Reinhard Seidel

Speicher-zu-Band-Disassembler

AIM-65 erzeugt assemblierfähiges Quellenprogramm

Das folgende AIM-65-Programm übersetzt Maschinencode, der im Arbeitsspeicher steht, in ein assemblierfähiges, symbolisches Quellenlisting und gibt es auf den Kassettenrecorder als Textdatei aus. Dadurch lassen sich auch sehr umfangreiche Programme neu assemblieren.

Das Programm wurde auf einen möglichst geringen Speicherbedarf ausgelegt, als Massenspeicher dient die Kassette. Pro aufgefundenem Label (Sprungziel o. ä.) werden nur zwei RAM-Bytes benötigt, so daß der AIM-65 bereits in der 4-KByte-Grundversion sehr umfangreiche Programme in einem Zug verarbeiten kann. Dies ist ein beachtlicher Vorteil gegenüber dem in mc 1981, Heft 2, veröffentlichten Programm, mit dem man nur einige hundert Befehle auf einmal disassemblieren konnte. Die neue Version ist auch in der Lage, Speicherzugriffe auf innerhalb des Programmbereichs liegende Datenfelder (Tabellen) zu erfassen und zu relativieren. Der von Datenfeldern generierte Quellencode ist ebenfalls assemblierfähig, wenn auch nicht logisch evident: Die vom Programm als Befehle decodierten Daten werden beim Assembliervorgang wieder in ihre ursprüngliche Form zurückverwandelt.

Der erzeugte Quellentext braucht nur auf zwei Ausnahmen hin überprüft zu werden, bevor er neu assembliert wird: Ladebefehle, die einen indirekten Zugriff auf Adressen innerhalb des disassemblierten Programms vorbereiten (Sprungvektoren u. ä.); und als Befehle interpretierte Datenfelder, die eine Adressierung im Programmbereich selbst vortäuschen und durch BYTE-Anweisungen ersetzt werden sollten. Beide Fälle können programmtechnisch nicht erkannt werden.

Bild 1 zeigt den Hex-Dump des Disassembler-Programms. Vor dem Start sollte man die korrekte Eingabe anhand der Prüfsummen in *Bild 2* kontrollieren. Der eingefügte Dialog sorgt für eine ausreichende Verständlichkeit der vom Programm erwarteten Eingaben und für eine Überwachung des Ablaufs. Nach dem Start bei hex 0200 fragt das Programm nach der hexadezimalen Anfangs- und Endadresse des zu bearbeitenden Ma-

```
0200 A9 00 85 11 A9 A4 85 13 A9 20 85 17 AD 00 A8 85
0210 1E 20 A3 E7 B0 FB 20 3B E8 20 10 F9 20 DD E5 20
0220 A7 E7 B0 FB AD 1C A4 85 21 AD 1D A4 85 22 20 13
0230 EA 20 71 E8 AD 13 A4 85 1C C9 54 F0 04 C9 4B D0
0240 34 A9 CF 8D 00 A8 85 1E A0 0E B9 0A 06 20 7A E9
0250 88 10 F7 C8 8C 29 A4 AE 15 A4 20 5F E9 C8 C9 0D
0260 D0 F8 A9 2C 88 F0 0C BD 38 A4 20 84 EA E8 EC 15
0270 A4 D0 F4 85 11 20 60 05 E8 8E 15 A4 8E 16 A4 20
0280 A3 E7 B0 FB A2 01 BD 1C A4 95 E3 95 E1 95 1F CA
0290 10 F4 20 3B E8 20 A7 E7 B0 FB 20 13 EA AD 1C A4

02A0 E9 0C 85 E5 AD 1D A4 E9 00 85 E6 AD 11 A4 85 1D
02B0 A9 00 8D 13 A4 8D 11 A4 20 76 05 B0 28 20 F9 F8
02C0 B0 23 20 45 04 B0 F1 A9 45 85 12 20 82 05 90 E8
02D0 20 5F 04 90 E3 A5 E1 69 01 85 E1 85 1F 90 04 E6
02E0 E2 E6 20 D0 D3 A0 FF C8 B9 01 06 91 1F C0 03 D0
02F0 F6 A2 01 C8 BD 1A A4 9D 25 A4 48 4A 4A 4A 4A 20
0300 51 EA 91 1F C8 68 29 0F 20 51 EA 91 1F CA F0 E3
0310 C8 A9 0D 91 1F 20 6B 05 20 44 EB 20 D4 04 20 76
0320 05 90 03 4C F6 03 A5 1F 38 E5 E5 A5 20 E5 E6 90
0330 1E A0 0B B9 1A 06 20 7A E9 88 10 F7 A5 E1 85 E3
0340 A5 E2 85 E4 A5 1F 85 E1 A5 20 85 E2 4C 38 04 A9
0350 00 A2 14 CA 9D 38 A4 D0 FA 8D 15 A4 20 60 05 20
0360 AF 05 20 6B 05 A9 39 85 12 20 82 05 20 62 04 A9
0370 41 85 12 B0 0E 20 C3 05 A5 1F 18 69 04 85 1F 90
0380 02 E6 20 A0 00 B1 12 91 1F F0 41 C9 3F F0 29 C8
0390 C9 20 D0 F1 A6 EA D0 0C AD 46 A4 91 1F F0 02 C8
03A0 24 88 D0 28 20 4C 04 90 29 B1 12 C9 30 91 1F A9
03B0 24 B0 01 C8 C6 12 D0 CF B9 2A 06 91 1F C8 C0 06
03C0 D0 F6 B9 38 A4 91 1F C8 C0 08 D0 F6 20 D0 04 4C
03D0 1E 03 A9 45 85 12 98 65 1F 85 1F 90 02 E6 20 20
03E0 82 05 A9 45 85 12 A9 24 90 CA 20 5F 04 20 C3 05
03F0 E6 12 A0 03 D0 8F A0 FF C8 B9 FB 05 91 1F C0 05
```

