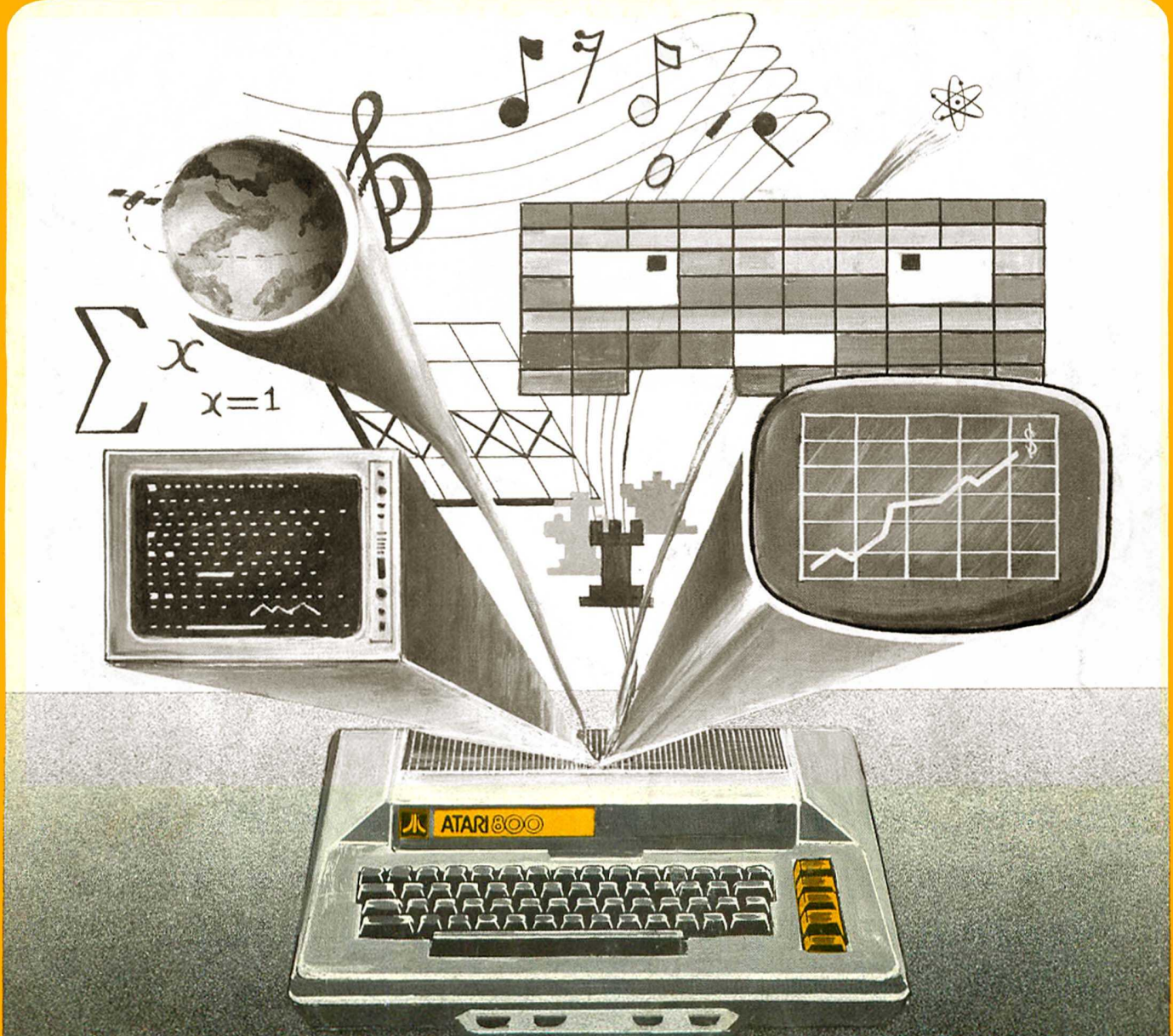
A.N.A.L.O.G. 400/800

ATARI NEWSLETTER AND LOTS OF GAMES

MAGAZINE



MÄRZ / APRIL 1981



**ATARI-Drucker Disk Files: NOTE & POINT verwenden
Überwache dein Scheckbuch Download Terminal!
Neue Produkte Gerüchte VCS Update**

