

# VIERTE DIMENSION

8/2

8. Jahrgang 1992 2. Quartal DM 7,50



Neue und alte Köpfe

STUTTGART:  
Forth Diaspora?

MOERS:  
wie gewohnt  
forth-aktiv

**Neu:**

Das Direktorium  
Die Redaktion

**Einstieg:**

Forth für Anfänger

**Rostock:**

Forth-Tagung '92  
Jahresversammlung der Forth-Gesellschaft

**Boxen:**

Forth im Programmierwettbewerb entthront!

**Mathe:**

Festkommapaket

**F-PC** Centronics mit F-PC steuern  
Light-Show mit F-PC zeigen  
Forth leicht mit F-PC lernen

**FORTH  
MAGAZIN**

Organ der FORTH-Gesellschaft e.V.

# SMAN - Der Software-Manager

Probleme mit der Verwaltung großer Mengen Quelltext?  
Rasches Finden von Quelltext-Modulen nicht möglich?  
Zusammenfügen von Modulen umständlich?  
Keine einheitliche Umgebung für verschiedene Compiler?

**SMAN kann's !**

DFF-Team, Frank Stüss  
An der Turnhalle 6  
6369 Schöneck 2  
Tel.: 06187-91503  
FAX: 06187-91725

## DIGI CAM

DIGI CAM steht für "Digitale Camera", sie ist akkubetrieben und in der Lage 32 Schwarz/Weiß-Bilder aufzunehmen, bevor sie mit einem Adapter an einen (Atari, Amiga), Apple-Macintosh oder PC angeschlossen wird. - Die Auflösung der Digi Cam beträgt 376 x 240 Bildpunkte, 256 Grauwerte/Bildpunkt. - Optik: Verschlusszeit automatisch, 1/30 bis 1/1000 Sekunde, Belichtungs- & Blitzautomatik. Das Universalsobjektiv der Digi Cam ist für Entfernungen ab 1m geeignet. - Abmessungen: 17,0x8,1x3,0 cm (Taschenformat)  
Temperaturbereich: 0 bis 30 Celsius, Gewicht: 263 Gramm - Eingangssignale: Ladestrom für Akku, ferngesteuerter Auslöser. - Ausgangssignale: Serielle Hochgeschwindigkeits-Schnittstelle RS-232C/RS-423), Lautsprecher. - Sonstiges: Stativgewinde, Schutzring um Auslöser und Objektiv, Griffmulden.

Hardware-Voraussetzungen: PC oder Apple-Macintosh (Atari und Amiga in Vorbereitung) mit serieller Schnittstelle. Am PC ist eine VGA-Grafikkarte und ein entsprechender Bildschirm erforderlich. - Software: Die Digi Cam wird mit Software ausgeliefert, die es erlaubt, die Bilder aus der Kamera zu übertragen und am Bildschirm des PCs oder Apple-Macintosh anzuzeigen. Die Bildspeicherung im TIFF-Format ist möglich. - Lieferumfang: Kamera, Adapter, Netzteil, serielles Übertragungskabel für PC, Datenkabel für Apple Macintosh, Gewindering für Zusatzlinsen, Software für Apple Macintosh und PC sowie Bedienungsanleitung. - DM 2498,-

## FOTOPLOTTER

Die Herstellung von Reprofilmen bis DIN A3 ist einfach, bequem, schnell und preiswert mit dem Lightpen-FOTO-Plotter SPL-450. Das Gerät ist für alle HP-GL-Code erzeugende Programme einsetzbar! Linotype o.ä. Filmbelichter sind nun nicht mehr erforderlich. Erstellen nun auch Sie Ihre Technischen Repro-Vorlagen in kurzer Zeit selbst! Komplett Erstausrüstung: 2 Light-, 8 Farbpens, 25 Filme, Entwicklungsmat. Rotlichtlichtl. ab DM 3499,-

## Leiterplattenentflechtung

Feinleiter-, Normal-, SMD-Layouts, Multilayertechnik. Wir kopieren auch Ihre Leiterplatten!  
Leiterplattenentflechtungs Programme PCB-layout für Atari ST  
PCB-layout: DM 199,-, PCB-layout: Großbildschirm DM 298,-, PCB-layout plus: Autorouter DM 348,-, PCB-layout professional: wie Plus jedoch Großbildschirm DM 698,-, PCB-NC: Platinenfräsen mit isert-NC-Maschine DM 1498,00. Fräs- & Plottservice für PCB-layout.

