

Handbuch zum volksFORTH Version 3.90#

Table of Contents

- [Handbuch zum volksFORTH Version 3.90](#)
- [Prolog](#)
- [Interpreter und Compiler](#)
- [Stack](#)
- [Assembler](#)
- [Vokabular-Konzept](#)
- [FORTH-Dateien](#)
- [Warum stellen wir dieses System frei zur Verfügung ?](#)
- [Warum soll man in volksFORTH83 programmieren?](#)
- [Hinweise des Lektors](#)

(work in progress)

[english machine-translation by Google](#)

(c) 1985-2010 FORTH-Gesellschaft e.V. Bernd Pennemann. Georg Rehfeld. Dietrich Weineck Klaus Schielslek. Jörg Staben. Klaus Kohl. Carsten Strotmann

Die Autoren haben sich In diesem Handbuch um eine vollständige und akkurate Darstellung bemüht. Die Im Handbuch enthaltenen Informationen dienen jedoch allein der Produktbeschreibung und sind nicht als zugesicherte Eigenschaften im Rechtssinne aufzufassen. Etwaige Schadensersatzansprüche gegen die Autoren gleich aus welchem Rechtsgrund sind ausgeschlossen. Es wird keine Gewähr übernommen dass die angegebenen Verfahren frei von Schutzrechten Dritter sind.

Unser Dank gilt der gesamten FORTH-Gemeinschaft, insbesondere Charles Moore, Michael Perry und Henry Laxen

Prolog#

volksFORTH83 ist eine Sprache, die in verschiedener Hinsicht ungewöhnlich ist. Denn FORTH selbst ist nicht nur eine Sprache, sondern ein im Prinzip grenzenloses Programmiersystem. Eines der Hauptmerkmale des Programmiersystems FORTH Ist seine Modalität. Diese Modularität wird durch die kleinste Einheit eines FORTH-Systems, das WORT, gewährleistet.

In FORTH werden die Begriffe Prozedur, Routine, Programm, Definition und Befehl alle gleichbedeutend mit Wort gebraucht. FORTH besteht also, wie jede andere natürliche Sprache auch, aus Wörtern.

Diese FORTH-Worte kann man als bereits kompilierte Module betrachten, wobei immer ein Kern aus einigen hundert Worten durch eigene Worte erweitert wird. Diese Worte des Kerns sind in einem FORTH83-Standard festgelegt und stellen sicher, dass Standard-Programme ohne Änderungen auf dem jeweiligen FORTH-System lauffähig sind. Ungewöhnlich ist, dass der Programmtext des Kerns selbst ein FORTH-Programm ist, im Gegensatz zu anderen Programmiersprachen, denen ein Maschinensprache-Programm zugrunde liegt. Aus diesem Kern wird durch ein besonderes FORTH-Programm, den Meta-Compiler, das lauffähige Forth System erzeugt.

Wie fügt man nun diesem lauffähigen Kern eigene Worte hinzu? Das kompilieren der Worte wird in FORTH mit COLON ":" eingeleitet und mit SEMICOLON ";" abgeschlossen.

- : erwartet bei der Ausführung einen Namen und ordnet diesem Namen alle nachfolgenden Wörter zu.
- ; beendet diese Zuweisung von Wörtern an den Namen und stellt das neue Wort unter diesem Namen für Aufrufe bereit.

Interpreter und Compiler#

Ein klassisches FORTH-System stellt immer sowohl einen Interpreter als auch einen Compiler zur Verfügung. Nach der Einschaltmeldung oder einem Druck auf die -Taste wartet der FORTH-Interpreter mit dem FORTH-typischen "ok" auf Ihre Eingabe. Sie können ein oder mehrere Befehlswörter in eine Zeile schreiben. volksFORTH beginnt erst nach einem Druck auf die RETURN-Taste mit seiner Arbeit, indem es der Reihe nach jeden Befehl in der Eingabezeile abarbeitet. Dabei werden Befehlsörter durch Leerzeichen begrenzt. Der Delimiter (Begrenzer) für FORTH-Prozeduren ist also das Leerzeichen, womit auch schon die Syntax der Sprache FORTH beschrieben wäre.