HINWEIS: ATARI® ist eine registrierte Marke von ATARI, INC., ein Unternehmen von Warner Communications und alle Verweise auf ATARI® sollten so markiert sein.
(Firma existiert nicht mehr)

A.N.A.L.O.G. 400/800

ATARI NEWSLETTER AND LOTS OF GAMES MAGAZINE



MÄRZ/APRIL 1981

NUMMER 2

Redaktion

Herausgeber/Verleger
MICHAEL DES CHENES
LEE PAPPAS

Redaktionsassistent
AMY LYNDE

Sekretariat
SUSAN MESCHKE

Künstlerische Leitung
BOB DESI

Geschäftsführer
LEE PAPPAS

Anzeigenannahme
MICHAEL DES CHENES

Typographie
HENDRICKX & LARRIVEE

Herstellung/Distribution
COLONY PRESS, INC.

Technikredaktion
CHARLES BACHAND
TOM HUDSON
ROBERT HARTMAN

Freie
Mitarbeiter
JERRY WHITE
TOM REPSTAD
JOSEPH TREM
ED STEWART
BILL LATINO
DAVID BOHLKE
PETER PAPPAS
PAUL HOFFMAN
BILL WILKINSON

Das A.N.A.L.O.G. 400 / 800
Magazin ist in keiner Weise
geschäftlich verbunden mit
ATARI. ATARI ist ein Marke der
ATARI, Inc.

SCHWERPUNKTE / ARTIKEL

Disk Files: NOTE & POINT verwenden Jerry White ...12
Du verschwendest Arrays Charles Bachand ...14
Player/Missile Grafik Joseph Trem ...20
Entfessele die Leistung von ATARIs Hauptprozessor ... Ed Stewart ...22
Grafik-"Schnipp"-Routine Tom Hudson ...30
Maschinensprache in BASIC umwandeln Paul Hoffman ...45
Assembler/Editor: Eine Anti-Anleitung Charles Bachand ...46

BESPRECHUNGEN / PROFILE

Decision Maker Mike Des Chenes ...25
ATARI Energy Czar Mike Des Chenes ...35
Mosaic Nachrüstsatz 16 KB Tom Hudson ...36
CE Software: War At Sea Lee Pappas ...37
ATARI Calculator Peter Pappas ...40
VersaWare: Mind Boggler-1 Mike Des Chenes ...42
Die ATARI-Drucker 820, 822, 825 ANALOG ...43

PROGRAMM LISTINGS

Überwache dein Scheckbuch Charles Bachand ...17
Kanonenduell David Bohlke ...23
Music Decomposer Charles Bachand ...26
Download Terminal! Bob Hartman & Lee Pappas ...38

KOLUMNEN

Editorial Mike Des Chenes4
Leserkommentare ANALOG5
ATARI News ANALOG9
Neue Produkte ANALOG ...10
Gerüchte ANALOG ...15
LISTEN Lee Pappas ...26
VCS Update Mike Des Chenes ...28
Der Spielsalon Tom Repstad ...32
User Groups ANALOG ...37
Bugs & Bytes Mike Des Chenes ...48
Im Scheinwerferlicht ANALOG ...50

Inländisches Abonnement: 1 Jahr 6 Hefte (2-mtl.) \$12; 12 Hefte \$22; 18 Hefte \$32.
Aufträge für Abonnements oder Adressänderungen (Postformular 3575) an

A.N.A.L.O.G. 400/800 Magazine
P.O. Box 23
Worcester, MA 01603
Telefon 617-892-3488 (Adresse und Telefon nicht mehr in Betrieb)

Kein Beitrag aus diesem Magazin darf ohne die Genehmigung des Herausgebers in irgendeiner Form reproduziert werden. Programmlistings bitte in gedruckter Form zuschicken. Artikel bitte als getippten Text mit Groß- und Kleinschreibung und doppeltem Zeilenabstand ausfertigen. Durch Zuschicken des Artikels an das A.N.A.L.O.G. 400/800 Magazin erkennen die Autoren an, dass ihr Material, das für die Veröffentlichung akzeptiert wird, in das ausschließliche Eigentum von ANALOG übergehen. Werden sie nicht zur Veröffentlichung angenommen, bleiben die Texte Eigentum des Autors. Sollen Beiträge zurückgesandt werden, bitte adressierten Rückumschlag beifügen.

