ATARI®

ATARI PROGRAMM

DXG 5726
Diskette

PROGRAMM UTILITIES II

c 1983 Jegliche Rechte vorbehalten
ATARI ELEKTRONIK - Vertriebsges. mbH

ATARINSIDE

ATARI PROGRAMM

DXG 5726
Diskette

PROGRAMM UTILITIES II

c 1983 Jegliche Rechte vorbehalten
ATARI ELEKTRONIK - Vertriebsges. mbH

ATARINSIDE

INHALTSVERZEICHNIS

1. EINLEITUNG 1

1.1 Programmbeschreibung 1
1.2 Hardware 1

2. PROGRAMMSTART 1

3. INSOMNIA 2

3.1 Programmbeschreibung 2
3.2 Programmstart 2
3.3 Bedienungsanleitung 2

3.3.1 Allgemeines 2
3.3.2 Bildschirmbeschreibung 3
3.3.3 Editierbefehle 4

3.3.3.1 Soundlänge vergrößern (A) 5
3.3.3.2 Soundlänge verkürzen (D) 5
3.3.3.3 Sound laden (LS) 5
3.3.3.4 Sound speichern (SS) 5
3.3.3.5 Funktion des Steuerknüppels 6
3.3.3.6 Funktion des Drehreglers 6
3.3.3.7 Stimmauswahl (1,2,3,4) 6
3.3.3.8 Stimmauswahl löschen (C) 7
3.3.3.9 Stimme spielen (P) 7
3.3.3.10 Stimme triggern (T) 7
3.3.3.11 Stimme editieren (E) 7
3.3.3.12 Stimmen re-synchronisieren (R) 8
3.3.3.13 Tiefer Tonbereich (LO) 8
3.3.3.14 Hoher Tonbereich (HI) 8
3.3.3.15 Anzeigebereich ändern (<,>) 8

3.4 Anwendung in BASIC-Programmen 9

3.5 warnungen 11

3.6 Technische Informationen 11

4. INSTEDIT 13

4.1 Programmbeschreibung 13
4.2 Programmstart 13
4.3 Bedienungsanleitung 14

4.3.1 Editieren 14
4.3.2 Zeichensatz abspeichern 16
4.3.3 Zeichensatz laden 17
4.3.4 Zeichensatz schreiben 17

ATARINSIDE

4.3.4.1 BASIC-Unterprogramm 17
4.3.4.2 DATA-Zeilen 18
4.3.4.3 BYTE-Anweisungen 19

4.3.5 Testmodus 19

4.4 Testbeispiele 20

5. DSEMBLER 21

5.1 Programmbeschreibung 21
5.2 Programmstart 21
5.3 Bedienungsanleitung 22

5.3.1 Dateneingabe und Programmunterbrechung 22
5.3.2 Hauptauswahl 22

5.3.2.1 Menü 22
5.3.2.2 Byte-Suche 23
5.3.2.3 Disassemblieren 24
5.3.2.4 Hexcode laden 25
5.3.2.5 JSR (Sprung) 25
5.3.2.6 Label erzeugen 26
5.3.2.7 Label löschen 26
5.3.2.8 Label speichern und abrufen 26
5.3.2.9 Label Tabelle 27
5.3.2.10 Speicherseiten 27
5.3.2.11 Voreinstellungen 28
5.3.2.12 Zahlenumwandlung 29

6. AUTOSTART 30

6.1 Einführung 30
6.2 Gebrauch von AUTOSTART 30
6.3 Automatisches Programm-laden 32
6.4 Experimentieren Sie mit AUTOSTART 32
6.5 Musteranwendung 32

1. EINLEITUNG

1.1 PROGRAMMBESCHREIBUNG

Auf dieser Diskette der ATARI Programmierhilfen finden sich wiederum vier Programme, mit denen sich nützliche Unterprogramme (auch Maschinensprache) erstellen lassen. Die so entstandenen Unterprogramme lassen sich leicht mit Ihren vorhandenen BASIC-Programmen verbinden.

INSOMNIA ist ein Soundeditor mit dem sich komplizierte Klänge entwickeln lassen, die dann vom BASIC aus mit wenigen Befehlen aufgerufen werden können.

Ein komfortabler Zeichengenerator ist das Programm INSTEDIT. Hier wird die Eigenschaft des ATARI-Computers ausgenutzt, den internen Zeichensatz durch einen Eigenen zu ersetzen.

Mit dem Programm DSEMBLER können Sie auf einfache Weise vorhandene Maschinenprogramme untersuchen.

Nach dem Einschalten des Computers lädt sich ein von Ihnen erstelltes BASIC-Programm automatisch in den Programmspeicher und startet von selbst, wenn Sie vorher das Programm AUTOSTART auf die Diskette geschrieben haben.

1.2 HARDWARE

Es werden folgende Einheiten benötigt:
- 32K RAM
- ATARI Diskettenstation
- wahlweise ein ATARI Drucker
  (nur für das Programm DSEMBLER)

2. PROGRAMMSTART

*** Für ATARI Privatcomputer

Wenn der ATARI Computer kein eingebautes BASIC hat, wird das BASIC-Steckmoudl in den Modulschacht eingebaut.

- Speicherkapazität mindestens 32K RAM -

1. Schalten Sie Ihren Computer aus.

2. Schalten Sie die Diskettenstation ein.

3. Legen Sie die Programm-Diskette in das Laufwerk ein und schließen Sie den Disketten-Schacht.

4. Die einzelnen Programme dieser Diskette lassen sich nun mit der SELECT-Taste auswählen. Durch Drücken der START-Taste wird das gerade ausgewählte Programm in den Speicherbereich geladen und automatisch gestartet.

3. INSOMNIA (Toneditor)

3.1 PROGRAMMBESCHREIBUNG

INSOMNIA ist ein Programm, mit dem Sie komplizierte Klänge von maximal 4 Sekunden Länge erzeugen können. Es stehen 4 Stimmen zur Verfügung, die unabhängig voneinander gesteuert werden können. Die erzeugten Klänge können in Ihr BASIC Programm integriert werden. Ein mitgeliefertes Maschinen-Unterprogramm ermöglicht es, daß Ihr Programm unabhängig von diesem Soundprogramm abläuft. Die Klänge werden durch drei einstellbare Parameter erzeugt:

Frequenz, Klangfarbe und Lautstärke.

Sie können mit Hilfe eines Steuerknüppels oder Drehreglers gezielt verändert werden. Die Werte werden auf dem Bildschirm graphisch dargestellt. Bereits während des Editiervorganges können die vier Stimmen abgehört werden, wahlweise einzeln oder gemeinsam, so daß eine genaue Einstellung möglich ist. Weiterhin stehen zwei Frequenzbereiche zur Verfügung, die die vom BASIC bekannten Grenzen überschreiten. Die erzeugten Klänge können abgespeichert und für weitere Bearbeitungen wieder in den Speicherbereich geladen werden.

3.2 PROGRAMMSTART

Nachdem Sie das Programm INSOMNIA vom Hauptmenü gewählt haben, wird das Programm automatisch gestartet.

3.3 BEDIENUNGSANLEITUNG

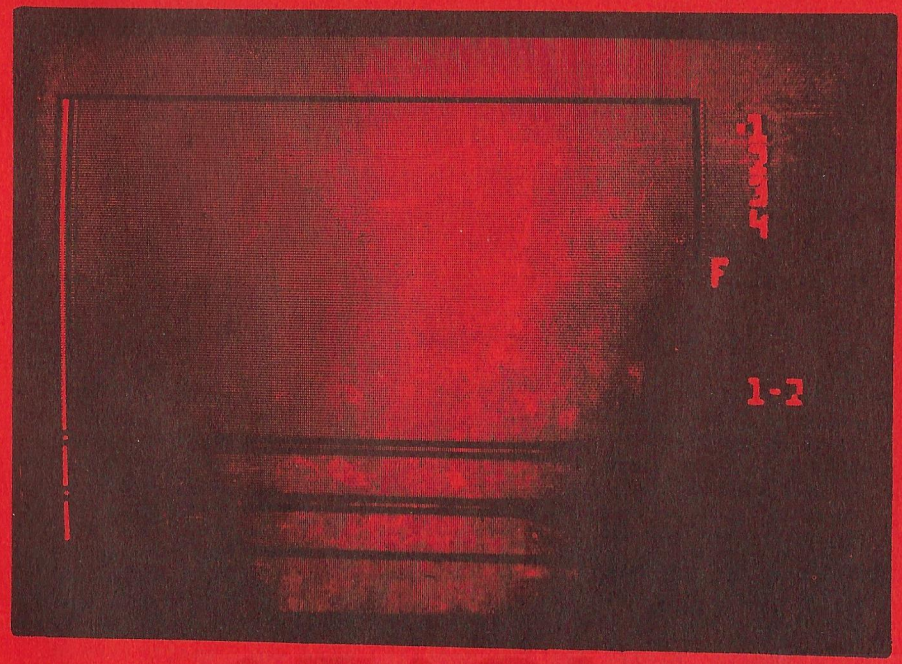
3.3.1 ALLGEMEINES

Das Ziel dieses Programmes besteht darin, Sound-Effekte zu erzeugen und diese ins BASIC zu integrieren. Für die Klangerzeugung steht Ihnen ein Editor zur Verfügung, der wesentliche Vorteile gegenüber der üblichen Methode, Klänge zu erzeugen, besitzt (in BASIC: FOR-Schleifen, SOUND-Befehle). Der entscheidende Vorteil besteht darin, daß das Unterprogramm, welches mit diesem Editor erzeugt und in Ihr BASIC-Programm integriert wird, während der Ausführung keine Unterstützung vom BASIC Programm benötigt und die Geschwindigkeit des BASIC Programms scheinbar nicht verlangsamt. Während bislang ein enormer Zeitaufwand für Sounds in Programmen (wegen FOR-Schleifen, SOUND- und DATA-Befehlen etc.) notwendig war, kann jetzt gleichzeitig ein Programm abgearbeitet werden und im Hintergrund ein Vogel zwitschern. Der Verlust an Rechenzeit ist minimal. Der Sound kann nun mit wenigen POKE-Befehlen gesteuert werden, d.h. daß andere Vorgänge bearbeitet werden können, während gleichzeitig Töne erklingen. Dieser Vorteil wird nur mit wenigen Beschränkungen "erkauft".