Der Compiler eines FORTH-Systems ist also Teil der Interpretieroberfläche. Es gibt daher keinen Compiler-Lauf zur Erstellen des Programmtextes wie in anderen Compiler-Sprachen, sondern der Interpreter wird mit neuen zur Problemlösung notwendigen Worten als Anwenderprogramm abgesichert.

Auch ":" (COLON) und ";" (SEMICOLON) sind kompilierte Worte, die aber für das System den Compiler ein- und ausschalten. Da sogar die Worte, die den Compiler steuern, "normale" FORTH-Worte sind, fehlen in FORTH die in anderen Sprachen üblichen Compiler-Optionen oder Compiler-Schalter. Der FORTH-Compiler wird mit FORTH-Worten gesteuert.

Der Aufruf eines FORTH-Wortes erfolgt über seinen Namen ohne ein explizites CALL oder GOSUB. Dies führt zum FORTH-typischen Aussehen der Wortdefinitionen:

```
: <name>
  <wort1> <wort2> <wort3> ... ;
```

Die Standard-Systemantwort in FORTH ist das berühmte "ok". Ein Anforderungszeichen wie 'A>' bei DOS oder ']' beim guten APPLE 11 gibt es nicht! Das kann dazu führen, dass nach einer erfolgreichen Aktion der Bildschirm völlig leer bleibt; getreu der Devise:

Keine Nachrichten sind immer gute Nachrichten!

Und - ungewöhnlicherweise - benutzt FORTH die so genannte Postfix Notation (UPN) vergleichbar den HP-Taschenrechnern, die in machen Kreisen sehr beliebt sind. Das bedeutet, FORTH erwartet immer erst die Argumente, dann die Aktion... Statt

3 + 2 und (5 + 5) * 10

heisst es

2 3 + und 5 5 + 10 *

Da die Ausdrücke von links nach rechts ausgewertet werden, gibt es in FORTH keine Klammern.

Stack#

Ebenso ungewöhnlich ist, dass FORTH nur ausdrücklich angeforderte Aktionen ausführt: Das Ergebnis Ihrer Berechnungen bleibt solange in einem speziellen Speicherbereich, dem Stack, bis es mit einem Ausgabebefehl (meist ".") auf dann Bildschirm oder dem Drucker ausgegeben wird. Da die FORTH-Worte den Unterprogrammen und Funktionen anderer Programmiersprachen entsprechen, benötigen sie gleichfalls die Möglichkeit, Daten zur Verarbeitung zu übernehmen und das Ergebnis

abzulegen. Diese Funktion übernimmt der STACK. In FORTH werden Parameter für Prozeduren selten in Variablen abgelegt, sondern meist über den Stack übergeben.

Assembler#

Innerhalb einer FORTH-Umgebung kann man sofort in der Maschinensprache des Prozessors programmieren, ohne den Interpreter verlassen zu müssen. Assembler Definitionen sind den FORTH-Programmen gleichwertige FORTH-Worte.

Vokabular-Konzept#

Das volksFORTH verfügt über eine erweiterte Vokabular-Struktur, die von W. Ragsdale vorgeschlagen wurde. Dieses Vokabular-Konzept erlaubt das Einordnen der FORTH-Worte in logische Gruppen.

Damit können Sie notwendige Befehle bei Bedarf zuschalten und nach Gebrauch wieder wegschalten. Darüber hinaus ermöglichen die Vokabulare den Gebrauch gleicher Namen für verschiedene Worte, ohne in einen Namenskonflikt zu kommen. Eine im Ansatz ähnliche Vorgehensweise bietet das UNIT-Konzept PASCAL oder MODULA-Compiler oder die Packages in Java.