Gesamter Inhalt Copyright ©1981 A.N.A.L.O.G. Magazin
Deutsche Übersetzung ©2009 GBXL
Danke an Sleepy & des-or-mad für die Hilfen
3. überarbeitete Auflage 3/2012

UNLEASH THE POWER OF ATARI'S CPU

(Entfessele die Leistung von ATARIs Hauptprozessor)

von Ed Stewart

Würdest du gerne bis zu 30% mehr Tempo aus deiner ATARI-6502-CPU herauskitzeln? Möchtest du diesen Vorteil außerdem ohne zusätzliche Ausgaben erzielen? Sollte deine Antwort nein lauten, magst du wahrscheinlich auch keinen Apfelkuchen; aber falls deine Antwort ja ist, lies weiter, und ich werde dir aufzeigen, wie man ein solches Kunststück mit einem einfachen BASIC POKE an der richtigen Stelle vollbringt.

Zuerst ein wenig Hintergrundinformation über eines der vielen Dinge, die in eurem ATARI-Computer so vor sich gehen. Die Besonderheit, die ich euch näherbringen möchte, ist, wie Bildinformationen auf den Fernseher gelangen. Es gibt einen speziellen Chip namens ANTIC, der die Hauptverantwortung dafür trägt, dass das Bild auf euren Fernseher erscheint. Der ANTIC erledigt das unabhängig vom Hauptprozessor 6502 in eurem Computer. Der ANTIC ist in Wahrheit eine einfache CPU mit eigenen Rechten. Er führt ein Programm aus, das im RAM liegt, genau so, wie die 6502-CPU ein Programm ausführt, das im RAM oder ROM liegt. Wir können deshalb den ATARI als simultan-verarbeitenden Computer bezeichnen, da mehr als eine CPU jederzeit aktiv sein kann. Ein seltsamer und irgendwie unglücklicher Vorgang läuft ab, wenn ein simultan-verarbeitendes System, wie der ATARI eines ist, aktiv Anweisungen ausführt – beide CPUs verlangen zeitgleich Zugriff auf den Speicher. Die beiden CPUs können aber nicht zur gleichen Zeit auf den Speicher zugreifen, also muss eine warten, bis die andere ihre Anforderung auf Zugriff abgearbeitet hat. Dieser Konflikt beim Speicherzugriff ist allen Computern mit mehr als einer CPU gemeinsam – von Mikro- bis zu Makrocomputern – und ist im Allgemeinen nichts, worüber man sich Gedanken machen müsste.

Der ANTIC holt seine Daten aus dem Speicher mittels einer Technik namens "Direct Memory Access" (direkter Speicherzugriff) oder DMA. Wann immer dieser Speicherzugriff erfolgt, wird die 6502-CPU vorübergehend angehalten. Dem DMA wird nachgesagt, er "stehle" dem Computer einen Teil seiner verfügbaren Zeit, genannt Taktzyklus. Es sind genau 1.789.790 Taktzyklen Computerzeit pro Sekunde verfügbar (NTSC-System). Wenn der DMA nicht einige Taktzyklen Rechenzeit "gestohlen" hätte, wäre der 6502 nicht angehalten worden und hätte seine Programmanweisungen eher fertig abgearbeitet. Die logische Folgerung daraus ist, dass je mehr DMA-Aktivitäten durch den ANTIC-Chip auftreten, desto mehr wird unser 6502 verlangsamt.