Bevor Sie mit dem Programm arbeiten, sollten Sie die folgenden Kapitel durchlesen, um einen besseren Überblick zu erhalten. Fertigen Sie dann mit dem DOS-Befehl DUPLICATE DISK eine Kopie der Programmdiskette an und arbeiten Sie mit der Kopie weiter. Gehen Sie danach alle Editierbefehle durch, um die Möglichkeiten des Programms kennenzulernen. Erwarten Sie bitte nicht, daß Sie bereits nach einer Sitzung schnell und problemlos die Sounds erzeugen können, die Ihren Vorstellungen entsprechen. Erst mit wachsenden Erfahrungen gelingt es, sofort den gewünschten Effekt zu erzielen.

3.3.2 BILDSCHIRMBESCHREIBUNG

Kurz nach dem Programmstart erscheinen auf dem Bildschirm die Arbeitsfelder, wie in Abb. 1 dargestellt. In den drei Arbeitsfeldern werden die Parameter Frequenz (F), Lautstärke (L) und Klangfarbe (D=Distortion) dargestellt. Sie entsprechen damit dem SOUND-Befehl im BASIC. Durch Ausnutzung der vollen Breite werden also 120 SOUND-Befehle repräsentiert.



ATARIANIE

Die Symbole in der Abbildung bedeuten im Einzelnen:

a) Diese beiden senkrechten Linien bilden den Cursor. Die sich zwischen den Linien befindenden Punkte entsprechen den Sounddaten, die gerade bearbeitet werden. Der Cursor wird mit dem Steuerknüppel oder dem roten Knopf des Drehreglers bewegt.

b) Diese grüne Linie markiert das Ende der Sounddaten und bestimmt somit die Spieldauer eines Sounds. Sie kann mit den Befehlen "A" und "D" verschoben werden.

c) Der erleuchtete Buchstabe zeigt an, welcher der drei Parameter bearbeitet wird. Der Parameter wird durch den roten Knopf am Steuerknüppel oder durch die SELECT-Taste gewechselt.

d) Dieser Punkt zeigt an, welche der vier Stimmen in den Arbeitsfeldern gerade dargestellt wird. Diese Funktion wird mit dem Befehl "E" gesteuert.

e) Die Ziffern stehen für die vier Stimmen. Leuchten eine oder mehrere Ziffern, so erklingen bei einem "P" oder "T" Befehl genau diese Stimmen.

f) Diese Pfeile weisen auf die Stimmen hin, für die der nächste Befehl gilt. Sie können direkt durch Eingabe der Ziffern 1-4 angesteuert werden.

g) Diese Punkte entsprechen der Frequenz. Je höher der Punkt, desto höher die Frequenz des Tones.

h) Diese Punkte entsprechen der Lautstärke. Je höher der Punkt desto lauter der Ton.

i) Diese Punkte entsprechen der Klangfarbe. Der Bereich reicht von 0 (unten) bis 14 (oben).

j) Die Arbeitsfelder weisen eine Breite von 120 Punkten auf. Das entspricht einer Zeit von 2,4 Sekunden.

k) In den Arbeitsfeldern kann nur ein Teilbereich der gesamten Tondauer (ca. 4 Sekunden) gezeigt werden. Die Zahlen zeigen den jeweils dargestellten Bereich an. (1-2 = erste und zweite Sekunde, 2-3 = zweite und dritte Sekunde, 3-4 = dritte und vierte Sekunde). Der Wechsel erfolgt mit den Befehlen "<" und ">".

3.3.3 EDITIERBEFEHLE

Die folgenden Abschnitte erklären die Funktion der Befehle. Durch Eingabe der Zahl oder des Buchstabens (in Klammern hinter der Überschrift) wird der beschriebene Befehl ausgeführt.

3.3.3.1 SOUNDLÄNGE VERGRÖSSERN (A)

Um überhaupt sinnvoll arbeiten zu können, muß zuerst mit diesem Befehl die Zeitdauer eingestellt werden, für die der Sound erklingen soll. Jedes Betätigen dieser Taste verschiebt die Markierungslinie um einen Punkt. Jeder Punkt entspricht einer Zeit von 1/50 Sekunde. Da 120 Punkte gleichzeitig dargestellt werden können, steht ein Bereich von 2,4 Sekunden zur Verfügung.

3.3.3.2 SOUNDLÄNGE VERKÜRZEN (D)

Dieser Befehl verkürzt beim Betätigen der D-Taste die Soundlänge um 1/50 Sekunde. Die Markierungslinie verschiebt sich dabei wieder nach links.

3.3.3.3 SOUND LADEN (LS)

Mit diesem Befehl können Daten von Diskette gelesen werden, die mit dem Befehl "Sound speichern" (SS) abgespeichert wurden. Der Sound kann nun, wie beschrieben, weiter bearbeitet werden.

Geben Sie nacheinander die Buchstaben "L" und "S" ein. Sofort verschwinden die Arbeitsfelder und es erscheint die Frage nach dem Dateinamen, unter dem die Daten abgespeichert wurden. Auf der Programmdiskette befindet sich ein Demoprogramm, das die Geräusche einer Dampfmaschine simuliert. Geben Sie daher als Antwort D:DEMO.SND ein und beenden Sie die Eingabe mit RETURN. In den Arbeitsfeldern erscheint der Frequenzverlauf, die Lautstärke (konstant 8), sowie die Klangfarbe (Einstellung 0). Um die Geschwindigkeit der Maschine zu erhöhen, tippen Sie mehrmals ein "D" ein.
Falls I/O-Fehler auftreten, wird das Programm nach ca. 5 Sekunden ohne Datenverlust fortgesetzt.

3.3.3.4 SOUND SPEICHERN (SS)

Um die einmal erzeugten Klänge zu sichern, geben Sie zweimal ein "S" ein. Wie schon beim "LS"-Befehl verschwinden die Arbeitsfelder, und es wird nach dem Dateinamen gefragt, unter dem die Daten abgelegt werden sollen. Verwenden Sie als Endung z.B. "SND", um die Datei später einfacher identifizieren zu können.

Es werden alle vier Stimmen in der maximalen Länge von 4 Sekunden (genauer 4,8 Sekunden) abgespeichert, unabhängig davon, ob eine Stimme überhaupt benutzt oder in welcher Länge sie benutzt wurde.

Falls I/O-Fehler auftreten, wird das Programm nach ca. 5 Sekunden ohne Datenverlust fortgesetzt.

3.3.3.5 FUNKTION DES STEUERKNÜPPELS

Durch Drücken des Steuerknüppels nach links oder rechts wird der Cursor in die entsprechende Richtung bewegt. Ein Drücken auf den roten Knopf bewirkt einen Wechsel des Parameters, angezeigt durch ein Aufleuchten des entsprechenden Buchstabens. Wenn also das "F" aufleuchtet, kann im oberen Arbeitsfeld die Frequenz eines Tones geändert werden. Diese Einstellungen erfolgen, indem der Steuerknüppel vorgeschoben oder zurückgezogen wird. Der Punkt zwischen den beiden senkrechten Linien (dem Cursor) wird dadurch nach oben oder unten verschoben. Aufgrund dieser Bewegungen können die Parameter exakt eingestellt werden. Berücksichtigen Sie, daß ein Punkt lediglich einer Zeit von 1/50 Sekunden entspricht. Um z.B. einen Dauerton von einer Viertelsekunde zu erzielen, müßten 12 nebeneinanderliegende Punkte dieselben Parameter aufweisen.

3.3.3.6 FUNKTION DES DREHREGLERS

Der Drehregler ermöglicht in vielen Fällen eine schnellere Einstellung der Werte. Die Einstellung des Parameters erfolgt nun mit der SELECT-Taste. Die jeweiligen Werte für Frequenz, Lautstärke und Klangfarbe werden durch die Einstellung des Drehreglers bestimmt. Der rote Knopf am Drehregler kontrolliert die Position des Cursors. Ein Drücken bewirkt eine Positionsänderung um einen Punkt nach rechts. Diese Methode erscheint anfangs zwar kompliziert, ist jedoch bei häufiger praktischer Anwendung effektiver.

Um beispielsweise mehrere Punkte auf eine einheitliche Lautstärke zu setzen, ist folgende Methode angebracht: Stellen Sie den Drehregler auf das entsprechende Lautstärkeniveau ein und drücken Sie dann mehrmals auf den roten Knopf. Der Cursor rückt jeweils um eine Stelle nach rechts und setzt dabei die Punkte auf das eingestellte Niveau.

Wenn der Drehregler nicht benutzt wird, sollte er ganz nach links gedreht werden, um andere Kontrollfunktionen (z.B. die eines gleichzeitig angeschlossenen Steuerknüppels) nicht zu beeinträchtigen. Solange die linke Position nicht erreicht ist, blinkt der Punkt unter dem Cursor.

3.3.3.7 STIMMAUSWAHL (1,2,3,4)

Bisher haben wir nur mit einer Stimme gearbeitet. Der ATARI Computer verfügt jedoch über vier Soundgeneratoren, die über diesen Editor angesprochen werden können. Um einige Befehle auszuführen, ist es wichtig zu wissen, auf welche der vier Stimmen diese

angewendet werden sollen. Es ist daher notwendig, mit dieser Auswahl eine Voreinstellung zu treffen. Drücken Sie die Taste "1". Rechts neben der Eins erscheint ein Pfeil. Drücken Sie nun die Tasten "3" und "4". Es erscheinen weitere Pfeile. Sie zeigen an, daß die folgenden Befehle für diese Stimmen auszuführen sind. Ein nochmaliges Betätigen derselben Tasten läßt die Pfeile wieder verschwinden.

3.3.3.8 STIMMAUSWAHL LÖSCHEN (C)

Der "C"-Befehl ist eine Vereinfachung des obigen Befehls. Alle eingesetzten Pfeile verschwinden. Sie können auch die ausgewählten Stimmen einzeln löschen.

3.3.3.9 STIMME SPIELEN (P)

Drücken Sie die Tast "1", danach die Taste "P". Jetzt wird die erste Stimme laufend abgespielt.

