

Dr. phil. Ingmar Thilo

# 6502-Texteditor

Softbus erleichtert Systemanpassung

Das 6502-Softbus-Programm ist eine fast wortgetreue Übersetzung des 6800-Texteditors aus mc 1983, Heft 4. Es läßt sich an beliebige 6502-Systeme anpassen und auch leicht erweitern.

Der Texteditor belegt in der vorliegenden Version vergleichsweise wenig Speicherplatz. Er funktioniert aber auch schon in halber Größe und kann je nach Ansprüchen bei allen Hardware-Erweiterungen mitwachsen. Einem Softbus-Programm ist es nämlich einerlei, aus wieviel Modulen es besteht, und in welcher Reihenfolge die einzelnen Module aufeinanderfolgen. Entscheidend ist allein, daß die benötigte Funktion samt den zugehörigen Modulen irgendwo auf

dem „Bus“ zu finden ist. Wer also nur wenig Speicherplatz hat, läßt die Module für höhere Editorfunktionen einfach heraus. Sonst muß dabei nichts geändert werden. Alle Module und das aus ihnen jeweils zusammengesetzte Programm sind im Speicher frei verschiebbar.

## Systemroutinen

Die Anschlüsse an Tastatur, Anzeige und Drucker werden über die Module

A0, A1 und A3 abgewickelt. In A0 wird ein Unterprogramm aufgerufen, das auf die Zeicheneingabe durch die Tastatur wartet und mit dem ASCII-Zeichen im Akku zum Editor zurückkommt. Das „Echo“ wird vom Editor erzeugt, ist also gegebenenfalls im Systemprogramm abzuschalten. Das Modul LVIDEO mit dem Namen A1 sorgt für die Zeichenausgabe auf dem Bildschirm. Auch hier wird das ASCII-Zeichen im Akku übergeben. In A3 schließlich gilt dasselbe für die Zeichenausgabe auf Anzeige und Drucker gleichzeitig.

## Parametereingaben und Programmstart

Vor Programmstart muß die Anfangsadresse des verwendeten Textspeichers in 0A und 0B der Zero-Page abgelegt sein. Die vorletzte Adresse des Textspeichers kommt nach 0C und 0D. Die erste Adresse des Softbus-Programms muß ebenfalls als Parameter übergeben werden. Sie wird vor Programmstart in den IRQ-Vektor geladen