Bild 1. Hex-Dump des Speicher-zu-Band-Disassemblers für den AIM-65. Die Startadresse ist hex 0200; vor dem Start empfiehlt sich allerdings, die korrekte Eingabe mit Hilfe von Bild 2 zu überprüfen

```
0400 D0 F6 20 D4 04 A0 00 A5 E3 48 A5 E4 48 B1 E1 91
0410 E3 E6 E1 D0 02 E6 E2 A5 E1 38 E5 1F A5 E2 E5 20
0420 B0 08 E6 E3 D0 E7 E6 E4 D0 E3 A5 E3 85 E1 A5 E4
0430 85 E2 68 85 E4 68 85 E3 A5 1D 8D 11 A4 A5 1E 8D
0440 00 A8 4C A1 E1 20 AF 05 A6 EA F0 11 CA D0 53 BD
0450 41 A4 C9 42 D0 07 BD 42 A4 C9 49 D0 45 38 60 A9
0460 00 2C A5 EA 85 18 E6 18 20 BC F8 A0 01 B1 1F 85
0470 1B 88 B1 1F 85 1A A5 E1 38 E5 E3 48 A5 E2 E5 E4

0480 85 10 F0 3F 88 B1 DF 38 E5 1B 85 19 88 B1 DF E5
0490 1A D0 27 A5 19 E5 18 B0 21 98 20 2A F9 A5 19 D0
04A0 03 68 18 60 AD 25 A4 38 E5 EA 8D 25 A4 B0 03 CE
04B0 26 A4 68 68 68 A0 00 4C B8 03 98 D0 C7 E6 E0 C6
04C0 10 10 BF 68 D0 04 38 4C C5 F8 A8 A9 00 48 F0 B4
04D0 A9 0D 91 1F 98 38 65 1F 85 1F 90 02 E6 20 A2 04
04E0 A5 1C DD 05 06 F0 04 CA 10 F8 60 8D 13 A4 A5 1F
04F0 38 E5 E1 85 1F A5 11 D0 27 A0 00 B1 E1 F0 16 20
0500 BC E9 C8 C4 1F D0 F4 A9 00 8D 13 A4 A5 E1 85 1F
0510 A5 E2 85 20 60 A9 0D 20 BC E9 20 0A E5 4C 0C 05
0520 CD 68 01 D0 D4 AD 37 A4 65 1F C9 41 90 CB A2 05
0530 BD 14 06 20 8B F1 CA 10 F7 E8 20 7B E5 A9 0D 20
0540 8B F1 A9 0D 20 8B F1 20 0A E5 A0 00 20 31 ED 20
0550 31 ED 88 D0 F7 A5 1C 8D 13 A4 20 6F E5 4C F9 04
0560 A2 0F BD 16 01 95 AD CA 10 F8 60 A2 0F B5 AD 9D
0570 16 01 CA 10 F8 60 AD 25 A4 38 E5 21 AD 26 A4 E5
0580 22 60 A0 FF C8 B1 12 20 84 EA E6 12 B1 12 20 84
0590 EA 91 1F 98 F0 EE A5 21 38 F1 1F 88 A5 22 F1 1F
05A0 90 0C C8 B1 1F ED 1A A4 88 B1 1F ED 1B A4 60 A9
05B0 01 8D 19 A4 20 2B E7 A2 13 1E 38 A4 5E 38 A4 CA
05C0 10 F7 60 A0 00 A9 41 85 14 A9 41 85 15 A9 30 85
05D0 16 20 DB F8 B0 1A 20 1D F9 20 1D F9 A5 16 E6 16
05E0 C9 39 D0 ED A5 15 E6 15 C9 5A D0 E1 E6 14 D0 D9
05F0 A0 04 B9 14 00 91 1F 88 10 F8 60 2E 45 4E 44 0D
0600 00 2A 20 3D 24 58 55 50 4B 54 3D 4B 4C 42 20 3A
0610 45 5A 49 53 20 45 4C 49 46 2E 57 4F 4C 46 52 45
0620 56 4F 20 59 52 4F 4D 45 4D 20 2E 42 59 54 20 24
```