Der ANTIC-Chip baut das komplette Fernsehbild 60-mal pro Sekunde auf (NTSC-System). Während jeder dieser 60 Wiederholungen werden dem 6502 viele Taktzyklen gestohlen. Während jeder dieser 60 Wiederholungen unterbricht (interrupt) der ANTIC-Chip außerdem den 6502 und lässt ihn Aufgaben abarbeiten wie das Aktualisieren verschiedener Softwarezähler (timer) und Auslesen der Spielsteuerungen (Joysticks und Paddles). Wenn der

6502 erledigt hat, was er auf Anforderung des ANTIC-Interrupts tun muss, darf er fortfahren mit dem, womit er beschäftigt war, bevor er vom ANTIC abgelenkt wurde. Es sollte jetzt klar sein, dass, obwohl der ANTIC unverzichtbar ist, er die 6502-CPU ausbremst. Aber um wie viel?

Ich schrieb ein einfaches BASIC-Programm für meinen ATARI 800, um diese Frage beantworten zu können. Eine FOR-NEXT-Schleife wird 100.000-mal ohne Einmischung durch andere Anweisungen wie folgt ausgeführt:

```
20 FOR I=1 TO 100000:NEXT I
```

Zuerst galt es zu messen, wie lange diese Schleife zur Ausführung braucht, wenn kein ANTIC-DMA aktiv ist. Ein POKE 559,0 schaltete den DMA aus und der Bildschirm wurde schwarz. Ein POKE 559,34 schaltete den DMA wieder ein und das vorherige Bild wurde wieder hergestellt. Die FOR-NEXT-Schleife wurde in den Grafikmodi 0-8 mit aktivem DMA ausgeführt, und die in Tabelle 1 gezeigten Zeiten wurden ermittelt. Die Laufzeiten verlängerten sich mit DMA von nur 10% in Grafikstufe 3 bis zu reichlichen 47% in Grafikstufe 8. Es ist leicht zu erkennen, dass man, wenn man umfangreiche Berechnungen vornimmt und den Bildschirm oder die Softwaretimer sowie Spielsteuerungen nicht braucht, den ANTIC DMA dann eine Weile abschalten kann und seine Ergebnisse eher erhält. Außerdem wird aus der Tabelle weiter unten deutlich, dass Programme schneller ausgeführt werden, wenn sie unter Grafikstufe 3, 4 oder 5 laufen.

Ich hoffe ihr habt ein wenig mehr über die ATARI-Computer gelernt und darüber, wie der ANTIC-DMA die 6502-CPU beeinflusst. Man könnte es irgendwann z.B. möglich machen, diese verborgene Leistung während eines Computerschachturniers zu entfesseln und wenn dann jemand fragt, wie in aller Welt man das gemacht hat, kann man lächeln und sagen – ich und mein DMA.

GRAFIKMODUS	LAUFZEIT SEKUNDEN	% STEIGERUNG (mit DMA)
KEIN DMA	148	
GRAPHICS 0	216	46
GRAPHICS 1	188	27
GRAPHICS 2	186	26
GRAPHICS 3	163	10
GRAPHICS 4	164	11
GRAPHICS 5	167	13
GRAPHICS 6	173	17
GRAPHICS 7	185	25
GRAPHICS 8	218	47

Assembler/Editor.....

EINE ANTI-ANLEITUNG

von Charles Bachand

Einer der Herausgeber rief mich eines Tages ziemlich unerwartet an. "Hi, hier ist Lee", sagte er. "Mir kam gerade eine großartige Idee über einen Artikel, den du für die nächste Ausgabe schreiben könntest. Wie wär's mit einer Anleitung zu dem neuen Assembler/Editor-Modul?" Nun, ich hasse es sogar Anleitungskartikel nur zu lesen, sollen den doch andere schreiben, aber ich hörte trotzdem weiter zu. "Oh, und übrigens", sagte er, als er seine Falle auslegte, "ich leihe dir meine Videokassette mit FLASH GORDON!" Das überzeugte mich. Ich musste irgendwie mit einem Anleitungskartikel herkommen. Obwohl ich Anleitungen verabscheue, wurde mir aufgezungen, eine zur Veröffentlichung vorzulegen. Letzten Endes, wie lange konnte es schon dauern, eine kleine Anleitung zu schreiben? Es sollten nicht mehr als fünf oder sechs Blatt Papier mit doppeltem Zeilenabstand werden, vielleicht eine Woche bis zehn Tage höchstens. "Bis wann soll er fertig sein?", fragte ich, nicht ahnend, welche Antwort folgen würde. "Ich brauche den Artikel bis Freitag!", war seine Antwort. Er konnte doch nicht diesen Freitag meinen? Bis Freitag waren es nur drei Tage. "Das ist dieses Freitag!", wiederholte er. Meine schlimmsten Befürchtungen über Lees Geisteszustand wurden bestätigt. Er hatte seinen Verstand verloren.