PROTOK	*EAEA EAEA 48EA A517 C901 D010 C617 BAEA	RAND	EO2B DOOC A502 D002 A930 8503 40EA EAEA
	48EA A230 00EA E617 68EA AAEA 68EA EAEA	ABSRET	EO2C DOOC A90C A2A1 00EA A221 00EA 40EA
HEXBYT	E030 D01C 48EA 6A6A 6A6A A231 00EA 68EA	ORDER	E018 D01C A21A 00EA A000 AA8A D10E F006
	A231 00EA A920 A2A1 00EA 40EA EAEA EAEA		C8C8 B10E D0F5 C8EA B10E AAEA 00EA EA40
HEXASC	E031 D01C 290F 18EA 6930 C93A 3004 18EA	MEM	E019 D01C A60C E400 D006 A60D E401 F00E
	6907 A2A1 00EA 40EA EAEA EAEA EAEA EAEA		A000 A210 00EA 9100 A21D 00EA EAEA 40EA
INCACT	E008 DOOC E600 D002 E601 A600 40EA EAEA	TAB	E025 D02C A000 B917 00EA C502 F014 1004
DECACT	E009 DOOC A600 D002 C601 C600 40EA EAEA		*C8EA 10F2 A920 A219 00EA A001 B100 C903
LPRVID	EOA3 D01C 4848 2040 02EA 68EA A610 E001		DOE2 40EA EAEA EAEA EAEA EAEA EAEA EAEA
	D003 2000 0268 E002 D004 A229 00EA 40EA	WRFORM	E01E D01C A2A0 00EA C90A F00C C90D F008
LINKIN	EOA0 DOOC 2070 0240 EAEA EAEA EAEA EAEA		C904 F004 C91F 3001 40A2 1800 EAA9 0040
LVIDEO	EOA1 DOOC 2040 0240 EAEA EAEA EAEA EAEA	RESPAC	EO24 D02C A600 8604 A601 8605 A209 00EA
TABORD	E01A D04C 48A2 6086 0E20 0E00 BACA CA9A		A000 B100 C920 F004 C90A D0F0 A208 00EA
	6818 6915 850E 6885 0F90 02E6 0E68 40EA		A600 8608 A601 8609 A228 00EA 40EA EAEA
	0622 0B17 0815 0512 141D 1313 1024 0225	SUCH	E028 D04C A604 8606 A605 8607 A600 E406
	1123 1227 1526 171C 0714 0F2C 011B 191F		F026 A000 B104 D100 F016 C903 F02A E604
ACTADO	*182B 1611 0000 0000 0000 0000 0000 0000		D002 E605 A608 8600 A609 8601 18EA 90DC
	E017 D02C A90C A2A1 00EA A000 8410 8402		E604 E600 18EA 90D4 A604 8600 A605 8601
	A50B 8501 A50A 8500 A904 9100 A208 00EA	LESABS	A21D 00EA A215 00EA 40EA EAEA EAEA EAEA
	A90A 9100 A208 00EA 9100 A208 00EA 40EA		E022 D01C A21D 00EA C904 F010 C90A D0F4
INSERT	E013 D03C A21E 00EA C900 F0F8 48EA A60D		A21D 00EA C904 F002 C90A D0E8 40EA EAEA
	860F A60C 860E D002 C60F C60E A000 A60E	LESEND	E01C DOOC A21D 00EA C904 D0F8 40EA EAEA
	D002 C60F C60E B10E C8EA 910E A60F E401	WRIGHT	E01B D01C A21E 00B1 C900 F0F8 A219 00EA
	DOEA A60E E400 D0E4 68A2 1900 EA18 90C4		*18EA 90F0 EAEA EAEA EAEA EAEA EAEA EAEA
RELOC	EO10 D02C 48EA A902 C510 DOOC A000 8410	LSFORM	E01D D03C A000 B100 C904 F032 E600 D002
	A507 8501 A506 8500 A904 D100 D004 C8EA		E601 E602 48EA A2A3 00EA 68EA C90A F01A
	9100 A001 A903 910C A000 A904 910C 6840		C920 D01A A602 E403 3014 48EA A90D A2A3
PRIOF	E027 DOOC A900 8510 40EA EAEA EAEA EAEA	DELET	00EA A90A A2A3 00EA 68EA A200 8602 EA40
LSWORT	E011 D01C A21D 00EA C904 F004 C920 D0F4		E01F D02C A000 B100 C904 F022 A601 860F
	40EA EAEA EAEA EAEA EAEA EAEA EAEA EAEA		A600 860E C8EA B10E 88EA 910E E60E D002
ACTAOA	E012 D01C A21D 00EA C904 F008 C93B F006	RETUR	E020 D02C A209 00EA A209 00EA A000 B100
	A602 D0F0 40EA A210 00EA A215 00EA 40EA		C90A F00C C904 D0F0 A208 00EA A208 00EA
LSWR	*E029 D02C A602 E403 3004 C91F 301A A000	ACTRET	A208 00EA A608 8606 A609 8607 40EA EAEA
	48EA A904 D106 D004 C8EA 9106 68EA A000		E021 D02C A220 00EA A200 8602 A21D 00EA
	9106 E606 D002 E607 40EA EAEA EAEA EAEA		C904 F004 C90A D0F4 A220 00EA A91C A2A1
SUTXWR	EO14 D01C A224 00EA C903 F00A EAEA EAEA	PRION	00EA A900 8502 A210 00EA 40EA EAEA EAEA
	A202 8610 40EA A606 8600 A607 8601 40EA	TABSET	E026 DOOC A901 8510 40EA EAEA EAEA EAEA
ACTBS	E015 D01C A602 F012 A600 D002 C601 C600		E023 D01C A216 A402 F010 E8EA B500 C900
	C602 A908 A2A1 00EA 40EA A22C 00EA 40EA		*D0F8 9400 A900 9501 40EA A900 8517 40EA
CONTR	E016 D01C A930 8503 A001 A903 910C A000		
	8417 A904 910C A217 00EA EAEA A21B D0E0		

