

TI - COMPILER Version 1.0

Co 1984 by Peter Kull

Das unerlaubte Kopieren dieser Programme wird strafrechtlich
verfolgt.

Beschreibung der Arbeitsweise

Im Prinzip arbeitet dieser Compiler wie der Basic-Interpreter, der in Ihrer Computer-Konsole in einem ROM abgespeichert ist. Das heißt, daß Ihre Basicprogramme, die an sich nur aus Texten bestehen, durch einen Interpreter entschlüsselt und die jeweils benötigten Einsprungadressen zu den betreffenden Befehlsroutinen berechnet werden. Nun kommt der Hauptvorteil eines Compilers.

Während der Basic-Interpreter diese Arbeit bei einem erneuten Aufruf dieser Zeile abermals verrichten muß, wird die Code-liste, die durch den Compiler erstellt wird, als neues Programm abgespeichert. Diese Code-Liste enthält außer den Sprungbefehlen auch sämtliche Variablen-Adressen, die durch einen Basic-Interpreter ebenfalls bei jedem Aufruf neu berechnet werden müssen. Dies stellt bei umfangreichen Programmen eine wesentliche Zeitersparnis dar.

Nachdem durch den Compiler ein neues Programm erstellt wurde, kann dieses nun durch einen LOADER abgearbeitet werden.

Dieser LOADER existiert in zwei Versionen. Beide Versionen enthalten jeweils die gleichen Unterprogramme, die den entsprechenden Basic-Befehlsroutinen gleichen. Der einzige Unterschied liegt in der Art der Zahlendarstellung. Während der FP-LOADER ausschließlich mit Fließkommazahlen arbeitet, verwendet der INT-LOADER das Integer Format.

Dieses Format kann durch den Prozessor direkt verarbeitet werden, weshalb die Arbeitsgeschwindigkeit ungleich höher ist. Der Nachteil dieses Formates liegt in dem begrenzten Darstellungsbereich einer Zahl. Sie muß einen ganzzahligen Wert zwischen -32678 und 32677 haben.

Da es jedoch eine Reihe von Anwendungen gibt in denen dieses Format benutzt werden kann, besitzt der INT-LOADER große Vorteile gegenüber der langsameren FP-Version.

Dies war eine kurze Beschreibung der Arbeitsweise des Compilers und der Loader-Versionen, auf die später eingegangen wird.

Da dieses Programm aus vier sehr umfangreichen Teilen besteht, von denen sich jeweils zwei permanent im Speicher befinden, mussten bei der Programmierung des Compilers und der Loader auf zwei zwar komfortable, jedoch nicht unersetzbare Basic-befehle verzichtet werden.

Dabei handelt es sich um den DEF und den SUB-Befehl. Selbstverständlich müssen Sie auch auf sämtliche Editierbefehle, wie z.B. RUN, TRACE, ON ERROR oder BREAK sowie auf die Befehle LINK und LOAD verzichten. Außerdem können Sie nur noch ein Datei File öffnen.

Beachten müssen Sie auch folgendes:

- 1) Sollten Sie die Programmabtastung in Ihren Basicprogrammen durch die geeigneten Befehle ausgeschaltet haben, so müssen diese vor der Compilierung wieder entfernt werden.
- 2) Jedes Datenfeld muß definiert werden, auch wenn das Feld weniger als zehn Elemente hat.
- 3) Die END-Anweisung darf nur am tatsächlichen Ende des Programmes stehen, nicht jedoch mitten im Programm, da die Compilierung an dieser Stelle abbricht.

Laden des Programmes

Auf Ihrer Programmdiskette befinden sich fünf Programmfiles.
Nachfolgend eine kurze Erklärung:

LOAD Dieses Basic-Programm lädt und startet sich automatisch nach dem Anwählen der Extended-Basic Option. Dadurch wird eine DSR-Routine sowie ein Menue-Programm geladen.

CLOAD Dabei handelt es sich um die erwähnte DSR-Routine.

MENU Unter diesem Namen wurde eine vielseitig einsetzbare Disketten-Direktory abgespeichert. Diese drei Programme sowie die Loader können Sie für den eigenen Gebrauch auf Ihre Disketten kopieren.