3.3.3.10 STIMME TRIGGERN (T)

Sofern noch die erste Stimme ausgewählt ist, betätigen Sie die Taste "T". Der Abspielvorgang der ersten Stimme wird beendet. Gleichzeitig erlischt die erleuchtete Ziffer, die anzeigt, daß diese Stimme gespielt wurde. Der T-Befehl kann auch benutzt werden, um die gewählten Stimmen genau einmal erklingen zu lassen. Dies ist z.B. nützlich, um das Geräusch einer Explosion zu imitieren, oder wenn Sie sich durch das ständige Abspielen gestört fühlen.

Probieren Sie sämtliche Kombinationen der Stimmen mit dem "P"- und "T"-Befehl aus.

3.3.3.11 STIMME EDITIEREN (E)

Wie schon erwähnt, werden mit dem Editor vier Stimmen angesprochen. Durch die Experimente mit dem "P"-Befehl haben Sie sicher festgestellt, daß die Stimme 2-4 des Demoprogrammes ebenfalls belegt sind. Diese können mit Hilfe des "E"-Befehls auf dem Bildschirm in den Arbeitsfeldern dargestellt und verändert werden.

Es ist jedoch notwendig, genau eine Stimme vorher auszuwählen. Geben Sie z.B. "2" und "E" ein, so werden die dargestellten Werte der ersten Stimme durch die der zweiten Stimme ersetzt. Auf diese Weise können alle vier Stimmen angesprochen werden. An dem links neben der Zahl stehenden Punkt können Sie erkennen, an welcher der vier Stimmen Sie gerade arbeiten.

ATARIMAGE.DE

3.3.3.12 STIMMEN RE-SYNCHRONISIEREN (R)

Während des Editiervorganges (ständiges Abspielen) wird es wahrscheinlich vorkommen, daß die Stimmen außer Phase geraten. Benutzen Sie diesen Befehl, um zwei oder mehrere (ausgewählte) Stimmen zu einem gemeinsamen Zeitpunkt zu starten. Die Länge der Stimmen sollte jedoch identisch sein, da sie sonst nach dem zweiten Abspielvorgang wieder außer Phase geraten würden.

3.3.3.13 TIEFER TONBEREICH (LO)

Mit diesem Befehl ist es möglich, die Zeitbasis für die Tonerzeugung von 64 KHz abzusenken. Das Ergebnis ist deutlich hörbar: Geben Sie während des Abspielvorganges ein "L" und dann ein "O" ein, und die Frequenz des gespielten Stückes wird merklich gesenkt. Auf diese Weise ist es möglich, sehr tiefe Töne zu erzeugen, wie sie im BASIC nicht bekannt sind. Technisch geschieht dies, indem das niederwertigste Bit im AUDCTL-Register (dezimal 53768) gesetzt wird.

3.3.3.14 HOHER TONBEREICH (HI)

Dieser Befehl ist die Ergänzung zum "LO"-Befehl. Er stellt den ursprünglichen, normalen Zustand wieder her, indem die Zeitbasis von 15 KHz wieder auf 64 KHz erhöht wird.

3.3.3.15 ANZEIGENBEREICH ÄNDERN (<,>)

In den Arbeitsfeldern können nur Soundlängen von ca. 2 Sekunden dargestellt werden. Für eine Stimme stehen jedoch ca. 4 Sekunden zur Verfügung. Mit diesem Befehl kann nun der Bildschirmausschnitt so verschoben werden, daß auch der übrige Bereich dargestellt werden kann. Die Anzeigen 1-2 weist darauf hin, daß die erste und zweite Sekunde dargestellt wird. Um den Bereich zu ändern, geben Sie solange ein "A" ein, bis die Markierung für die Soundlänge die rechte Hälfte des Arbeitsfeldes erreicht. Drücken Sie dann die ">"-Taste. Das Arbeitsfeld wird neu aufgebaut, es werden nun die Sekunden 2 und 3 dargestellt, ebenfalls angezeigt durch die zwei Ziffern 2-3 im rechten unteren Bildschirmbereich. Bei Fortsetzung dieser Schritte kann der Bereich für die dritte und vierte Sekunde zur Anzeige gebracht werden.

Der Vorgang kann umgekehrt werden, indem Sie die Befehle "D" und "<" anwenden. Sie erhalten schließlich wiederum die Darstellung der ersten und zweiten Sekunde.

3.4 ANWENDUNG IN BASIC-PROGRAMMEN

Der nächste Schritt besteht darin, die erzeugten Klänge in Ihr BASIC-Programm zu integrieren. Auf der Programmdiskette ist ein BASIC-Unterprogramm enthalten, das diese Aufgabe übernimmt. Dieses Unterprogramm liest die Sound-Datei (mit dem "SS"-Befehl erzeugt) bis zu der "End-of-Sound"-Markierung, so daß der Speicherplatzbedarf im BASIC-Programm der tatsächlichen Länge einer Stimme entspricht (und nicht der maximalen gespeicherten Länge von über 4 Sekunden).

Die Integration erfolgt in 3 Schritten:

1. Das Unterprogramm wird mit dem Befehl ENTER "D:INSTALER.LST" an das BASIC-Programm angehängt. Es werden dabei die Programmzeilen 30000-30100 belegt. Achten Sie darauf, daß in Ihrem Programm diese Zeilen nicht belegt sind.

2. In Ihrem Programm müssen Sie drei Strings mit folgender Anweisung initialisieren:

```
DIM CIO$(39),SET$(10),SNDF$(14)
```

3. Der String SNDFILE$ muß den Dateinamen der mit dem "SS"-Befehl abgespeicherten Sounddaten enthalten. Weiterhin muß das Unterprogramm einmal aufgerufen werden, um den Sound zu initialisieren. Dies geschieht mit folgender Anweisung:

```
SNDFILE$="D:DEMO.SND":GOSUB 30000
```

Wenn diese Schritte befolgt werden, ist der Sound in Ihr Programm integriert und kann mit wenigen POKE-Befehlen kontrolliert werden. Die folgende Tabelle zeigt die entsprechenden Speicheradressen und deren Funktion.

ATARIMEDIE

Adresse	Name	Beschreibung	Kommentar
1536	TRIG1	triggert Stimme 1	POKE Adresse, 1 bewirkt ein einmaliges Spielen der Stimme, vorausgesetzt, es wurde der Triggermodus eingestellt (s. unten)
1537	TRIG2	triggert Stimme 2	
1538	TRIG3	triggert Stimme 3	
1539	TRIG4	triggert Stimme 4	
1540	MODE1	Modus von Stimme 1	POKE Adresse, 0 bewirkt ein ständiges Spielen. Mit POKE Adresse, 1 wird der Triggermodus eingeschaltet.
1541	MODE2	Modus von Stimme 2	
1542	MODE3	Modus von Stimme 3	
1543	MODE4	Modus von Stimme 4	
1544	COUNT1	Länge von Stimme 1	Hier ist die Länge jeder Stimme in 1/50 Sekunden gespeichert. Kann z.B. benutzt werden, um einen Motor schneller laufen zu lassen.
1545	COUNT2	Länge von Stimme 2	
1546	COUNT3	Länge von Stimme 3	
1547	COUNT4	Länge von Stimme 4	
1548	PTR1	gespielte Note St. 1	In diesen Adressen ist gespeichert, welche Note gerade gespielt wird. Bereich von 0 bis Countx.
1549	PTR2	gespielte Note St. 2	
1550	PTR3	gespielte Note St. 3	
1551	PTR4	gespielte Note St. 4	
1552,53	FPTAB1	Frequenztab. St. 1	Bei diesen 16-Bit Adressen beginnen die Daten für die Frequenzen. Sie sollten nicht geändert werden.
1554,55	FPTAB2	Frequenztab. St. 2	
1556,57	FPTAB3	Frequenztab. St. 3	
1558,59	FPTAB4	Frequenztab. St. 4	
1560,61	CPTAB1	Kontrolltab. St. 1	Bei diesen 16-Bit Adressen beginnen die Daten für die Audio Control Register. Sie sollten nicht geändert werden.
1562,63	CPTAB2	Kontrolltab. St. 2	
1564,65	CPTAB3	Kontrolltab. St. 3	
1566,67	CPTAB4	Kontrolltab. St. 4	

Um diese Funktion besser zu verstehen, laden Sie nur das Unterprogramm und ergänzen es mit folgenden BASIC-Zeilen:

```
1000 DIM CIO$(39),SET$(10),SNDFILE$(14)
1010 SNDFILE$="D:DEMO.SND":GOSUB 30000
1020 INPUT X,Y: POKE X,Y: GOTO 1020
```

Es werden zwei Daten eingelesen: Adresse und Kontrollwert. Geben Sie z.B. folgende Werte ein:

```
?1541,0 -- um Stimme zwei erklingen zu lassen
?1541,1 -- um Stimme zwei auszuschalten
?1537,1 -- um Stimme zwei einmal erklingen zu lassen
?1540,0 -- um Stimme eins auszuschalten
```

3.5 WARNUNGEN

a) Das Unterprogramm, genannt INSTALER, benutzt folgende Variablen:

CIO$, SET$, SNDFILE$, E, S, X, HI, LO

Sie sollten sie daher nicht in Ihrem BASIC-Programm verwenden, um Probleme zu vermeiden.

b) Der INSTALER löscht den Bildschirm. Graphische Befehle sollten daher erst nach der Initialisierung von INSTALER ausgeführt werden.

c) Der IOCB #6 wird für die Initialisierung von INSTALER benötigt. Erst nachdem diese ausgeführt ist, steht er Ihnen zur Verfügung.

d) Das Programm wird beendet, wenn im INSTALER ein Ein- oder Ausgabefehler auftritt (z.B. wenn die spezifizierte Datei keine Sounddatei ist).

e) Nach jeder Ein- oder Ausgabe über den seriellen Datenbus (z.B. Diskette oder Drucker) ist der Sound mit dem Befehl SOUND 0,0,0,0 neu zu initialisieren, da er sonst nicht mehr abgerufen werden kann.

3.6 TECHNISCHE INFORMATIONEN

Die Effektivität von INSOMNIA beruht auf einem Maschinen-Unterprogramm, VBSOUND genannt, das ebenfalls auf der Programmdiskette unter dem Namen VBSOUND.6 vorhanden ist. Es ist als Vertival Blank Interrupt (VBI) Service Routine im Programm implementiert, d.h. es wird jede 1/50 Sekunde ausgeführt, parallel mit der VBI Service Routine des Betriebssystems. Der Sound wird also automatisch gesteuert und ist somit vom übrigen Programmablauf unabhängig. VBSOUND wird mit Hilfe der Maschinenroutine VBSOUND auf Speicherseite 6 abgelegt.