FORTH-Dateien#

FORTH verwendet oftmals besondere Dateien für seine Programme. Dies ist historisch begründet und das Erbe einer Zeit, als PORTH noch sehr oft Aufgaben des Betriebssystems übernahm. Da gab es ausschliesslich FORTH-Systeme, die den Massenspeicher vollständig selbst ohne ein DOS oder dazwischenliegendes Betriebssystem verwalteten und dafür Ihre eigenen Dateistrukturen benutzten. Diese Dateien sind so genannte Blockfiles und bestehen aus einer Aneinanderreihung von 1024 Byte grossen Blöcken. Ein solcher Block, der gerne SCREEN genannt wird, ist die Grundlage der Quelltext-Bearbeitung in FORTH. Allerdings können mit dem volks4TH auch normale Dateien bearbeitet werden, die im Dateiformat des nativen Betriebssystems (MS-DOS; Atari TOS, Apple-DOS, AMS-DOS, CP/M ...) vorliegen, sog. "Stream Files".

Generell steht hinter jeder Sprache ein bestimmtes Konzept, und nur mit Kenntnis dieses Konzeptes ist möglich, eine Sprache effizient einzusetzen. Das Sprachkonzept von FORTH wird beschrieben in dem Buch "In FORTH denken" ([Thinking Forth](#)) von Leo Brodie (Hanser Verlag).

Einen ersten Eindruck vom volksFORTH83 und von unserem Stolz darüber soll dieser Prolog vermitteln. volksFORTH83 ist ein "Open-Source"-System, bei dessen Leistungsfähigkeit sich die Frage stellt:

Warum stellen wir dieses System frei zur Verfügung ?#

Die Verbreitung, die die Sprache FORTH gefunden hat, war wesentlich an die Existenz von figFORTH geknüpft. Auch figFORTH ist ein open-source Programm (früher wurde dies Public Domain genannt, aber heute ist der Ausdruck Open-Source genauer), d.h. es darf inklusive des Quelltextes weitergegeben und kopiert werden. Trotzdem haben sich bedauerlicherweise verschiedene Anbieter die einfache Adaption des figFORTH an verschiedene Rechner sehr teuer bezahlen lassen.

Das im Jahr 1979 erschienene figFORTH ist heute nicht mehr so aktuell, weil mit der weiteren Verbreitung von Forth eine Fülle von eleganten Konzepten entstanden sind, die z.T. in den Forth83-Standard und den Forth ANSI-Standard Eingang gefunden haben. Daraufhin wurde von Laxen und Perry das F83 geschrieben und als Public Domain verbreitet. Dieses freie 83er-Standard-FORTH

für MS-DOS mit seinen zahlreichen Utilities ist recht komplex und wird auch nicht mit Handbuch geliefert.

Wir haben ein neues Forth für verschiedene Rechner entwickelt. Das Ergebnis ist das volksFORTH83, eines der besten Forth-Systeme, die es gibt. Das volksFORTH soll in die Tradition der oben genannten Systeme, insbesondere des F83, anknüpfen und die Verbreitung der Sprache FORTH fördern.

volksFORTH wurde unter dem Namen ultraFORTH zunächst für den C64 geschrieben. Nach Erscheinen der Rechner der Atari ST-Serie entschlossen wir uns, auch für sie ein volksFORTH83 zu entwickeln. Die erste ausgelieferte Version 3.7 war, was Editor und Massenspeicher betraf, noch stark an den C64 angelehnt. Sie enthielt jedoch schon einen verbesserten Tracer, die GEM-Bibliothek und die anderen Tools für den ST. Der nächste Schritt bestand in der Einbindung der Betriebssystem-Dateien. Nun konnten Quellentexte auch vom Desktop und mit anderen Utilities verarbeitet werden. Die dritte Adaption des volksFORTH entstand für die CP/M-Rechner (8080-Prozessoren), wobei speziell für den Schneider CPC auch die Grafikfähigkeit unterstützt wird. Dann wurde das volksFORTH für die weit verbreiteten Rechner der IBM PC-Serie angepasst.

In den 90er Jahren wurden Rechner mit vielen Megabyte an Hauptspeicher und Festplattenplatz zum Standard, und grafische Betriebssysteme wie Windows oder MacOS setzen sich durch.

Aber volksFORTH ist heute immer noch für Rechner mit begrenzten Systemressourcen interessant, sei es auf alten Homecomputern aus den 80er Jahren oder PDAs und Mobiltelefonen.