Da ich "FLASH GORDON" verpasst hatte, als er im Kino lief, war im Zweifelsfall Lee den Vorzug zu geben das Beste. Im Grunde genommen hat jemand, der eine Anleitung schreibt, drei Möglichkeiten. Er kann eine Anleitung schreiben oder er kann über etwas anderes schreiben und es eine Anleitung nennen (in der Hoffnung, dass jemand Bestimmtes in seinem verrückten Zustand es nicht merkt), oder er kann ein wenig von beidem tun so wie ich. Ich will mich einfach nicht selbst langweilen, wenn ich es später durchlesen muss, Rechtschreibkontrolle usw.

An dieser Stelle will ich Mal ein paar Worte einfließen lassen über das Schreiben von Artikeln für das A.N.A.L.O.G.-Magazin. Wir sind darauf angewiesen, dass ihr eure Wünsche und Ideen dem Rest von uns ATARI-Besitzern mitteilt. Viele klagen häufig darüber, dass sie mit ihrem Computer alleine gelassen sind. Das ist wirklich eure eigene Schuld. So ist zum Beispiel die Hälfte allen Spaßes am Schreiben eines "Spiel"-Programms das Wissen, dass andere die Freude an deiner Arbeit mit dir teilen. Und wenn es in einem Magazin abgedruckt wird, können Tausende ebenfalls an dem Spaß teilhaben. Und ich habe einen sehr eigennützligen Grund, aus dem ich euch alles das erzähle. Je mehr die Leute Artikel schreiben und Programme einreichen, desto weniger habe ich zu tun und kann mich darauf konzentrieren, duftige Spielprogramme zu schreiben.

Das Assembler/Editor-Modul, das erst vor wenigen Monaten herauskam, ist ein dringend notwendiger Helfer, nützlich für jeden, der ernsthaft daran interessiert ist, kompakte und effiziente Programme zu schreiben. Es hat lange bis zur Veröffentlichung gedauert, da das Modul schon fast ein Jahr vor der Dokumentation fertiggestellt war. ATARI entschied zu warten, bis das Handbuch

fertig war, bevor das ROM herausgegeben wurde. Die Leute, die dieses Programm entwickelt haben, schafften es eine Menge Funktionen in die 8 KB Speicherplatz zu quetschen, mit denen sie auskommen mussten. Es gibt im Assembler/Editor drei verschiedene Bereiche.

Der Editor ist eine erweiterte Version des Bildschirmeditors, der im ATARI eingebaut ist. Neben den üblichen Editorfunktionen wie Cursorpositionierung sowie Zeilen einfügen und löschen verfügt er außerdem über Kommandos zum neu Nummerieren (die Zeilennummerierung nimmt in einem festgelegten Intervall zu); waren also die ursprünglichen Nummern 10, 12, 13, 22, würde die Eingabe von REN Zeilennummern mit der Folge 10, 20, 30, 40 ergeben. Neben dem Löschen von Zeilen durch Eingabe der betreffenden Zeilennummer kann man nun eine Reihe von Zeilen im Programm löschen, indem man DEL XX,YY eingibt. Will man in einem Programm alle Zeilen von 300 bis 450 löschen, muss man nur DEL 300,450 eintippen.