Der gesamte Bildspeicher wurde nach unten verlegt, um am Ende des RAM-Bereiches Platz für die Sounddaten zu schaffen.

Weitere Informationen zu diesem Thema finden Sie im Kapitel 10 des DE RE ATARI.

Die von INSOMNIA erzeugten Sounddateien haben Standard Binärformat (siehe Disk Operating User Manual). Diese Binärdatei besteht im ersten Teil aus der VBSOUND-Routine mit den entsprechenden Tabellen und Zeigern, sowie einem zweiten Teil, in dem die Sounddaten abgelegt werden.

4. INSTEDIT (Zeichengenerator)

4.1 PROGRAMMBESCHREIBUNG

Bei allen ATARI Computern besteht die Möglichkeit, den vorhandenen ATARI-Zeichensatz durch Zeichensätze eigener Wahl zu ergänzen. Daraus ergeben sich zahlreiche Möglichkeiten, wie z.B. Erzeugen von anderen Schriftzeichen (z.B. die deutschen Umlaute Ä, Ö, Ü,). Erzeugen von speziellen technischen Zeichen (z.B. Integrale, Transistorschaltbild), und das Zeichnen von Bildern (längere Strings), womit sich die Möglichkeit der Computeranimation ergibt (umschalten zwischen verschiedenen Zeichensätzen).

Dieser Zeichensatzgenerator hilft Ihnen dabei, jedes beliebige Zeichen in einer 8 x 8 Matrix zu erzeugen. Dabei wird jedes Zeichen in den sechs möglichen Graphikstufen auf dem Bildschirm abgebildet. Es existiert weiterhin ein Testmodus, in dem die erzeugten Zeichen miteinander kombiniert werden können, um größere Objekte zu erzeugen (z.B. auch für Player-Missile-Graphik).

Die Zeichen werden punktweise erzeugt. Sie können einzelne Zeichen kopieren, löschen, spiegeln, drehen und punktweise in der 8 x 8 Matrix verschieben, jeweils in normaler oder inverser Darstellung. Es kann jederzeit der ATARI Zeichensatz abgerufen werden oder die ursprüngliche Form des bearbeiten Zeichens.

Die Zeichensätze können auf vier verschiedene Arten abgespeichert werden, als Zeichensatz für weitere Bearbeitung, als BASIC-Unterprogramm, als BASIC-Data Anweisungen und als .Byte-Anweisungen für ASSEMBLER Quellprogramme.

4.2 PROGRAMMSTART

Stecken Sie den Stecker des Steuerknüppels in die erste Buchse Ihres Computers. Nachdem Sie das Zeichengenerator-Programm vom Hauptmenü gewählt haben, wird das Programm automatisch gestartet, und es erscheint folgendes Bild:

```
A ATARI      ZEICHEN-
D DREHEN      SATZ-      1 EDITIEREN
I INVERT.      GENE-     2 SPEICHERN
K KOPIEREN      RATOR    3 LADEN
L LÖSCHEN                4 SCHREIBEN
S SPIEGELN   AUSWAHL -->
U URSPRUNG
```

Editiermöglichkeiten 8x8 Matix Hauptauswahl

ATARIMUSEUM

Die vier Möglichkeiten der Hauptauswahl leuchten in unterschiedlicher Weise solange auf, bis eine der Zahlen 1-4 gedrückt wurde.

Weiterhin erscheint der vollständige ATARI Zeichensatz sowie eine Anzahl von Fragezeichen, die in den verschiedenen Graphikstufen dargestellt werden.

4.3 BEDIENUNGSANLEITUNG

Ein Zeichensatz besteht aus 128 verschiedenen Zeichen (ATASCII-Code 0-127), die unter Benutzung der SHIFT- oder CTRL-Taste abgerufen werden können. Die übrigen Zeichen des ATARI-Zeichensatzes (dez. Code 128-255) sind keine eigenständigen Zeichen, sondern es handelt sich lediglich um eine andere farbliche Darstellung. Jedes einzelne Zeichen besteht aus 8x8 Punkten. Je 8 Punkte (=8 Bits) einer Zeile bilden ein Byte, d.h. jedes Zeichen erfordert 8 Byte. Die Bits eines Zeichens können nun ganz gezielt verändert werden. Die verschiedenen Möglichkeiten werden in den nächsten Kapiteln ausführlicher erklärt.

4.3.1 ZEICHEN EDITIEREN

Um spezielle Zeichen zu erzeugen, geben Sie eine "1" ein (1 - Editieren). Es erscheint die Frage:

WELCHES ZEICHEN ?

Je nachdem, welche Taste oder Tastenkombination Sie drücken, erscheint das entsprechende Zeichen in der 8x8 Matrix. (Anm.: Sofern Sie keinen anderen Zeichensatz geladen haben, erscheinen die bekannten Zeichen des ATARI-Zeichensatzes, z.B. Taste R für "r" (vorher CAPS LOWR drücken), Taste SHIFT + R für "R" und Taste CTRL + R für einen senkrechten Balken). Parallel dazu verändert sich der Bildschirm in folgender Weise:

a) Die Fragezeichen werden durch das neugewählte Zeichen ersetzt.

b) Rechts oben über der Hauptauswahl erscheint der ATASCII-Code des entsprechenden Zeichens. (z.B. CHR$(114) = "r").

Sie haben nun grundsätzlich drei Möglichkeiten, dieses Zeichen zu verändern

a) Mit Hilfe des Steuerknüppels
b) Mit Hilfe der linken Menüauswahl
c) Mit Hilfe der Cursor-Tasten

Zu a) Bringen Sie mit Hilfe des Steuerknüppels das Rechteck an die Position, an der Sie eine Veränderung am Zeichen vornehmen wollen. Gehört diese Stelle zum Zeichen, kann durch Drücken auf den roten Knopf dieser Punkt gelöscht werden. Umgekehrt wird an einer "leeren" Position durch Betätigen des roten Knopfes ein Punkt erzeugt. Sie erhalten zur Kontrolle des Vorgangs ein akustisches Signal, das beim Löschen eines Punktes durch Fallen der Tonhöhe, beim Setzen eines Punktes durch Ansteigen der Tonhöhe angezeigt wird. Die Veränderung des Zeichens geschieht nicht nur im Bearbeitungsfeld, sondern auch auf der gesamten unteren Bildschirmhälfte, so daß auch die Wirkung auf die anderen Zeichen gesehen werden kann.

Zu b) Mit der im linken Bildschirmrand stehenden Auswahl werden die Möglichkeiten, ein Zeichen zu verändern, wesentlich erweitert. Ihnen stehen 7 Befehle zur Verfügung:

A ATARI
D DREHEN
I INVERTIEREN
K KOPIEREN
L LÖSCHEN
S SPIEGELN
U URSPRUNG

Diese Befehle können jederzeit und in beliebiger Reihenfolge gegeben werden. Sie bedeuten im Einzelnen:

A - Es erscheint jeweils das Zeichen des ATARI-Zeichensatzes, das dem ATASCII-Code der gedrückten Taste entspricht. Der ATARI-Zeichensatz kann jederzeit abgerufen werden, unabhängig davon, welcher Zeichensatz gerade bearbeitet wird.

D - Durch Drücken des Taste "D" wird das in der Matrix stehende Zeichen um 90 Grad im Uhrzeigersinn gedreht. Wiederholen Sie diesen Befehl viermal, so steht das Zeichen wieder in seiner ursprünglichen Stellung.

I - Mit dieser Funktion wird das gerade bearbeitete Zeichen in dem Anzeigenmodus "Video invers" dargestellt. Nochmaliges Drücken bewirkt die Rückkehr in die Normalform.

K - Mit diesem Befehl ist es möglich, das zu bearbeitende Zeichen durch jedes beliebige andere Zeichen des vorhandenen Zeichensatzes zu ersetzen. Geben Sie auf die Frage:

WELCHES ZEICHEN KOPIEREN?

das Zeichen an, welches den Platz des gerade bearbeiteten Zeichens einnehmen soll.

ATARINSIDE

L - Mit diesem Befehl wird das Zeichen gelöscht, so daß es mit dem Leerzeichen identisch ist. Sie haben nun die Möglichkeit, ein Zeichen neu zu entwerfen.

S - Mit diesem Befehl wird das Zeichen an der senkrechten Achse gespiegelt. Beachten Sie, daß bei spiegelsymmetrischen Zeichen wie z.B. "A" dieser Befehl scheinbar keine Wirkung hat.

U - Mit dieser Funktion wird das Zeichen in der Form dargestellt, wie es zum Zeitpunkt des Aufrufs (mit "1") im Speicher stand. Es wird sich daher beim ersten Programmlauf nicht von dem jeweiligen ATARI-Zeichen unterscheiden. Erst wenn wieder ein Befehl der Hauptauswahl angeführt wird, wird das Zeichen aus der Matrix in den Zeichenspeicher übernommen, so daß Sie bei erneutem Editieren desselben Zeichens auf die letzte Form zurückgreifen können.

Zu c) Die dritte Möglichkeit, ein Zeichen zu bearbeiten, besteht darin, die Position des Zeichens innerhalb der Matrix mit Hilfe der Cursor-Tasten (Pfeil-Tasten) zu verändern. Drücken Sie eine der Cursor-Tasten (ohne CTRL und SHIFT), und das Zeichen verschiebt sich in die entsprechende Richtung.

Mit den genannten Befehlen sind Ihnen alle denkbaren Möglichkeiten gegeben, ein Zeichen zu erzeugen. Um ein anderes Zeichen zu editieren, können Sie jederzeit die "1" drücken. Es wiederholt sich dann der Ablauf, wie in diesem Abschnitt beschrieben. Sind Sie mit dem erzeugten Zeichensatz zufrieden, können Sie diesen auf verschiedene Arten abspeichern. Betätigen Sie dazu entweder die Taste "2" oder "4". Beide Möglichkeiten werden jedoch noch ausführlich beschrieben.