## Atari Erweiterungen

1Mbyte bis 14Mbyte ab DM 170,-, ADspeed 16Mhz für alle ST's DM 598,-, TOS 1.04 auf 70ns Eprons nur DM 216,-, TOS 1.04 DM 89,-, ATonce PC Erw. incl. Einb. DM 498,-, NVDI-Software macht aus Ihrem ST fast einen T1! DM 99,-, sämtliche Teile werden von uns eingebaut  
Wir reparieren auch Ihren Atari ST!

Layout-Service-Kiel

Eckernförder Str. 83, 2300 Kiel 1, Tel: 0431-180975, Fax 17080

Info anfordern !

Schneller programmieren mit

embedded  
65C02  
FORTH

Ideal für :

Steuerungen, Prüfgeräte, Meßgeräte



089 / 8418317

R. Deliano

Steinbergstr. 37

8034 Germering



## IMPRESSUM

### Name der Zeitschrift

VIERTE DIMENSION  
FORTH MAGAZIN  
Organ der Forth-Gesellschaft e.V.

### Herausgeber

Forth-Gesellschaft e.V.  
W-8044 Unterschleißheim

### Redaktionsleitung

Rolf Kretzschmar (rk), (verantwortlich)  
Rote Gasse 7, W-5112 Baesweiler  
(Redaktionsadresse)  
Tel/Fax: 02401-88891

### Redaktion

Arndt Klingelberg (akg), Alsdorf  
Tel.: 02404-61648 Fax: 02404-63039  
Klaus-Peter Schleisiek (kps), Aachen  
Tel/Fax: 0241-873462

### Layout, Satz, Herstellung

ORGA Sport, Rilkestr. 8, W-5110 Alsdorf  
Tel/Fax: 02404-61425

### Grafik, Illustration, Layout

Rolf Kretzschmar (rolf)

### Anzeigenverwaltung

Forth-Büro, Postfach 1110  
W-8044 Unterschleißheim  
Tel.: 089-3173784 oder  
Forth-Mailbox Tel. 089-8714548 8N1  
"Konferenz Vierte Dimension"

### Redaktionsschluß

Erste Woche im Feb., Mai, Aug., Nov.

### Erscheinungsweise

vierteljährlich

### Auflage

1000

### Preis

Einzelheft DM 7,50, Abonnementpreis  
DM 40,-, bei Auslandsadresse DM 45,-  
inklusive Versandkosten

### Manuskripte und Rechte

Berücksichtigt werden alle eingesandten Manuskripte von Mitgliedern und Nichtmitgliedern. Leserbriefe können ohne Rücksprache gekürzt wiedergegeben werden. Beiträge der Redaktion sind vom jeweiligen Redakteur mit seinem Kürzel (s.o.) gekennzeichnet. Für die mit dem Namen des Verfassers gekennzeichneten Beiträge übernimmt die Redaktion lediglich die presserechtliche Verantwortung. Die in diesem Magazin veröffentlichten Beiträge sind urheberrechtlich geschützt. Übersetzung, Vervielfältigung, Nachdruck sowie Speicherung auf beliebige Medien ist auszugsweise nur mit genauer Quellenangabe erlaubt. Die eingereichten Beiträge müssen frei von Ansprüchen Dritter sein. Veröffentlichte Programme gehen - soweit nicht anders vermerkt - in die Public Domain über. Für Fehler im Text, in Schaltbildern, Aufbauskizzen etc., die zum Nichtfunktionieren oder evtl. Schadhafwerden von Bauelementen oder Geräten führen, kann keine Haftung übernommen werden. Sämtliche Veröffentlichungen erfolgen ohne Berücksichtigung eines eventuellen Patentschutzes. Warennamen werden ohne Gewährleistung einer freien Verwendung benutzt.