Hat es dich schon immer gestört, die Zeilennummern des Programms eingeben zu müssen, besonders wenn sie als Zahlenfolge festgelegt sind? Hast du dir schon immer gewünscht, dass die Maschine das für dich erledigt? Jetzt könnt ihr eure Tröten und Partyhüte auspacken, der Assembler/Editor macht das alles für euch! Nachdem man den Befehl NUM eingegeben hat, erzeugt der Editor nach jedem RETURN eine Zeilennummer. Man gibt die Programmanweisungen ein, der Computer übernimmt den Rest. Ach, ich wünschte mir, dass einige dieser Funktionen in ATARI BASIC eingefügt worden wären.

Wir kommen jetzt zu den zwei String-Kommandos, die jemand umsichtigerweise in den Editor eingebaut hat. Das Erste ist FIND. Dieses Kommando braucht man, um einen bestimmten String (Zeichenfolge) in einem Programm aufzufinden. Lasst uns zum Beispiel annehmen, dass wir den String "START" finden wollen. Gib einfach 'FIND/START,' ein und das erste Vorkommen dieses Strings wird angezeigt. Sehr wahrscheinlich wollt ihr alle Stellen angezeigt bekommen, an denen der gesuchte String vorkommt. Hängt einfach ein 'A' an das Kommando an, sodass es 'FIND/START,/A' lautet und die Maschine gibt alle Zeilen, die den String "START" beinhalten, auf dem Bildschirm aus. Man kann sogar die Zeilen begrenzen, liste alle Zeilen von Zeile X bis Zeile Y auf, die den String enthalten. Bisher habe ich das Zeichen "/" als Trennzeichen benutzt. Tatsächlich kann man jedes Zeichen mit Ausnahme von Leerzeichen, Tabulator und Return nehmen. Und sollte der gesuchte String das Zeichen "/" enthalten, nimmt man ein anderes. 'FIND./2.,A' ist zulässig. Hier wurde "." anstelle von "/" verwendet, da der String ein "/" beinhalten würde. 'FIND?/2?,A' ist auch zulässig. Das ist auf jeden Fall besser, als ein 500 Zeilen langes Programm nach ein paar Schreibfehlern zu durchsuchen.

Was nun, wenn man nach dem Auffinden aller Stellen mit dem String "/2" merkt, dass man einen Fehler gemacht hat. Es sollten stattdessen alle "/4" sein. Man kann entweder jede Zeile durchgehen, die den String

enthält, oder den Computer das für uns erledigen lassen. Das bringt uns zu der letzten Editorfunktion, die ich für erwähnenswert halte. Das ist das Kommando REP. Um alle Vorkommen des Strings "/2" durch "/4" zu ersetzen, müssen wir bloß 'REP./2./4.,A' eingeben und ruckzuck ist es passiert. Diesmal habe ich den Punkt "." als Trennzeichen verwendet, da er ja nicht als Zeichen innerhalb des Strings vorkommt. Wird am Ende "/,A" weggelassen, würde nur das erste Vorkommen von "/2" geändert. Man kann sogar einen Bereich an Zeilennummern angeben, innerhalb dessen ersetzt werden soll, wie beim Kommando "FIND". Was aber, wenn man nicht alle Vorkommen ersetzen will? Hängt man am Ende des Kommandos ein ",Q" statt eines ",A" an, listet der Computer die Zeile mit dem String auf und fragt dich per "?" ab. Gibt man ein "Y" (=ja) gefolgt von einem Return ein, wird der String ersetzt. Wird nur Return gedrückt, bleibt der String unverändert.

Für I/O-Operationen (Input/Output = Eingabe/Ausgabe) mit dem Editor hat man drei Anweisungen zur Verfügung. Laden eines Programms in den Computer wird mit dem Kommando "ENTER" erreicht. Man tippt das Wort ENTER ein gefolgt von dem Zeichen "#" und das Gerät, von dem geladen werden soll. Einladen eines Programms vom Kassettenrekorder erreicht man mit "ENTER#C:". Für das Laden eines Diskettenfiles nimmt man "ENTER#D:YIPPIE".