4.3.2 ZEICHENSATZ ABSPEICHERN

Durch Eingabe einer "2" kann jederzeit der Editiermodus abgeschlossen werden und der bis dahin entwickelte Zeichensatz auf Diskette gespeichert werden. Dieser Zeichensatz kann zu einem späteren Zeitpunkt wieder in den Speicher geladen und mit dem Zeichengenerator weiter verändert werden. Zunächst erscheint die Frage:

NAME DES ZEICHENSATZES?

Geben Sie den Namen ein (max. 8 Buchstaben), unter dem der Zeichensatz abgespeichert werden soll. Die Nummer der Diskettenstation entfällt, da eine Voreinstellung für die Disketten-station (D1:) existiert. Ebenso kann keine Extension angegeben werden, da alle Zeichensätze automatisch die Extension .SET erhalten.Geben Sie keinen Namen ein und betätigen gleich die RETURN Taste, so wird das Kommando abgebrochen und Sie gelangen in den Anfangszustand zurück. Der Computer wartet auf eine neue Eingabe (Ziffern 1-4).

4.3.3 ZEICHENSATZ LADEN

Mit der Eingabe einer "3" kann ein auf Diskette gespeicherter Zeichensatz für eine weitere Bearbeitung wieder in den Speicher geladen werden. Um Ihnen einen Eindruck über die unterschiedlichen Zeichensätze zu vermitteln, befinden sich auf der Programmdiskette drei verschiedene Zeichensätze (ATARI.SET, SCRIPT.SET, COMPUTER.SET). Geben Sie auf die Frage:

WELCHEN ZEICHENSATZ LADEN?

z.B. den Namen COMPUTER ein und schließen Sie die Eingabe mit RETURN ab. Sobald der Zeichensatz geladen ist, werden alle Zeichen durch eine neue Form ersetzt. Sie können nun mit Hilfe der bekannten Befehle diesen Zeichensatz verändern. Geben Sie statt eines Namens nur RETURN ein, so wird der Ladebefehl abgebrochen und Sie gelangen in den Anfangszustand zurück, in dem der Computer auf eine neue Eingabe wartet.

4.3.4 ZEICHENSATZ SCHREIBEN

Um die erzeugten Zeichensätze auch problemlos in Ihre BASIC- ODER ASSEMBLER-Programme zu integrieren, gibt es folgende drei Möglichkeiten, Zeichensätze auf Diskette zu schreiben:

1. als BASIC-Unterprogramm
2. als DATA Zeilen für BASIC-Programme
3. als Byte-Anweisungen für ASSEMBLER-Programme.

Nach Eingabe einer "4" (für schreiben) erscheint folgende Auswahl auf dem Bildschirm:

```
WELCHE ART : 1. BASIC
             2. DATA
             3. .BYTE
```

Wählen Sie die Zahl für die entsprechende Schreibmethode.

4.3.4.1 BASIC-UNTERPROGRAMME

Nach Eingabe der "1" erscheint die Frage:

NAME DES ZEICHENSATZES?

Geben Sie nun einen Namen ein (z.B. TEST) und beenden Sie die Eingabe mit RETURN. Es wird nun auf der Diskette eine Datei mit LIST-Format erzeugt, die den eingegebenen Namen sowie die Extension .LST trägt (z.B. TEST.LST). Dieses Unterprogramm kann später mit dem ENTER-Befehl (z.B. ENTER"D:TEST.LST) an jedes beliebige BASIC-Programm angehängt werden. Vom Unterprogramm werden die Zeilen 0 sowie der Bereich 30000 - 31000 benutzt. Das

ATARIMUSEUM.DE

Unterprogramm wird mit GOSUB 30000 aufgerufen. Es kann jedoch für Testzwecke auch selbständig mit RUN gestartet werden (RETURN in der letzten Programmzeile entfernen). Beachten Sie, daß für diese Datei 14 freie Sektoren auf der Diskette benötigt werden.

4.3.4.2 DATA-ZEILEN

Die zweite Möglichkeit, einen Zeichensatz abzuspeichern, besteht darin, die einzelnen Bytes eines Zeichens (jedes Zeichen besteht aus 8 Bytes)in Form von Dezimalzahlen in DATA-Zeilen zu speichern. Jede DATA-Zeile enthält 24 Dezimalzahlen, d.h. in jeder DATA-Zeile werden drei Zeichen gespeichert. Um einen vollständigen Zeichensatz abzuspeichern, sind somit 43 DATA-Zeilen erforderlich (entspricht ca. 3K). Der Vorteil bei dieser Speicherart liegt darin, daß kein vollständiger Zeichensatz abgespeichert werden muß und daß auf einzelne Zeichen besser zugegriffen werden kann. Mit Hilfe der READ und POKE Befehle können diese Werte direkt in die Speicherzellen eingeschrieben und für bestimmte Zwecke (z.B. Computeranimation) verwendet werden. Der Abspeichervorgang verläuft im Einzelnen folgendermaßen:

1. Sie geben einen Dateinamen an. (Vom Programm wird automatisch die Endung .DAT angehängt).

2. Sie geben dann an, bei welcher Zeilennummer die erste DATA-Zeile geschrieben werden soll (z.B. 1000).

3. Als nächstes geben Sie den Zeilenabstand ein (z.B. "2" für die Nummerierung 1000,1002,1004,1006,...).

4. Auf die Frage "ERSTES ZEICHEN?" geben Sie die Taste oder Tastenkombination ein, die dem ersten Zeichen entspricht, das gespeichert werden soll. (Beachten Sie, daß der ATASCII-Code über die Reihenfolge entscheidet).

5. Auf die Frage "WIEVIELE ZEICHEN?" geben Sie die Anzahl der Zeichen an, die abgespeichert werden sollen (Maximal 127 Zeichen).

6. Auf die Meldung "STARTEN MIT RETURN" können Sie in verschiedener Weise reagieren. Entweder Sie geben ein RETURN ein, dann wird die Datei nach den von Ihnen vorgegebenen Werten geschrieben, oder Sie drücken die ESC-Taste, dann werden die eingegebenen Daten als gegenstandslos betrachtet und Sie gelangen in den Anfangszustand zurück.

7. Die erzeugte Datei kann später mit dem ENTER-Befehl an existierende BASIC-Programme angehängt werden (z.B. mit ENTER"S:TEST.DAT"). Achten Sie darauf, daß keine Überschneidungen zwischen Zeilennummern stattfinden.

4.3.4.3 BYTE-ANWEISUNGEN

Die Arbeitsweise verläuft analog Absatz 3.4. Die erzeugten Dateien benötigen jedoch 43 Sektoren auf einer Diskette. Die Endung für diese Dateien lautet .BYT. Die Zeichen werden auch in diesem Fall byt-weise als Hexadezimalzahlen in den .BYTE-Anweisungen dargestellt.

4.3.5 TESTMODUS

Um die Wirkung zwischen editierten Zeichen besser beurteilen zu können oder um größere Objekte zu entwickeln, die sich aus mehreren Zeichen zusammensetzen, existiert ein sogenannter Testmodus, in dem die Zeichen in sechs verschiedenen Textmodi dargestellt werden können. Es sind dies die drei bekannten BASIC-Textmodi 0,1 und 2, sowie die weniger bekannten Textmodi 3,4 und 5, auf die Sie ebenfalls zugreifen können (siehe Kapitel 4.4). Der Modus IR 3 (IR=Instruction Register) unterscheidet sich vom BASIC-Modus 0 (8x8 Matrix) durch die Zeichenhöhe (10x8 Matrix), d.h. es können z.B. Zeichen mit Unterlängen dargestellt werden. Die Modi IR 4 und 5 arbeiten ebenfalls mit 40 Zeichen pro Zeile, unterscheiden sich jedoch in der Zeichenhöhe und in der Tatsache, daß die einzelnen Zeichen vielfarbig dargestellt werden. Die Farben der Zeichen können mit Hilfe des SETCOLOR-Befehls verändert werden.

Sie gelangen in den Testmodus, indem Sie die OPTION-Taste drücken. Es erscheint die Auswahl:

0-2 GR.
3-5 IR.
6 HALTEN
7 START

Nach Eingabe einer Zahl (0 bis 5) wird der untere Bildschirmbereich gelöscht und Sie können im jeweiligen Textmodus beliebig schreiben und die Zeichen miteinander kombinieren. Durch nochmaliges Drücken der OPTION-Taste gelangen Sie wieder in den Editiermodus zurück. Die auf dem Bildschirm erzeugten Zeichen bleiben dabei erhalten.

Verlassen Sie dagegen den Testmodus durch Eingabe einer "7", so wird das Programm neu gestartet. Der gerade bearbeitete Zeichensatz geht dabei jedoch verloren, sofern er nicht vorher auf Diskette gesichert wurde.

Geben Sie eine "6" ein, so wird die untere Bildschirmhälfte nicht gelöscht (im Gegensatz zu der Eingabe "0"-"5"), und Sie gelangen in den Textmodus, in dem Sie schon geschrieben haben (funktioniert also nur, wenn Sie schon einmal in einem der Textmodi 0-5 Zeichen auf dem Bildschirm geschrieben haben).

ATARIINSIDE

4.4 BEISPIELE

Auf der Programmdiskette befinden sich nicht nur die Beispielzeichensätze ATARI.SET und SCRIPT.SET, sondern auch zwei Beispielprogramme, die die Anwendung der Textmodi IR 3,4 und 5 zeigen.

Die Programme können mit

RUN"D:DEMO1

und

RUN"D:DEMO2

gestartet werden.

DEMO1 zeigt in der oberen Bildschirmhälfte den BASIC-Modus 0 und im unteren Teil den IR Modus 3. Es wird der normale ATARI-Zeichensatz verwendet, so daß die Kleinbuchstaben im IR Modus beschnitten werden. Dieses Problem läßt sich jedoch mit Hilfe eines speziellen Zeichensatzes lösen.

DEMO2 ist ein Beispiel für die IR Modi 4 und 5. Die mehrfarbigen Zeichen können mit Hilfe der OPTION-, SELECT- und START-Tasten verändert werden.