## Direktorial

von Jörg Staben

Mittlerweile steuert die Forth Gesellschaft auf ihr 10-jähriges Bestehen zu und findet sich in einer Welt wieder, die sich in diesen zehn Jahren völlig verändert hat.

So umwälzend die Änderungen im politischen Bereich waren, so gravierend hat sich auch die gesamte EDV-Landschaft gewandelt.

Diese Landschaft wird kaum noch von der herausragenden Einzelleistung bestimmt, sondern große Konzerne und millionenschwere Firmen geben den Ton an. Auf überdimensionierten Rechnern läuft zentnerschwere Software, deren Schwerpunkt neben der Erfüllung der eigentlichen Aufgaben im wesentlichen in einem attraktiven Erscheinungsbild liegt.

In diesem Umfeld tut sich nicht nur ein Konzept wie Forth schwer, dessen Grundlage die Beschränkung auf das Notwendige und Überschaubare ist.

Doch während Giganten wie OS/2 in den Startlöchern stehen, wollen zusehends wieder mehr Anwender eines Programmiersystems die eigenen Ideen und Vorstellungen unbeschränkt verwirklichen können. Und dazu ist Forth nach wie vor die erste Wahl.

Dies zu vermitteln, ist unser neues Redaktionsteam in Aachen/Alsdorf/Baesweiler angetreten. Mit großem Elan und viel Enthusiasmus möchte die neue Truppe der Vierten Dimension all' denen eine Plattform bieten, die mit Freude in Forth programmieren oder auch einfach nur Spaß an dem ungewöhnlichen Sprachkonzept haben. Nach diesem erklärten Selbstverständnis soll die Vierte Dimension noch mehr als sonst den alltäglichen Umgang mit Forth genauso anschaulich zeigen wie auch das Spiel mit der Sprache selbst.

Mit einem Seitenblick darauf, daß kommerzielle Magazine wie DOS TOOLBOX oder CHIP TOOL zum Jahreswechsel 91/92 eingestellt wurden: Wenn es uns gelingt, daß die Leser der VD voll Spannung auf die nächste Ausgabe warten und nach der Lektüre erbot oder erfreut zur Diskette oder zum Telefon greifen, dann haben wir unser Ziel mehr als erreicht.



# weltweit

## Wer? Wo? Was? Wann?

bearbeitet von Ulrich Hoffmann

### Konferenzen

#### **FORML - FORth Modification Laboratory**

erstmalig 1979 in Großbritannien  
Wann/wo: seit 1980 in Asilomar Monterey Kalifornien am Wochenende nach Thanksgiving  
Thematik: Wie läßt sich Forth verbessern und verändern? Implementierungstechniken und neue Konzepte  
Organisation: FIG

#### **Rochester Forth Conference**

erstmalig seit 1981  
Wann/wo: typischerweise im Juni des Jahres an der Universität in Rochester  
Thematik: Vorstellung neuester Entwicklungen und Produkten, Forum der kommerziellen Forth-Entwickler und Händler  
Organisation: IAFR

#### **SIG Forth Conference**

erstmalig seit 1989  
Wann/wo: im ersten Quartal des Jahres typischerweise im Süden der USA  
Thematik: Vorstellung neuester Entwicklungen und Produkten, Forum der professionellen Forth-Programmierer  
Organisation: ACM SIG FORTH

#### **EuroFORTH conference**

erstmalig 1985 als euroFORML  
Wann/wo: im Oktober des Jahres im Wechsel in Southampton(GB) und im kontinentalen Europa.  
Thematik: Vorstellung und Austausch jüngster Ergebnisse und Entwicklungen ind Produkte, Forum von Forth Entwicklern, Anwendern und Anbietern.  
Organisation: euroFORTH Org.