schinencodes, z. B. FROM=8000, TO=8100. Die Eingabe kann ein- bis vierstellig erfolgen. Werden mehr als vier Stellen eingegeben, so werden nur die letzten vier berücksichtigt. Jede Eingabe ist mit der Return-Taste zu beenden. Anschließend fragt das Dialogprogramm, wohin der Assemblercode geleitet werden soll: OUT=. Zulässige Eingaben sind T (Kassette), P (eingebauter Drucker), U (vom Anwender über Sprungvektor definiert), X (keine Ausgabe) oder eine beliebige andere Taste (Editor).

```
(D)FROM=200 TO=300
7B90
(D)FROM=300 TO=400
6F02
(D)FROM=400 TO=500
8DE5
(D)FROM=500 TO=630
840E
```

Bild 2. Prüfsummen von Bild 1, errechnet mit dem Prüfprogramm aus mc 2/1981, Seite 36, bzw. mc 1982, Heft 5, Seite 55

```
(M)=0700 A2 02 BD 0C
( ) 0704 07 20 BC E9
( ) 0708 CA 10 F7 00
( ) 070C 0D 43 4D F2
(*)=200

(G)/
FROM=700    TO=70E
OUT=E
FROM=710   TO=F00
 0700 A2 LDX £02
 0702 BD LDA 070C,X
 0705 20 JSR E9BC
 0708 CA DEX
 0709 10 BPL 0702
 070B 00 BRK
 070C 0D ORA 4D43

(T)
* =$0700
=(L)
/
OUT=
* =$0700
LDX £$02
AA1 LDA AA0,X
JSR $E9BC
DEX
BPL AA1
BRK
AA0 ORA $4D43
.END
```

Bild 3. Ein kleines Beispiel: Disassemblieren in den Texteditor. Von oben nach unten:
Hexadezimaler Speicherinhalt des Beispiel-Programms; Start des Disassemblers; internes AIM-Disassembler-Format; Aufruf des Texteditors (T) und Auflisten des Inhalts (L). Die Bytefolge 0D 43 4D, ursprünglich ein Datenfeld („MC"), wurde hier als 3-Byte-ORA-Befehl interpretiert