5. DSEMBLER

5.1 PROGRAMMBESCHREIBUNG

Der DSEMBLER ist ein geeignetes Instrument, um den Maschinencode des 6502 Mikroprozessors besser verstehen und erlernen zu können. Durch gezielte Entscheidungen kann ein weites Aufgabenfeld abgedeckt werden. So können Sie z.B. den gesamten im Speicher vorhandenen Maschinencode disassemblieren und im folgende Format auf dem Bildschirm oder Drucker ausgeben lassen:

1. Adresse der disassemblierten Anweisung
2. Hexadezimalwert eines jeden Byte
3. Name des Labels, falls vorhanden
4. Operationscode der jeweiligen Anweisung
5. Die Operanden (im 6502 Standartformat); wenn sie die Adressen markieren, wird beim Debugvorgang anstelle der Adresse das Label erscheinen.

Mit dem DSEMBLER können Sie weiterhin folgende Aufgaben durchführen:

1. Umwandlungen von Dezimalzahlen im Bereich von 0 bis 65535 in Hexadezimalzahlen und umgekehrt.
2. Erzeugen von Labeln, d.h. einer Adresse wird ein Name zugeordnet. Weiterhin ist das Umbenennen oder Entfernen von schon existierenden Labeln möglich.
3. Ausdrucken einer Tabelle aller existierenden Labels.
4. Abspeichern der Labels als File auf Cassette oder Diskette.
5. Ein Maschinenprogramm unter Benutzung des Hexcodeladers sehr schnell in den Speicherbereich laden.
6. Sie können sich den gesamten Speicherbereich seitenweise auflisten lassen (z.B. zum schnellen Durchsuchen).
7. Sprünge zu Maschinenprogrammen, mit Benutzung der X, Y und A Register.
8. Der gesamte Speicherbereich kann nach einer bestimmten Byte-Folge durchsucht. Wildcards sind zulässig.
9. Durch einen vorhandenen Freiraum kann der DSEMBLER leicht für andere Aufgaben erweitert werden.

Warnung:
Machen Sie sich für den Fall, daß das System 'abstürzt', vom momentan bearbeiteten Code eine Kopie.

5.2 PROGRAMMSTART

Nachdem Sie das DSEMBLER Programm vom Hauptmenü gewählt haben, wird es automatisch gestartet. Auf dem Bildschirm erscheint die:

DSEMBLER HAUPTAUSWAHL:

ATARIMAGE

5.3 BEDIENUNGSANLEITUNG

5.3.1 DATENEINGABE UND PROGRAMMUNTERBRECHUNG

Der DSEMBLER ist ein Programm, das sich hervorragend für das Erlernen der 6502 Maschinensprache eignet. Die vorliegenden Informationen sind zwar sehr umfangreich, das Arbeiten mit diesem Programm ist jedoch sehr einfach. Die folgenden Hinweise werden Ihnen das Einarbeiten in dieses Programm erleichtern.

Wenn der DSEMBLER gerade kein Kommando abarbeitet, wartet er auf eine Eingabe von Ihnen. Alle Eingaben werden über die Tastatur des Computers eingegeben und mit RETURN abgeschlossen. In einigen Fällen ist eine Eingabe nicht unbedingt erforderlich, das Programm kann mit RETURN fortgesetzt werden. In diesen Fällen werden voreingestellte Werte benutzt (z.B. bei den Kommandos D, P und T). In allen anderen Fällen führt ein RETURN ohne vorherige Eingabe zum Abbruch und das Programm fragt nach einer neuen Aufgabe.

Der DSEMBLER verarbeitet Integerzahlen aus dem Bereich 0 bis 65535 (bzw. $0 bis $FFFF). Das Benutzen anderer Werte kann problematisch werden. Der DSEMBLER akzeptiert sowohl Dezimal- als auch Hexadezimalzahlen. Hexadezimalzahlen sind immer mit dem Dollar-Zeichen ($) einzugeben.

5.3.2 HAUPTAUSWAHL

5.3.2.1 MENÜ

Nach dem Programmstart erscheint folgende Auswahl von Kommandos auf dem Bildschirm:

```
DSEMBLER - HAUPTAUSWAHL :

  A: Hauptauswahl        P: Speicherseiten
  B: Byte-Suche          R: Label abrufen
  D: Disassemblieren     S: Label speichern
  H: Hexcode laden       T: Label Tabelle
  J: JSR (Sprung)        V: Voreinstellungen
  K: Label löschen       Z: Zahlenumwandlung
  L: Label erzeugen
```

Auf die Frage: "WELCHER BUCHSTABE (HAUPTAUSWAHL) ?" (diese Frage wird noch häufiger erscheinen) geben Sie einen der Buchstaben ein (z.B. Z für Zahlenumwandlung) und schließen die Eingabe mit RETURN ab. Sie gelangen so in die entsprechenden Unterprogramme.

Die Hauptauswahl erscheint nicht nur nach dem Programmstart, sondern auch in folgenden Fällen:

1. Wenn auf Fragen mit dem Buchstaben A (für Hauptauswahl) geantwortet wird.
2. Wenn Aufgaben mit RETURN (d.h. ohne Eingabe von Daten) abgebrochen werden.

5.3.2.2 BYTE-SUCHE

Nach Eingabe des Buchstaben B erscheint folgende Auswahl auf dem Bildschirm:

```
BYTE-SUCHE:

1. Startadresse = $00
2. Endadresse   = $FF
3. Wild card ?    Nein
4. Byts eingeben

Welche Nummer?
```

Mit Hilfe dieses Unterprogramms können Sie den gesamten Speicherraum nach einer von Ihnen bestimmten Bytefolge durchsuchen lassen. Für ein einzelnes Byte können Sie eine sogenannte Wildcard bilden, d.h. Sie brauchen den Wert des Byts nicht exakt zu spezifizieren. Während der Suche ignoriert das Programm die Wildcard, es sei denn es existiert dafür ein Platzhalter.

Die Kommandos 1 bis 4 haben folgende Bedeutung:

1. Startadresse : Hiermit wird der Bereich festgelegt, an dem die Suche beginnen soll.

2. Endadresse : Sie bestimmt bei welcher Adresse die Suche abgebrochen werden soll.

3. Wildcard : Benutzen Sie dieses Kommando, um den Wert der Wildcard zu bestimmen. Wenn Sie keine Wildcard verwenden wollen, geben Sie einen Wert ein, der größer ist als 255 ($FF).

4. Bytes eingeben: An dieser Stelle können Sie die Bytes eingeben, nach denen Sie suchen lassen wollen. Jedes Byte wird einzeln eingegeben. Die Anzahl der eingegebenen Bytes muß zwischen 2 und 14 liegen. Nach Eingabe eines RETURN wird der Suchvorgang ausgelöst. Jede Adresse, die den spezifizierten Bytes entsprach, wird ausgedruckt.

Statt einer Zahl können Sie auch einen Buchstaben der Hauptauswahl eingeben. Sie gelangen so direkt in ein anderes Unterprogramm oder wieder in die Hauptauswahl (z.B. mit A).

ATARISIDE

Ein weiteres Beispiel soll helfen, dieses Unterprogramm besser zu verstehen. Wollen Sie z.B. im Speicherbereich in dem sich das BASIC-ROM befindet ($A000 - $BFFF), einen LDA Befehl, der von einem BNE Befehl gefolgt, wird suchen lassen, so müßte die Eingabe wie folgt aussehen:

```
B RETURN        -- Unterprogramm Byte-Suche

1 RETURN
$A000 RETURN    -- Startadresse eingeben

2 RETURN
$BFFF RETURN    -- Endadresse eingeben

3 RETURN
$FF RETURN      -- Wildcard setzen


F RETURN        -- zur Kontrolle

4 RETURN        -- Bytes eingeben
$A5 RETURN      -- Code für LDA
$FF RETURN      -- Wildcard für den Operanden
$DO RETURN      -- Code für BNE
```

Der folgende Ausdruck hängt von Ihrem BASIC-ROM ab. Es können daher einige Adressen Datenfelder repräsentieren und keinen Maschinencode. Sie können dann eine Überprüfung mit dem Unterprogramm D (disassemblieren) vornehmen.

5.3.2.3 DISASSEMBLIEREN

Die Hauptaufgabe des DSEMBLER besteht darin, den im Speicher vorhandenen Maschinencode zu disassemblieren. Wie bei jedem anderen Disassembler sollten Sie jedoch nicht in einem Datenfeld,oder beim 2. oder 3. Byte einer Anweisung den Disassembliervorgang beginnen, da Sie sonst unsinnige Ergebnisse erhalten.

Zunächst wird nach der Startadresse gefragt. Geben Sie statt einer Zahl gleich RETURN ein, so wird ein voreingestellter Wert genommen. In diesem Fall wird an der Stelle fortgesetzt, an der der letzte Disassembliervorgang geendet hat. Als nächstes ist einzugeben, wieviele Anweisungen disassembliert werden sollen. Voreingestellter Wert ist 16. (Geben Sie statt einer Ziffer einen Buchstaben der Hauptauswahl an, so gelangen Sie sofort in das gewählte Unterprogramm.)

Der Ausdruck, den Sie erhalten, ist aus folgenden 5 Feldern zusammengesetzt:

Feld 1: Enthält die Adresse (hexadezimal) der disassemblierten Anweisung.

Feld 2: Enthält die Bytes (hexadezimale Werte) der Anweisung.

Anmerkung: Beide Felder benutzen nicht das Dollar-Zeichen zur Kennzeichnung von Hexadezimalzahlen.

Feld 3: Enthält den Labelnamen, sofern die Adresse mit einem Label bezeichnet wurde.

Feld 4: Enthält den Operationscode der Anweisung.

Feld 5: Enthält die Operanden (sofern vorhanden) unter Benutzung des 6502 Standard Formats. Steht der Operand in Beziehung zu einem Label, wird statt der Adresse das Label ausgedruckt. Direktoperanden weisen niemals auf Label.

Sie gelangen in die Hauptauswahl zurück, indem Sie z.B. statt der Startadresse den Buchstaben A eingeben. Wollen Sie statt der Bildschirmausgabe einen Ausdruck auf dem Drucker, lesen Sie Kapitel 5.3.2.11 (Voreinstellungen).