COMPILER Dieses Programm wurde mit einem Kopierschutz versehen. Vermeiden Sie einen Kopierversuch, da es zur Programmzerstörung führen kann.

FP-LOADER Die Floating Point Version der Basic-Routinen.

INT-LOADER Die Integer Version

Menu

Nachdem durch die vorher beschriebenen Lade-Routinen dieser Programmteil geladen wurde, erscheint auf dem Bildschirm eine Wahlliste, die vier Punkte umfaßt.

LOAD:

- 1 BASIC
- 2 COMPILER
- 3 FP-LOADER
- 4 INT-LOADER

1 BASIC

Nach der Anwahl der 1. Option erscheint die Diskettendirektory. Es werden bis zu acht Programme angezeigt. Sollten Sie die nächsten acht Eintragungen sehen wollen, so drücken Sie die Taste "Proceed". Durch drücken der Taste "REDO" erscheinen wiederum die ersten acht Eintragungen.

Durch Druck auf die entsprechende Zahlentaste wird das Basicprogramm geladen und gestartet. Sollten Sie Ihr Programm nur laden wollen, so drücken Sie SHIFT und die entsprechende Taste.

Wollen Sie nach dem Erscheinen der Direktory ohne Programmwahl ins Basic zurück, so drücken Sie lediglich die Taste "ENTER".

Sie können aus dem Basic durch den Befehl CALL LINK("MENU") wieder ins Menueprogramm zurückkehren.

2 COMPILER

Es erscheint ebenfalls die Directory. Laden Sie das gewünschte Basicprogramm wie vorhin beschrieben. Daraufhin wird nach dem neuen Programmnamen (z.B. DSK1.Demo), unter dem das compilierte Programm abgespeichert werden soll, gefragt. Ihnen stehen für diese Eingabe 14 Zeichen zur Verfügung. Hinter Ihre Eingabe wird automatisch ein Stern (Asteric) angefügt, um die Unterscheidung von Compilierten- und Basic-Programmen zu erleichtern.

Daraufhin wird nach der gewünschten Höchstlänge für Zeichenketten gefragt. Übergehen Sie diese Abfrage, so wird eine Höchstlänge von 15 Zeichen angenommen. Dies betrifft nur Stringvariable-Texte, die nicht in Variable übernommen werden müssen, können beliebig lang sein.

Sie können die Arbeit des Compilers beobachten, da in der linken oberen Ecke immer die momentan bearbeitete Programmzeilennummer erscheint.

3 FP-LOADER

Wählen Sie auf der Directory das gewünschte, compilierte Programm. Es wird daraufhin ohne weitere Eingaben geladen und gestartet. Der Rücksprung während des Programmablaufes ins Menüprogramm erfolgt durch den Druck auf die CLEAR-Taste.

Aus jeder dieser Menü-Optionen können Sie durch Druck auf die BACK-Taste ins Anfangsmenü zurückkehren.

4 INT-LOADER

Die Vorgehensweise entspricht der aus Punkt 3. Beachten Sie jedoch, daß in dieser Version ausschließlich mit Integerzahlen gearbeitet wird.

Berechnungen der Standardfunktionen wie z.B. SIN oder LOG führen zum Ergebnis =0.

Fehlermeldungen

Menu	No Basic Program No Compiler Program
	Falsch gewähltes Program File oder das Basicprogramm ist zu groß.
Compiler	BAD COMMAND IN XXX Unerlaubter Befehl in Zeile XXX
	BUFFER FULL IN XXX Variablenspeicher ist voll ab Zeile XXX Wählen Sie die kleinstmögliche String-Länge.
	PROGRAMSPACE FULL IN XXX Ihr Basicprogramm ist zu groß für die Compilierung.
	WORKSPACE FULL IN XXX Arbeitsspeicher voll ab Zeile XXX Verteilen Sie die Befehle in der betreffenden Zeile auf mehrere kleinere Zeilen.
	LINE NUMBER XXX NOT FOUND Die betreffende Zeilennummer wurde im Programm nicht gefunden.

Copyright by Peter Küll Alle Rechte vorbehalten.