VolksForth ist in der Version 3.90 für die folgenden Rechnersysteme verfügbar:

- 6502 CPU
 - Commodore C64
 - Commodore C16
 - Commodore Plus4
 - Atari XL/XE
 - Apple I
 - Apple II
- Z80 CPU
 - CP/M
 - Amstrad/Schneider CPC unter AMS-DOS
 - Amstrad NC100
 - Sinclair Research Z88
- 8088 CPU (Intel / AMD)
 - MS-DOS (in einer DOS-BOX auch unter Windows, Linux, OS/2, MacOS)
- 68000 CPU
 - Atari ST

Warum soll man in volksFORTH83 programmieren?#

Das volksFORTH83 ist ein ausgesprochen leistungsfähiges und kompaktes Werkzeug. Durch resistente Runtime-Library, Compiler, Editor und Debugger sind die ermüdenden ECLG-Zyklen ("Edit, Compile, Link and Go") überflüssig. Der Code wird Modul für Modul entwickelt, kompiliert und getestet. Der integrierte Debugger ist die perfekte Testumgebung für Forth-Worte. Es gibt keine riesigen Hexdumps oder Assemblerlistings, die kaum Ähnlichkeit mit dem Quellentext haben. Ein anderer wichtiger Aspekt ist das Multitasking. So wie man ein Programm in einzelne, unabhängige Module oder Worte aufteilt, so sollte man es auch in einzelne, unabhängige Prozesse aufteilen können. Das ist in den meisten Sprachen nicht möglich. Das volksFORTH83 besitzt einen einfachen, aber leistungsfähigen Multitasker.

Schliesslich besitzt das volksFORTH83 noch eine Fülle von Details, über die andere FORTH-Systeme nicht verfügen:

- Es benutzt an vielen Stellen Vektoren und sog. deferred Worte, die eine einfache Umgestaltung des Systems für verschiedene Gerätekonfigurationen ermöglichen.
- Es besitzt einen Heap für "namenlose" Worte oder für Code, der nur zeitweilig benötigt wird. Der Blockmechanismus ist so schnell, dass er auch sinnvoll für die Bearbeitung grosser Datenmengen, die in Files vorliegen, eingesetzt werden kann.
- Das System umfasst Tracer, Decompiler, Multitasker, Assembler, Editor, Printer-Interface ... Das volksFORTH83 erzeugt, verglichen mit anderen FORTH-Systemen, relativ schnellen Code, der aber langsamer als der anderer Compilersprachen ist.

Mit diesem Handbuch soll die Unterstützung des volksFORTH83 noch nicht beendet sein. Die FORTH Gesellschaft e.V., ein gemeinnütziger Verein, bietet dafür die Plattform. Sie gibt die Vereins-FORTH-Zeitschrift "VIERTE DIMENSION" heraus. Die Forth Gesellschaft kann über die Webseite <http://www.forth-ev.de> erreicht werden.

Hinweise des Lektors#

Diesem Handbuch zum volksFORTH83 ist sowohl als Nachschlagewerk als auch als Lehrbuch für FORTH (speziell volksFORTH) gedacht. Deshalb handelt es sich nicht, wie bei den anderen volksFORTH-Handbüchern, um eine Auflistung des Vokabulars. Statt dessen wird mit ausführlichen Beschreibungen und Programmbeispielen in vielen Kapiteln die Möglichkeiten des FORTH-Systems erklärt. Ergänzt werden die einzelnen Kapitel jeweils um Wortbeschreibungen der darin vorkommenden Befehle (Glossar). Zur Unterscheidung von Beschreibung, FORTH-Worten, Programm-Eingaben und -ausgaben wird mit unterschiedlichen Schrifttypen gearbeitet:

Beschreibungen erfolgen in Proportionalischrift mit Randausgleich. **FORTH-Befehle** werden im Text durch Fettschrift hervorgehoben. *Eingaben* und *Programm Listings* verwenden eine nichtproportionale Schriftart. Ausgaben des FORTH-Interpreter/Compiler sind unterstrichen.

Weiter mit [Kapitel 2](#).