5.3.2.4 HEXCODE LADEN

Der DSEMBLER enthält zwar keinen Assembler, jedoch kann mit diesem Unterprogramm Maschinencode sehr leicht in den Speicherbereich geladen werden (für die meisten Assemblerprogramme wird der Hexadezimalcode der Bytes mit angegeben). Das Programm fragt als erstes nach der Adresse, bei welcher der Ladeprozeß beginnen soll. Danach kann Byte für Byte eingegeben werden. Der Ladeprozeß wird beendet, indem Sie statt eines Bytes ein RETURN eingeben.

Anmerkung:
Geben Sie jedes Byte als ein- oder zweiziffrige Hexadezimalzahl ein, jedoch OHNE DOLLARZEICHEN. Dies ist die einzige Abweichung von der Regel, daß Hexadezimalzahlen mit dem $-Zeichen gekennzeichnet werden müssen.

5.3.2.5 JSR (JUMP SUBROUTINE)

Benutzen Sie diesen Befehl, um zu maschinengeschriebenen Unterprogrammen zu springen. Dabei kann das X, Y und A-Register vorbesetzt werden. Das Programm fragt nach jedem Register, sowie der Startadresse, zu der gesprungen werden soll. Es kann auch das P-Register vorbesetzt werden. Lesen Sie bitte im Abschnitt 5.3.2.11 (Voreinstellungen) nach. Nachdem der Rücksprung (RTS) aus dem Maschinenprogramm erfolgt ist, wird die Kontrolle an den DSEMBLER zurückgegeben und es erscheinen die Inhalte der X-, Y-, A- und P-Register auf dem Bildschirm.

Anmerkung:
Wenn Sie keine genauen Kenntnisse über diese Materie haben, wird das System zusammenbrechen. In einem solchen Fall schalten Sie einfach den Computer aus und laden den DSEMBLER, wie unter 5.1 beschreiben, wieder in den Speicherbereich.

5.3.2.6 LABEL ERZEUGEN

Ein Label ist ein Name (Marke), der einer Adresse im Speicher entspricht. Als erstes wird nach der Adresse gefragt, die mit einem Label bezeichnet werden soll. Danach ist der Labelname einzugeben. Dieses Unterprogramm kann auch benutzt werden, um schon existierende Label umzubennen. Die Umbenennung erfolgt, indem einfach die alte Adresse angegeben wird, sowie der neue Name.

Die folgenden Regeln sollten außerdem beachtet werden:

1. Es dürfen maximal 254 Label erzeugt werden.

2. Die Adresse $00 sollte nicht mit einem Label bezeichnet werden.

3. Labelnamen müssen aus mindestens 2 Zeichen und dürfen maximal aus 8 Zeichen bestehen. Obwohl das Programm keine Vorschriften in Bezug auf die Zeichen macht, sollten Sie nur solche Zeichen verwenden, die von Ihrem Drucker darstellbar sind.

5.3.2.7 LABEL LÖSCHEN

Dieses Kommando können Sie benutzen, um vorhandene Label wieder zu entfernen. Auf die Frage "Labeladresse ?" geben Sie die Adresse des Labels an, welches gelöscht werden soll. Zur Verdeutlichung ein Beispiel: Sie wollen ein Label entfernen, das ANFANG genannt wurde. Leider ist Ihnen die Adresse nicht bekannt. In diesem Fall müssen Sie in der Label-Tabelle nachsehen, unter welcher Adresse das Label zu finden ist (wird in Abschnitt 5.3.2.9 beschrieben). Ist Ihnen die Adresse bekannt, geben Sie diese ein und schließen die Eingabe mit RETURN ab. Sie erhalten dann die Meldung "ok... ANFANG gelöscht", zur Bestätigung, daß das Label entfernt wurde.

5.3.2.8 LABEL SPEICHERN UND ABRUFEN

Mit diesem Kommando können Sie die verwendeten Labels für spätere Anwendungen auf Cassette oder Diskette sichern. Geben Sie auf die Frage "Welcher Buchstabe (Hauptauswahl)?" ein S ein, so können Daten gesichert werden, und mit R wieder in den Speicher geladen werden.

Bei Benutzung des R Kommandos werden alle im Speicher befindlichen Label gelöscht und durch die vom Speichermedium abgerufenen Label ersetzt. Es hat sich als günstig erwiesen, die Label (Standardlabel), mit denen Sie häufiger arbeiten und die für sie eine bestimmte Bedeutung haben, gleich nach dem Programmstart in den Speicher zu laden. Sie können für die speziellen Abschnitte im Speicher spezielle Label anlegen und diese dann abspeichern.

Für Diskettenbenutzer:

Auf die Frage "Eingabe: (D:FILENAME oder C)" geben Sie den Filenamen an, unter dem die Label gespeichert werden sollen. Wir empfehlen die Endung LBL an den Filenamen anzuhängen, um zu verdeutlichen, daß es sich um einen Label-File handelt (also z.B. D:STANDART.LBL oder für Speichern auf einer zweiten Diskettenstation D2:STDRT.LBL). Beachten Sie, daß ein Label-File 23 Sektoren benötigt.

Für Cassettenbenutzer:

Auf die Frage "Eingabe: (D:FILENAME oder C)" geben Sie einfach das C ein, da in Verbindung mit dem Programmrecorder keine Filenamen benutzt werden können. Dann befolgen Sie die bekannten Anweisungen für den Betrieb des Programmrecorders.

5.3.2.9 LABEL TABELLE

Mit diesem Kommando erhalten Sie eine Tabelle, in der alle Label mit den dazugehörigen Adressen abgelegt sind. Geben Sie auf die Frage "Anfangsbuchstabe ?" den Buchstaben der Label an, die Sie auflisten lassen wollen (z.B. F für alle Label, die mit F beginnen oder GAT für alle Label die mit der Kombination GAT beginnen). Geben Sie keinen Buchstaben an und drücken sofort RETURN, werden alle vorhandenen Label aufgelistet. Auf dem Bildschirm erscheinen nicht mehr als 40 Label zur Zeit. Um die Tabelle fortzusetzen, betätigen Sie einfach die RETURN Taste. Sie können jedoch auch die Ausgabe abbrechen, indem Sie einen Buchstaben der Hauptauswahl eingeben (z.B. A für Rückkehr in die Hauptauswahl). Wenn alle Label ausgedruckt wurden, erhalten Sie die Meldung "Ende der Tabelle". Diese Meldung erhalten Sie auch, wenn keine Label in der Tabelle eingetragen sind. Wollen Sie die Tabelle auf dem Drucker ausgegeben haben, so lesen Sie bitte den Abschnitt 5.3.2.11 (Voreinstellungen).

5.3.2.10 SPEICHERSEITEN

Der gesamte Speicher Ihres Computers ist in 256 "Seiten" mit jeweils 256 Byte unterteilt (256*256 entspricht 64 K). Die Seitennummer entspricht also dem höherwertigen Byte einer Adresse. Die Hauptanwendung des Kommandos "P"

liegt darin, den Speicher schnell nach bestimmten ATASCII-Zeichen zu durchsuchen. Geben Sie auf die Frage "Welche Seitennummer ?" die Nummer der Seite an, die auf dem Bildschirm aufgelistet werden soll. Ein weiteres Betätigen der RETURN Taste läßt jeweils die nächste Seite auflisten. Geben Sie stattdessen einen Buchstaben der Hauptauswahl an, so gelangen Sie in das entsprechende Unterprogramm.

Der umfangreiche Zeichensatz der ATARI Computer erlaubt es, jedes Byte anders auf dem Bildschirm darzustellen. Einzige Ausnahme ist der Wert 155 (CHR$(155)=RETURN), dessen Kontrollfunktion nicht verhindert werden kann. Aus diesem Grund wird dieses Byte durch das Escape-Zeichen dargestellt und es ertönt zusätzlich ein Piepton.

5.3.2.11 VOREINSTELLUNGEN

Nach Eingabe des Buchstaben V erhalten Sie folgende Auswahl:

1: Labelanzahl
2: Drucker an/aus
3: P-Register setzen
4: Alle Files der Diskette
5: Alle *.LBL Files

Auf die Frage "Welche Nummer?" ist eine der Zahlen einzugeben. Anweisung 2 und 3 sind mit Schaltern zu vergleichen. Beim ersten Aufruf werden sie gesetzt, beim zweitenmal zurückgesetzt. Sie erhalten zur Kontrolle eine Meldung welcher Zustand eingenommen wurde.

zu 1: Nach Eingabe der 1 wird die Anzahl, der im Speicher vorhandenen Label ausgedruckt.

zu 2: Diese Voreinstellung hat nur Auswirkungen auf die Unterprogramme D (disassemblieren) und T (Label Tabelle). Im ersten Fall wird der disassemblierte Code auf dem Drucker ausgegeben, im zweiten Fall sowohl auf dem Drucker, als auch auf dem Bildschirm.

zu 3: Diese Anweisung hat nur Auswirkung auf das Unterprogramm J (JSR). Ist die Voreinstellung entsprechend vorgenommen, so muß bei jedem Sprung zu einem Maschinenprogramm das Prozessor Status Register notwendig werden. Dies ist notwendig, da sonst die Wahrscheinlichkeit für den Systemzusammenbruch sich erhöht. Sie können dieses jedoch noch verhindern, indem Sie das Statusregister mit P=32 oder P=$20 besetzen.

zu4: und 5: Nur für Diskettenbenutzer:
Anweisung 4 listet alle Files auf, die auf Diskette vorhanden sind, Anweisung 5 nur die mit der Endung LBL. Auf die Frage "Welche Diskettenstation ?" ist die Nummer der Diskettenstation einzugeben, die die entsprechende Diskette enthält. Voreingestellter Wert ist 1, d.h. statt einer 1 können Sie gleich RETURN eingeben.

5.3.2.12 ZAHLENUMWANDLUNG

Mit diesem Unterprogramm können Dezimalzahlen in Hexadezimalzahlen umgewandelt werden und umgekehrt. Hexadezimalzahlen müssen mit dem Dollar-Zeichen ($) gekennzeichnet werden. Es werden nur Zahlen aus dem Bereich 0 bis 65535 bzw. $0 bis $FFFF umgewandelt. Geben Sie auf die Frage "Welche Nummer ?" statt einer Zahl einen Buchstaben der Hauptauswahl an, so gelangen Sie direkt zum gewünschten Unterprogramm.