#### **Echtzeit Messe/Kongress**

erstmalig 1990 in Sindelfingen  
Wann/wo: ab 1993 im Juni in Karlsruhe  
Thematik: Fachkongreß über Entwicklungen im Echtzeitbereich, Fachmesse zur Vorstellung neuer Echtzeit-Produkte, Anwender, Entwickler, Anbieter, Praktiker, Theoretiker  
Organisation: DREBINGER

#### **Forthtagung der Forth Gesellschaft/Mitgliederversammlung**

erstmalig 1988 in München  
Wann/wo: typischerweise im Frühjahr des Jahres an wechselnden Orten in Deutschland

Thematik: Austausch neuester Entwicklungen im Umfeld von FORTH  
Organisation: FG

### Organisationen

#### **FIG Forth Gesellschaft e.V.**

Vereinigung von Forth Interessierten im deutschsprachigen Raum.

#### **FIG Forth Interest Group**

Amerikanische Vereinigung von Forth Interessierten,

#### **ACM SIG FORTH**

Association for Computing Machinery's Special interest Group on Forth (Fachgruppe Forth der ACM)

### Publikationen

#### **Journal of Forth Application and Research**

Wann/wo: seit 1983 vierteljährlich Hrsg: IAFR Preis: ca \$75,- / Jhrg. (Übersee)

#### **Forth Dimensions**

Wann/wo: seit 1979 zweimonatlich Publikation der FIG Hrsg: FIG Preis: ca. \$52 / Jhrg. (Übersee)

#### **ACM SIG Forth Newsletter**

Wann/wo: seit 1989 vierteljährlich Publikation der ACM SIG Forth, Hrsg: ACM SIG FORTH, Preis: ca. \$50 / Jhrg. (Übersee)

#### **Vierte Dimension**

Wann/wo: seit 1985 vierteljährlich Publikation der Forth Gesellschaft eV, Hrsg: Forth Gesellschaft, Preis: ca. DM 40 / Jhrg.

### Medien

#### **FORTHnet**

Amerikanische Verbindung mehrerer Forth spezifischer Mailboxen, Verbindung zum USEnet.

#### **GENIE Forth Forum**

Amerikanisches Diskussionforum um und über Forth, Verbindung zum USEnet.

#### **USEnet**

Gruppe comp.lang.forth Internationales Diskussionsbrett über Forth, Standardisierung, Implementationen, Bildung, Programmier-techniken und -philosophien. Verbindung mit dem amerikanischen FORTHnet, zu GENIE und zum FNEAS Filerserver, der Zugriff auf ein großes Forth File-Archiv erlaubt. Wird auch über Zerberusnetz verteilt: dazu Ansprechpartner: FG

#### **Zerberus Netz FORTHEV/FORUM**

In ganz Deutschland verteiltes Brett des Zerberus Netzes, in dem eine deutschsprachige Diskussion über Forth geführt wird. Verbindungen zum Maus-Netz und dem FIDO-Netz. FORTEV/NEWS Verteilung der USEnet Gruppe comp.lang.forth über das Zerberus-Netz. Ansprechpartner: FG

#### **Mailbox der FG: 089/8714548**

Stellt comp.lang.forth (FORTEV/NEWS) und FORTHEV/FORUM zur Verfügung, Filearchiv, Quelltexte der Vierten Dimension  
Standardisierung

**ANS ASC X3/X3J14 Technical Committee** Ziele: Erstellung eines Sprach-Standards für Forth (ANSI), Draft-Proposed-Standard ist verabschiedet

### Adressen

**FIG** Forth Gesellschaft eV, Postfach 1110, W-8044 Unterschleißheim, Tele/Fax 089-3173784, Mailbox 089-8714548

**FIG** Forth Interest Group, P.O.Box 8231, San Jose, CA 95155

**ACM SIG FORTH** Association for Computing Machinery's Special Interest Group on Forth, ACM, Inc., 11 West 42nd St., New York, NY 10036  
Tel (212)-869-7440

**IAFR** Institute for Applied Forth Research, 70 Elmwood Avenue, Rochester, NY 14611 Tel (716)-235-0168, Fax (716)-328-6426

**FIGUK** Forth Interest Group UK, 58 Woodland Way, Morden, Surrey, SM4 4DS

**EuroFORTH** Org. Kontinent. Europa: EuroFORTH -office, Uhlenhorster Weg 3, W-2000 Hamburg 76, Tel 040-2296441, Fax 040-2297205  
Großbritannien: MPE, 133 Hill Lane UK  
Southampton, SO1 5AF, (703)631441