Bild 1. Das 6502-Programm ist modular strukturiert und im Speicher frei verschiebbar. Die Systemanschlüsse sind unterstrichen und bedeuten von oben nach unten: Anzeige, Tastatur, Drucker, Anzeige

und sorgt hier dafür, daß der richtige Softbus aufgerufen wird. Über diese Schaltstelle können also auch mehrere „Busse“ verzahnt und abwechselnd betrieben werden. Bei Benutzung nur eines Busses kann man die Anfangsadresse direkt in den Vektor laden.

Gestartet wird mit dem Modul CONTR entweder durch Namensaufruf und Sprung auf den Bus, also A216 00EA, oder durch direkten Sprung auf die erste Adresse hinter dem Modulkopf, also auf den Befehl A930 in der dritten Spalte (Bild 1). Ist das Programm z. B. unverändert in den Speicherbereich 0300 aufwärts geladen, so ist das die Adresse 0574.

### Debugging mit dem Softbus-Protokoll

Die Unterprogramme PROTOK mit HEX-BYT und HEXASC in den ersten sechs Zeilen des Programms haben die Aufgabe, auf Wunsch den Bus zu kontrollieren und alle Namensaufrufe auf dem Bildschirm zu protokollieren. Dabei entstehen dynamische Flußdiagramme, die den realen Programmablauf unmittelbar dokumentieren. Der Programmablauf selbst wird dadurch nicht beeinträchtigt. Eingeschaltet wird dieses Programm durch eine Eins in Zelle hex 17 der Zero-page.

Bei bereits funktionierendem Editor wird zum Einschalten der Softbus-Protokollierung der Cursor aus der Nullposition eine Stelle nach rechts bewegt.

Drückt man jetzt auf die Tabulatorsetzta- ste, wird die Eins in die Tabulatorzelle 17H geschrieben, und auf dem Bild-

schirm erscheint bereits das Protokoll der auf diesen Tastendruck folgenden Programmschritte. Alle weiteren Tastendrucke werden ebenso protokolliert.

Jedes Protokoll endet mit A0, da jedes Einzelprogramm am Schluß über das Modul A0 die nächste Tasteneingabe anfordert. Zurückgeschaltet wird durch Tabulatorsetzen bei Cursorstellung Null.

Wie wertvoll diese Softbus-Protokollierung für die Fehlererkennung ist, braucht wohl nicht eigens betont zu werden. Bei Programmausbrüchen ist immer das zuletzt im Protokoll aufgeführte Modul oder die Busstrecke zu diesem Modul kaputt. Kehrt das Programm auf Umwegen zurück, genügt ein Vergleich von Ist- und Soll-Protokoll. Softbus-Programme lassen sich also nicht nur in vergleichsweise kurzer Zeit entwickeln, sondern auch in entsprechend kurzer Zeit austesten und korrigieren.

Da sich das Testprogramm nicht selbst testet, muß man die ersten sechs Zeilen sehr sorgfältig eintippen und bei auftretenden Fehlern zuerst überprüfen. Sind sie in Ordnung, können die folgenden Module auch einzeln protokolliert werden. Die Prozedur dazu: Eins nach 17H, Modulname nach X und Sprung auf den Bus, anschließend Rücksprung in das System. Beispiel: A2 20, 00 EA, 4C 00 1C.