ATARIINSIDE

6. AUTOSTART

6.1 EINFÜHRUNG

Mit AUTOSTART können Sie beim Einschalten Ihres Compters automatisch ein BASIC-Programm starten. Um den automatischen Programmstart zu erreichen, lassen Sie das AUTOSTART Programm einmal laufen. AUTOSTART fragt dann nach einem RUN oder LOAD Befehl für das BASIC Programm, das automatisch ausgeführt werden soll. Anschließend erzeugt es ein AUTORUN.SYS File, das Ihr Programm ausführt, sobald Sie Ihren Computer eingeschaltet und die Programmdiskette eingeladen haben.

Hier sind einige Beispiele für die vielfältigen Nutzungsmöglichkeiten für AUTOSTART.

1. Wenn Sie eine Diskette mit mehreren Programmen haben, die Sie normalerweise zu verschiedenen Zeiten benutzen, so können Sie ein Menü-Programm für diese Programme schreiben und dieses Menü mit AUTOSTART automatisch auf dem Bildschirm erscheinen lassen, sobald der Computer eingeschaltet ist.

Dieses System verdeckt die üblichen Schritte für das Einladen und die Ausführung eines Programms und vereinfacht Routinetätigkeiten. Wenn Sie AUTOSTART auf diese Weise benutzen, ist Ihr Programm auch für Laien einfacher zu gebrauchen.

2. Sie können AUTOSTART für die automatische Ausführung eines speziellen Programms auf einer Diskette verwenden, so daß Computer-Neulinge nur Ihre Programmdiskette einlegen sowie die Diskettenstation und den Computer einschalten müssen, um mit Ihrem Programm zu beginnen.

3. Sie können AUTOSTART für Initialisierungsroutinen (wie z.B. Farb- oder Randeinstellungen der Screens ändern oder Peripheriegeräte initialisieren) vornehmen.

4. Sie können andere auf der Diskette befindliche Dateien vor dem Zugriff der User schützen, wenn Sie POKE Befehle in Ihr Basic-Programm einfügen, um die BREAK Taste zu blockieren und um die Funktion der SYSTEM RESET Taste abzuändern, so daß das Drücken dieser Taste einen Cold Start verursacht . Sie können AUTOSTART anschließend benutzen, um Ihr Programm automatisch laufen zu lassen. Das Musterbeispiel am Ende dieser Anleitung zeigt eine Methode, diesen Schutz einzufügen.

6.2 GEBRAUCH VON AUTOSTART

Es erscheint folgende Anzeige auf dem Bildschirm:


```
AUTORUN.SYS-ERZEUGER FUER BASIC
DOS VERSION 2.X

WELCHER BEFEHL ?
```

Geben Sie entweder den RUN oder LOAD Befehl ein, um das BASIC-Programm aufzurufen, das Sie automatisch ausgeführt haben möchten, und drücken Sie die RETURN Taste. Wahrscheinlich werden Sie für die meisten Anwendungen eher den RUN Befehl als den LOAD Befehl benutzen wollen. Um z.B. ein Programm namens CHECKING.BAS automatisch einzuladen, beantworten Sie die Anzeige mit RUN "D:CHECKING.BAS"<RETURN>. (Wichtige Anmerkung: Beachten Sie, daß Sie den Befehl auch mit Anführungszeichen beenden. Es würde sonst ein ERROR 170 -- Datei nicht gefunden -- Fehler auf dem Bildschirm erscheinen, wenn Sie versuchen, das Programm sich selbst ausführen zu lassen.
Ihr Befehl kann 128 Zeichen lang sein.

Nachdem Sie obige Information eingegeben haben, erzeugt AUTOSTART ein AUTORUN.SYS-File auf der Diskette in Diskettenstation 1. Ihre Antwort auf die obige Anzeige hängt daher teilweise davon ab, wieviele Diskettenstationen Sie benutzen.

Wenn Sie eine Diskettenstation benutzen:
Wenn Sie nur eine Diskettenstation benutzen, muß das AUTORUN.SYS-File auf der gleichen Diskette sein wie das Programm, das automatisch ablaufen soll. Entfernen Sie daher nach Einladen von AUTOSTART in das RAM die AUTOSTART Diskette und schieben Sie die Diskette mit Ihrem Programm, das automatisch laufen soll, ein. Dann geben Sie entweder den RUN oder LOAD Befehl als Antwort auf die Bildschirmanzeige ein und drücken die RETURN Taste (z.B. RUN "D:CHECKING.BAS" <RETURN>). AUTOSTART erzeugt dann das AUTORUN.SYS-File auf der Diskette mit Ihrem Programm.

Wenn Sie mehr als eine Diskettenstation haben:
Entscheiden Sie, in welche Diskettenstation Sie normalerweise Ihre Programmdiskette einlegen und geben Sie die Nummer dieser Station in Ihrem RUN oder LOAD Befehl an. Wenn Sie beispielsweise Ihre Programmdiskette immer in Station 2 einlegen, dann geben Sie den Anfangsbuchstaben und die Nummer vor dem Dateinamen ein (z.B. RUN "D2:CHECKING.BAS" <RETURN>). AUTOSTART erzeugt dann das AUTORUN.SYS-File in Diskettenstation 1, aber Ihr Befehl gibt an, welche Station das Programm enthält, das automatisch ausgeführt werden soll.

ACHTUNG! Da AUTOSTART ein AUTORUN.SYS-File erzeugt, dürfen Sie kein anderes AUTORUN.SYS-File auf der Diskette in Station 1 haben. Wenn auf dieser Diskette ein AUTORUN.SYS-File ist, wird sie von AUTOSTART mit dem neuen AUTORUN.SYS-File überschrieben.

6.3 AUTOMATISCHES PROGRAMM-LADEN

Sobald Sie AUTOSTART einmal benutzt haben, um das Programm zu bestimmen, das nach dem Einschalten von selbst ausgeführt werden soll, brauchen Sie es für diese spezielle Anwendung nicht mehr einsetzen.

Von da an schieben Sie einfach die Diskette mit dem AUTORUN.SYS-File in Station Nr. 1 (und Ihre Programmdiskette in die spezifische Station, wenn Ihr Programm sich auf einer anderen Diskette befindet) und schalten den Computer an. Das Programm wird automatisch ausgeführt.

Um das automatische Programmladen zu stoppen, können Sie entweder das AUTORUN.SYS-File löschen, indem Sie die DOS Option D (DELETE FILE) benutzen, oder Sie lassen AUTOSTART noch einmal laufen und geben eine andere Datei an, die von der gleichen Diskette geladen werden soll. Das jetzt geschaffene AUTORUN.SYS-File wird das vorige ersetzen.

6.4 EXPERIMENTIEREN SIE MIT AUTOSTART

Sie können auch andere BASIC-Befehle mit AUTOSTART verwenden. Da BASIC an einigen Stellen wieder die Kontrolle übernimmt, sind die Ergebnisse nicht immer vorhersehbar. Experimentieren Sie, um herauszufinden, was Sie noch alles mit AUTOSTART anfangen können.

6.5 MUSTERANWENDUNG

Nehmen wir an, Sie wollen jedesmal, wenn Sie Ihrem Computer eine spezielle Diskette einladen, folgende Dinge tun:

1. Die BREAK-Taste blockieren

2. Die Funktion der SYSTEM RESET Taste abändern, um damit einen Cold Start zu erzeugen

3. Ein Programm einladen und starten (das Programm fragt den User nach Eingabe einer Nummer ; anschließend erzeugt es eben die Anzahl von Zufallsnoten, jede Note für eine beliebige unbestimmte Zeitdauer)

Sie können AUTOSTART benutzen, um dieses Programm automatisch ausführen zu lassen. Das Programm selbst sieht wie folgt aus:


```
10 REM BREAK TASTE BLOCKIEREN
20 POKE 16,64:POKE 53774,64
30 REM KALTSTART BEI SYSTEM RESET
40 POKE 580,1
50 REM *** PROGRAMM TON ***
60 REM FALSCHE EINGABE --) ZEILE 70
70 TRAP 70
80 ? "*      GEBEN SIE EINE ZAHL EIN ";
90 INPUT ZAHL
100 REM GENAU 'ZAHL' TOENE ERZEUGEN
110 FOR SCHLEIFE=1 TO ZAHL
120 REM TON ERZEUGEN
130 SOUND 0,INT(RND(0)*255)+1,10,12
140 REM LAENGE DES TONES
150 ZEIT=INT(RND(0)*40)+1
160 FOR LAENGE=1 TO ZEIT:NEXT LAENGE
170 REM NAECHSTER TON
180 NEXT SCHLEIFE
190 REM TON ABSCHALTEN
200 SOUND 0,0,0,0
210 REM NAECHSTE EINGABE
220 GOTO 70
```

(Das Programm befindet sich unter dem Namen NOTES.BAS auf dieser Diskette.)

(Beachten Sie, daß der Computer die BREAK-Taste nach jeder seriellen I/O-Operation oder nach Änderung der Graphik-Modi wieder in Gang setzt; daher müssen Sie die POKE-Befehle an den entsprechenden Stellen Ihres Programms einfügen.)

Sie sichern dieses Programm auf der Diskette in Diskettenstation Nr. 1 ab und nennen die Datei NOTES.BAS. Der Befehl dazu lautet SAVE "D:NOTES.BAS". Als nächstes laden Sie das AUTOSTART-Programm wie eingangs beschrieben und beantworten die Bildschirmanzeige wie folgt:

```
AUTORUN.SYS ERZEUGER FUER BASIC
DOS VERSION 2.X

WELCHER BEFEHL ? RUN"D:NOTES.BAS" <RETURN> EINGEBEN
```

AUTOSTART erzeugt dann eine AUTORUN.SYS Datei auf der Diskette, welche dann automatisch die Befehle in Ihrer Datei NOTES.BAS ausführt. Das heißt, jedesmal wenn Sie die Diskette von der Diskettenstation Nr. 1 in Ihrem Computer laden, wird das Programm NOTES.BAS automatisch ausgeführt.

ATARI INSIDE

ATARI®

A Warner Communications Company

ATARI-Elektronik Vertriebsgesellschaft mbH
Postfach 60 01 69 · Bebelallee 10 · 2000 Hamburg 60

Jegliche Rechte vorbehalten.
Vermietung, Verleih, Vervielfältigung
und öffentliche Aufführung verboten.

ATARIINSIDE