Das Speichern eines Programms (Quelltext) erledigt man mit einer Variante des Kommandos "LIST". Normalerweise wird das LIST-Kommando dazu benutzt, ein Programm auf dem Bildschirm (LIST) oder einem Drucker (LIST#P:) auszugeben. Um ein Programm auf Kassette zu speichern, gibt man "LIST#C:" ein; zum Speichern auf Diskette "LIST#D:YIPPIE". Das Modul verfügt auch über die Kommandos "LOAD" und "SAVE", die aber ausschließlich zum Laden und Speichern von Binärfiles verwendet werden können.

Eine Variation von LIST ist PRINT. PRINT macht das Gleiche wie LIST, nur dass es die Zeilennummern beim Verarbeiten herausfiltert. Für dieses Kommando habe ich bisher kaum Verwendungsmöglichkeiten gefunden mit Ausnahme einer Gelegenheit. Einmal war ich dabei, ein Datenfile mit Strings zu erzeugen (ja Leute, das Modul prüft nicht auf Syntaxfehler wie BASIC, da es die Programme nicht in einer zu Tokens umgewandelten Form speichert). Ich hatte die Zeilen mit Zeilennummern eingegeben, damit der Editor sie nicht als Befehle annimmt. Nachdem das Editieren beendet war, wurden die Zeilennummern nicht mehr gebraucht, also wollte ich ein BASIC-Programm schreiben, um sie einzulesen und wieder zurückzuschreiben, aber ohne die Zeilennummern. Lee indes hatte eine großartige Idee. "Warum gibst du die Daten nicht mit PRINT auf Diskette aus, dann brauchst du kein Konvertierungsprogramm zu schreiben", sagte er. Wisst ihr, es könnte noch Hoffnung für ihn bestehen.

Nun, wie setzt man diese zusätzlichen Editorfunktionen für das Editieren von BASIC-Programmen ein. Ihr lest richtig! Man könnte mehrere Zeilen auf einmal löschen, Zeichenfolgen ersetzen und automatisch Zeilennummern erzeugen. Wie mache ich das mit BASIC-Programmen, fragt du sicherlich? Zuerst fährt man den Computer mit BASIC hoch und lädt das Programm, danach listet man es auf das Speichergerät, egal ob Kassettenrekorder "C:" oder Diskettenlaufwerk "D:YIPPIE". Dann ersetzt man das BASIC-Modul durch das Assembler/Editor-Modul. Sobald ENTER vom Kassettenrekorder oder Dis-

kettenlaufwerk ausgeführt wurde, voila, kann man nun mit einem viel besseren Editor am BASIC-Programm arbeiten. Der umgekehrte Weg: wir listen das Programm an das Speichergerät, nehmen wieder BASIC, führen in BASIC ein ENTER vom Speichergerät aus. Nun braucht man die Programme nur noch mit SAVE zu speichern, fertig.

In dieser Ausgabe habe ich nur die Funktionen des Editors vorgestellt. In der nächsten Ausgabe wird der Assembler besprochen, der es ermöglicht, Programme in Maschinensprache zu erzeugen. Und der Debugger, welcher es ermöglicht, Speicherstellen einzusehen, die Programmausführung zu überwachen, Maschinencode zu disassemblieren und vieles mehr. Das Modul lohnt sich wirklich, und wie gesagt, ließ lange auf sich warten.

So, du glaubst also, dass ich die Artikel nicht lese, bevor sie in Charlie wandern??? Für die nächste Ausgabe hast du besser die Anleitung zur Assemblersprache fertig oder es gibt keine geliehenen Science-Fiction-Videos mehr! - Lee

MOVIE THEMES 10 SELECTIONS

SUPERMAN • STAR WARS • 2001
JAMES BOND • CE3K • STING
GALACTICA and more



8K CASSETTE only \$12.00

FOR USE WITH ATARI MUSIC COMPOSER

A.N.A.L.O.G 400/800
PO. BOX 23
WORCESTER, MA 01603

ATARI® is a Registered Trademark of Atari, Inc.