### Und los geht es mit dem Schreiben

Nach dem Start sitzt man vor dem leeren Bildschirm wie vor einem leeren Blatt Papier. Geschrieben wird wie auf der Schreibmaschine. Was darüber hinaus alles möglich ist, zeigt Bild 2 mit den

Editorkommandos. Die Beschreibung dazu ist in [1] zu finden. Etwas anders geht nur das Setzen des Tabulators: Er wird hier immer an der Stelle gesetzt, an der sich der Cursor gerade befindet, also wie beim Einschalten der Softbus-Protokollierung bereits gezeigt. Der Rand wird davon getrennt, aber sonst ähnlich mit dem Randsetzer gesetzt. Soll er über den alten Rand nach rechts hinausgehen, befördert man den Cursor mit einer Buchstabenfolge dorthin. Der Text kann sofort im neuen Format ausgedruckt werden. Ein Nachbessern von Trennungen ist nur erforderlich, wenn vorher irgendwo die automatische Trennung nicht abgewartet werden konnte.

### Schreiben mit Textbausteinen

Um das Arbeiten mit Textbausteinen möglichst effektiv zu machen, benützt der Editor eine assoziative Methode, die das eigene „Kramen im Gedächtnis“ nachahmt. Wollte man z. B. hier mit einem vorfabrizierten Baustein aus dem Textvorrat fortfahren, müßte man so verfahren:

Sti (Druck auf die Taste „Textkombination“; aus „Sti“ wird „Stic“ zum Zeichen, daß ein Wort gefunden ist; Druck auf die Taste „Abschnitt zurück“, und der folgende Abschnitt wird an die Stelle von „Stic“, hier also an die Stelle dieses Abschnitts geschrieben.)

Auf das Kommando „Textkombination“ sucht der Editor im folgenden Textspeicher nach dem begonnenen Stichwort und schaltet bei Erfolg auf automatisches Weiterschreiben. Alle Lesekommandos verwandeln sich dadurch zugleich in Schreibkommandos. Mit „Buchstabe“, „Wort“, „Zeile lesen“ usw. wird einfach weitergeschrieben, mit „Abschnitt zurück“ dagegen erscheint der gesamte Abschnitt, in dem das Stichwort steht. Der Cursor bleibt dann am

Taste	hex	ASCII	Name	Assembl.	Funktion
CTRL K	0B	VT	17	ACTADO	Alles von vorn
CTRL F	06	ACK	22	LESABS	Abschnitt lesen
CTRL E	05	ENQ	12	ACTAOA	Zeile lesen
CTRL T	14	DC4	1D	LSFORM	Buchstaben lesen
CTRL V	16	SYN	11	LSWORT	Lesen bis Zwischenraum
CTRL W	17	ETB	1C	LESEND	Bis Textende lesen
CTRL A	01	SOH	1B	WRIGHT	Überschreiben
CTRL S	13	DC3	13	INSERT	Schreiben mit Verschieben
CTRL H	08	BS	15	ACTBS	Cursor zurück
CTRL Y	19	EM	1F	DELET	Radieren
CTRL O	0F	S1	2C	ABSRET	Abschnitt zurück
CTRL G	07	BEL	14	SUTXWR	Textkombination
CTRL P	10	DLE	24	RESPAC	Cursorsprung
CTRL B	02	STX	25	TAB	Tabulator
CTRL Q	11	DC1	23	TABSET	Tabulator setzen 1)
CTRL U	15	NAK	26	PRION	Drucker ein
CTRL R	12	DC2	27	PRIOF	Drucker aus
CTRL X	18	CAN	2B	RAND	Rand setzen 2)

1) Zurücksetzen durch Setzen in Cursorstellung 0.  
Durch Setzen in Cursorstellung 1 wird die Softbus-Protokollierung eingeschaltet.  
2) Normalrand durch Setzen in Cursorstellung 0.

