

# Rund um das INTEL-HEX-Format

Franz Fiala, N, TGM

DSK-407:HEX2BIN.C, HEXPAT.C, BIN2HEX.C

Der Schlüssel zum Programmieren von EPROMs ist das HEX-Format. Es gibt mehrere HEX-Formate, z.B. von MOTOROLA und von INTEL. In der Folge wird vom INTEL-Format ausgegangen.

Das Ergebnis des Bindens (Linken) bei der Programmentwicklung sind Binärdateien, die neben dem eigentlichen Programm auch Hinweise für den Debugger oder Lader enthalten. Dieses Format ist herstellerspezifisch und im allgemeinen nicht portierbar. Es dient ja auch im allgemeinen zum Vorbereiten des Ladens mit Programmhilfen desselben Herstellers. Es enthält auch neben den eigentlichen Ladedaten auch Symbolinformation.

Soll das Programm dagegen in ein EPROM geladen werden, müssen Geräte von Fremdherstellern benutzt werden. Um mit ihnen zu kommunizieren, hat sich das HEX-Format als quasi-Standard eingebürgert.

## Das INTEL-HEX-Format

Das "Hexadezimalformat für auf Lochstreifen aufgezeichnete Objektdateien" von Intel umfaßt zwei Hauptbereiche **Symboltabelle** und **Objektkode**. Beim Programmieren von EPROMs nutzt man lediglich den Objektkodeteil aus; der Vollständigkeit halber werden hier aber beide Verfahren dargestellt.

Eine Hexadezimaldatei (HEX-Datei) gliedert sich in eine Folge von Aufzeichnungen, die in Hexadezimalziffern im ASCII-Format festgehalten sind. So werden beispielsweise statt des Werts 1AH die beiden ASCII-Zeichen "1" und "A" aufgezeichnet. Alle Aufzeichnungen werden durch ein Erkennungszeichen eingeleitet, das bei Symboltabellen ein ASCII-Leerschritt (<SP>) und bei Objektkode ein ASCII-Doppelpunkt (":") ist. Ihm folgen die übrigen Angaben in der Form (ASCII-Zeichen sind in Anführungsstriche eingeschlossen):

### Symboltabelle

```
<SP> <Zeilennummer> <SP> <symbolischer Name> <SP> <Wert>
```

wobei das Tabellenende durch ein Dollarzeichen bezeichnet ist:

```
<SP> "$"
```

### Objektkode

```
":" <Länge> <Anfangsadresse> „00“ <Daten> <Prüfsumme>
```

mit der Endaufzeichnung:

```
":" <Länge> <Startadresse> „01“ <Prüfsumme>
```

Dabei gibt die <Länge> die Zahl der Datenbytes in der Aufzeichnung an. Sie liegt zwischen "00" und "FF" in Datenaufzeichnungen und ist "00" in der Endaufzeichnung. Die <Länge> hat immer 2 Stellen. Die <Anfangsadresse> bezeichnet mit 4 Stellen die Speicherstelle, an welche die Daten geladen werden. Die <Startadresse> ist entweder "0000" oder gibt die Stelle an, an der mit der Programmabarbeitung begonnen werden soll.

Die <Prüfsumme> schließlich, ist der negative Wert (im sogenannten Zweierkomplement) der 8-Bit-Summe aller ASCII-Werte der Aufzeichnung von der <Länge> bis zum letzten Datenbyte einschließlich. Die Summe dieser Werte plus die <Prüfsumme> muß Null ergeben, andernfalls ist ein Fehler aufgetreten. Wie die Prüfsumme korrekt gebildet wird, können Sie in den Source-Kodes der HEX-Programme nachlesen.

## Beispiel für eine HEX-Datei

```
:04A126000220000013 // 4 Bytes ab A126
:03000B0002A1004F // 3 Bytes ab 000B
:0DA10000C0E00521E52170020520D0E0320D // 13 Bytes ab A100
:10A10D00E4FFFE3EF94E8E94FD50070FEF7001EE // 16 Bytes ab A1D0
:06A11D000E80F0639001CA // 6 Bytes ab A11D
:03A12300080E822AF // 3 Bytes ab A123
:03A0000002A12B8F // 3 Bytes ab A000
:10A12B00787FE4F6D8FD90A8007F007E01E4F0A3D1 // 16 Bytes ab A12B
:0AA13B00DFCDEFA75812102A1802D // 10 Bytes ab A13B
:10A1450002A10DE493A3F8E493A34003F68001F282 // 16 Bytes ab A145
:10A1550008DF48029E493A3F85407240CC8C3331B // kontinuierlicher
:10A16500C4540F4420C8834004F456800146F6FEA // Kode
:01A12A000034 // 1 Byte ab A12A
:00000001FF // Schlussrecord
```

Die folgenden Programme erleichtern den Umgang mit HEX-Dateien. Sie sind auf den Disketten zu diesem Heft auch im Source-Kode verfügbar, daher leicht für andere Zwecke anpaßbar.