Wurde mit T die Bandausgabe gewählt, so übernimmt das Programm die Motorsteuerung (dafür muß man den Gap-Wert in Zelle A409 vor dem Programmstart auf mindestens hex 10 setzen) und fragt nach der maximalen Block-Anzahl pro Quelltext-File: FILE-SIZE: BLK=. Die Eingabe erfolgt hexadezimal ein- oder zweistellig. Werden mehr Stellen eingegeben, so sind nur die letzten zwei relevant. Falls nur Return eingegeben wird, so nimmt das Programm den Wert 2C an, womit der AIM-Texteditor beim späteren Laden eines Files zu etwa 80 % aufgefüllt wird (4-KByte-AIM).
Schließlich fragt das Programm noch nach dem freien Arbeitsspeicherbereich für die Symboltabelle und gegebenenfalls für den Editor, wobei wieder Anfangs- und Endadresse hexadezimal einzugeben sind (FROM...TO...). Anschließend erfolgt die Programmausführung, wobei der disassemblierte Code in rascher Folge auf dem Display erscheint. Im ersten Durchlauf wird die Symboltabelle im RAM angelegt, und im zweiten wird der Quellencode an das Ausgabegerät oder den unmittelbar an die Symboltabelle anschließenden Editor-Textbereich ausgegeben, wobei gleichzeitig die zusätzlich erforderlichen Assembler-Anweisungen (Programmzähler-Initialisierung, .BYT-, .FILE-, .END-Anweisungen) eingefügt werden. Bei der Bandausgabe wird zwischen aufeinanderfolgende Files ein Zwischenraum von etwa 1,5 s Dauer eingefügt, um einen gewissen Spielraum bei späteren Korrekturen zu erhalten.
Wurde die Ausgabe auf den Editor gewählt, so wird der Quellentext zum Schluß bis zum Anfang des eingegebenen Arbeitsspeicherbereichs vorgeschoben, wobei die nun überflüssige Symboltabelle überschrieben wird. Das Programm endet mit einem Sprung in den Monitor des AIM-65, von wo aus man beispielsweise mit T den Editor aufrufen kann. *Bild 3* zeigt ein kleines Beispielprogramm, das in den Texteditor disassembliert wurde (OUT=E).
Hier schließlich noch einige Angaben über das Programm selbst. Es belegt in der Zero-Page die Adressen 0010...0022 sowie (zusammen mit dem Texteditor) 00DF...00EB. Zusätzlich werden zahlreiche AIM-Systemadressen verwendet, z. B. für die Kassettenausgabe (ein Umschreiben der Software auf andere Computer ist damit nicht sinnvoll). Das Programm kann sich natürlich auch selbst disassemblieren: Falls man es auf diese Weise in einen anderen Speicherbereich schieben möchte, muß man noch wissen, daß im Bereich 05FB...062F ein Datenfeld mit Dialogtexten steht, das beim Disassemblieren in unsinnige Befehle übersetzt wird.

Telefon-Sprachspeichersystem

Ärgern Sie sich auch oft, daß Sie den gewünschten Partner per Telefon nur nach zahlreichen Wählversuchen erreichen, weil entweder besetzt ist oder niemand abhebt? Dabei würde es oft genügen, einfach eine Nachricht zu hinterlassen und erst später eine Antwort zu erhalten.

IBM hat sich über dieses Problem Gedanken gemacht und ein Sprachspeichersystem vorgestellt, das in vorhandene Vermittlungssysteme des gleichen Herstellers nachträglich eingebaut werden kann. Benutzer können das digital arbeitende Speichersystem (es wird eine Abart der Deltamodulation verwendet) über Mehrfrequenz-Tastenapparate, wie sie in Nebenstellenanlagen schon oft üblich sind, oder, falls von einem herkömmlichen Impulswahl-Apparat aus gearbeitet wird, mit Hilfe eines taschenradiogroßen Gerätes steuern, das über Tasten und einen Lautsprecher zur Erzeugung der Codiertöne verfügt.
Die gesprochenen Nachrichten werden entweder in einer Art elektronischem Postfach gespeichert, bis sie der Empfänger abruft, oder das System ruft selbst zu einer bestimmten Zeit beim Empfänger an und übermittelt ihm die Nachricht beliebig oft. Um einen Mißbrauch zu verhindern, ist jedem Benutzer ein persönlicher Kennwort-Code zugeteilt. Die Bedienerführung erfolgt ebenfalls in natürlicher Sprache, die fest im System gespeichert ist.
Die kleinste Ausbaustufe umfaßt ein Vermittlungssystem IBM-1750 mit 100 Nebenstellen, einen Rechner IBM-Serie/1, Modell F00, mit 500 KByte Speicherkapazität und die Sprachspeicher-Software. Interessenten können sich das System in den IBM-Vertriebszentralen Düsseldorf, Frankfurt, Hamburg, München und Stuttgart vorführen lassen. Fe.