Bild 2. Die Editorkommandos

Adresse	hauptsächliche Verwendung
00 und 01	aktuelle Adresse
02	Zeile
03	Rand
04 und 05	aktuelle Adresse beim Suchen
06 und 07	aktuelle Adresse beim automatischen Schreiben
08 und 09	Retten der aktuellen Adresse
0A und 0B	Textspeicher: Anfangsadresse
0C und 0D	Textspeicher: Endadresse-1
0E und 0F	Adressenvariable für (IND),Y
10	Status-Schalter
11 bis 16	frei für Erweiterungen
17 ff.	Tabulatorstops nach Bedarf

Bild 3. Die Zero-Page-Belegung

Anfang des Abschnitts stehen. Paßt der gefundene Baustein, drückt man auf „Abschnitt lesen“ und schreibt den nächsten Abschnitt selbst weiter. Paßt der gefundene Baustein nicht, wiederholt man den Vorgang mit einem anderen Stichwort.

Am Anfang sollten die Tasten „Alles von vorn“ und „bis Textende lesen“ immer wieder zur Kontrolle benutzt werden. Der Text muß vorher, am besten gleich vor Schreibbeginn, einmal mit CTRL D abgeschlossen werden.

### Erweiterungen sehr gefragt

Ein Softbus-Programm ist leicht in andere CPU-Sprachen zu übersetzen. Wer hier neue oder verbesserte Module produziert, kann in jedem Fall „international“ mitreden. Auch sollte man sich die Mühe machen, irgendwo herumliegende Programmbrocken in Modulform zu packen und dadurch für alle zu neuem Leben zu erwecken. Für die Namensgebung gilt folgende Empfehlung: Hilfsmodule mit Pseudobefehlen und Programmen, die den Bus bedienen, erhalten Namen zwischen 00 und 0F. Normale Programm-Module bekommen Namen zwischen 10 und 9F. Die Namen A0 bis BF sind für Verbindungsmodule zu anderen Programmen reserviert, und ganze Busse werden mit Namen zwischen C0 und FF benannt.

Die Belegung der Zero-Page zeigt Bild 3. Die Zellen hex 11 bis 16 sind für Erweiterungen freigehalten. Wer mehr Platz benötigt, muß die Tabulatorstops begrenzen und von der Grenze aufwärts anbauen. Die ASCII-Controlzeichen stehen mit den Namen, die sie aufrufen, in der Tabelle von TABORD. Reichen die freien Plätze für Erweiterungen nicht aus, kann das Modul um eine Zeile oder mehr verlängert werden. Am Schluß der Tabelle muß aber einmal 0000 bleiben. Wird die Zeilenanzahl eines Moduls verändert, darf man nicht vergessen, die zugehörige Angabe im Modulkopf zu korrigieren. Sie steht an der siebenten Stelle des achtstelligen Modulkopfes und ist immer gleich der Zeilenzahl minus 1 in hexadezimaler Schreibweise. Verlängert man also TABORD um eine Zeile, muß der Modulkopf E01A D04C zu E01A D05C korrigiert werden.

### Literatur

[1] Thilo, Dr. I.: 6800-Texteditor für Schreifaule. mc 1983, Heft 4.



CP/M-80, CP/M-68K und optional auch MS-DOS und CP/M-86 schluckt der M68P als Betriebssystem

## Sord mit 68000

Mit einer 16-Bit-CPU 68000 arbeitet der Computer M68 von Mitsui/Sord. Da in seinem Inneren aber auch ein Z80-Herz pocht, kann er auch CP/M-Programme laden und ausführen und ist damit kompatibel zum M23P (Test in mc 9/1983; auch dessen Betriebssystem ist CP/M-kompatibel). Für die 16-Bit-CPU wird CP/M-68K mitgeliefert.

Mit Arbeitsspeicher wurde nicht gezeigt:

256 KByte stehen in der Grundversion zur Verfügung, aufrüstbar ist das Gerät bis 1 MByte. Wer MS-DOS oder CP/M-86 als Betriebssystem fahren will, kann auch eine „Softcard“ mit einer 8088-CPU nachrüsten. An Grafik beherrscht der M68 immerhin 16 Farben auf 640 mal 400 Punkten; dafür ist ein eigener Grafikprozessor vorhanden.

Fe.

## Spruch des Monats

„Ich wollt, ich wäre ein gescheiter Satz, oder der Redakteur käme!“

Douglas R. Hofstadter in „Spektrum der Wissenschaft“