## Analyse von Hexdateien (HEX2BIN.C)

Während die meisten höherpreisigen EPROM-Programme selbstverständlich auch das INTEL-HEX-Format als Input annehmen, hat der Programmierer ALLO2 (Hergestellt in Taiwan, verschiedene Lieferfirmen) mit dem INTEL-HEX-Format so seine Schwierigkeiten und will grundsätzlich nur reine Binärdateien als Input. Zwar gibt es ein mitgeliefertes Umwandlungsprogramm, das aus einem HEX-Format das gewünschte Binärformat herstellt aber zumindest in der verfügbaren Version lief das Programm fehlerhaft.

So entstand HEX2BIN, ein Programm mit Doppelfunktion. Einerseits analysiert es eine HEX-Datei und legt eine Analysedatei an. In dieser Funktion kann es für Unterrichtszwecke verwendet werden. Zusätzlich fertigt es eine Binärdatei an, die bei der ersten belegten Adresse beginnt. Aufruf:

```
HEX2BIN name
```

wobei **name** eine Datei mit dem vollen Namen **name.HEX** ist. HEX2BIN generiert zwei Dateien: **name.HDO** (Hex-Dokumentation) und **name.BIN** (eine Binärdatei beginnend mit dem niedersten HEX-Record).

## Patchen von HEX-Dateien (HEXPAT.C)

Manchmal kann es notwendig sein, die Inhalte von HEX-Dateien nachträglich zu verändern, da an den anderen Stellen der Programmentwicklung keine Möglichkeit dazu besteht. Zum Beispiel, wenn man überhaupt nur eine HEX-Datei besitzt und diese verändern möchte oder, wenn der entstandene Code in irgendeiner Weise unbefriedigend ist.

Das Programm HEXPAT ermöglicht die Veränderung von Ladeadressen von Kodeteilen. Bei allen solchen Veränderungen ist zu beachten, daß jede Zeile durch eine Prüfsumme geschützt ist; jede Veränderung einer Zahl muß durch die Veränderung der Prüfsumme begleitet sein.

Weitergehende inhaltliche Änderungen (etwa ein HEX-Editor), können in Anlehnung an dieses Beispiel leicht vorgenommen werden. Aufruf:

```
HEXPAT name dd ss
```

wobei **name** eine Datei mit dem vollen Namen **name.HEX** ist. **ss** ist die unerwünschte Adresse (höherwertiges Byte, zweistellig) und **dd** ist die gewünschte Adresse (höherwertiges Byte, zweistellig). Es entsteht eine Datei **name.HEY**, die den gepatchten Code enthält. (Beispiel für die Anwendung siehe vorige Seite, und Beitrag Remote-Debugger)

## BIN-dateien in HEX-Dateien (BIN2HEX.C)

Wann braucht man das?

- Manchmal hat man ein EPROM und hätte gerne eine HEX-Datei davon.
- Wie das vorige Beispiel gezeigt hat, ist es etwas umständlich, eine HEX-Datei zu verändern, da man die Prüfsumme jeweils korrigieren muß. Es ist daher einfacher, die HEX-Datei mit HEX2BIN in eine Binärdatei zu verwandeln, diese mit DEBUG zu editieren und danach mit diesem Programm BIN2HEX wieder ins HEX-Format zurückzuverwandeln.

Aufruf:

```
BIN2HEX name oooo
```

wobei **name** eine Datei mit dem vollständigen Namen **name.BIN** ist und **oooo** ein Offset, der angibt, ab welcher Adresse die HEX-Datei zu bilden ist. Es entsteht eine Datei **name.HEX**, beginnend bei **oooo**.

Aus Platzgründen wird der Code zu diesen Programmen nicht dargestellt. Der Source-Kode und die ausführbare EXE-Datei kann über die Diskette PCN-DSK-407 bestellt oder über die Mailbox His Master's Voice geladen werden. □