**DREBINGER** Ludwig Drebing GmbH, Fachmessen und Kongresse, Destouchesstr. 16, W-8000 München Tel 089/333033, Fax 089/332761



<b>Impressum</b>	.....	1
<b>Direktorial</b> Anstelle eines Editorials	<i>Jörg Staben</i> .....	1
<b>Forth-weltweit</b> Wichtige Forth-Adressen und Termine	<i>Ulrich Hoffmann</i> .....	2
<b>Inhaltsverzeichnis</b>	.....	3
<b>Direktorium</b> Vorstellung des neuen Direktoriums:	<i>Staben, Beierlein, Hoffmann</i> .....	4
<b>Redaktion</b> Vorstellung der neuen VD-Redaktion	<i>Klingelberg, Kretzschmar, Schleisiek</i> .....	5
<b>Redaktorial</b> Checkliste und Hinweise für Autoren	<i>rk</i> .....	6
<b>Festkommapaket</b> Ein hilfreiches Komplettpaket. Nicht nur für LMI PC/Forth.	<i>Heinrich Hohl</i> .....	7
<b>Light-Show</b> Eine Muster-Applikation, die zeigt, wie unter F-PC und mit TCOM die Tastatur angesprochen werden kann.	<i>Jörg Staben</i> .....	10
<b>Eine Einführung in die Arbeit mit Forth</b> Diesen kleinen Grundkurs können Sie gut einem Einsteiger oder Umsteiger zusammen mit F-PC in die Hand geben.	<i>Rolf Kretzschmar</i> .....	13
<b>Tagung '92 in Rostock</b> Ein Bericht von der Forth-Tagung und der Jahresversammlung der Forth-Gesellschaft e.V.	<i>Michael Kalus</i> .....	17
<b>Programmier-Wettbewerb '92</b> Ein Bericht über den Programmierwettbewerb während der Messe 'Echtzeit '92' in Sindelfingen.	<i>Klaus-Peter Schleisiek</i> .....	20
<b>Echtzeit '92</b> Der Autor hat beim Programmierwettbewerb den zweiten Platz mit Forth auf einem AT 386 erreicht. Er berichtet, wie er die Centronics-Schnittstelle des AT genutzt hat.	<i>Arndt Klingelberg</i> .....	22
<b>Neues aus der FG</b> Projektförderung durch die FG. In dieser Rubrik werden auch künftig die Direktoren oder das Büro der Forth-Gesellschaft wichtige Mitteilungen machen.	<i>Jörg Staben</i> .....	25
<b>Vorschau auf die VD 8/3</b>	.....	25
<b>Wanted</b> Wer Anregungen oder Wünsche bezüglich Forth hat, kommt hier zu Wort.	.....	26
<b>Briefe</b>		
Kein PRINZ FORTH!!	<i>Friederich Prinz</i> .....	27
Brief aus der Provinz	<i>Forth-Gruppe Moers</i> .....	29
Die Forth-Gruppe Stuttgart	<i>Wolf-Helge Neumann</i> .....	30
Prügel für das Schwein in der Kiste	<i>Jörg Plewe</i> .....	31
<b>Gruppen, Fachberatung, Ansprechpartner</b>	.....	32

# Direktorium

## Jörg Staben

Kurz zu meiner Person: Ich bin 36 Jahre alt und seit 6 Jahren mit Frau und Sohn verheiratet. Mein Brot verdiene ich bei einem großen Waschmittelkonzern in Düsseldorf, wo ich in Wechselschicht eine Produktionsanlage steuere - also ein rein handwerklicher Beruf. Zu FORTH kam ich über meinen ersten Rechner 1984, für den als erste zuladbare Sprache ein cassettengestütztes FORTH angeboten wurde. Leider konnte ich mangels Lehrbuch nie diese Sprache lernen. Dafür sorgte erst drei Jahre später Klaus Schleisiek-Kern mit seinem volksFORTH für den PC. Seitdem bin ich in der FORTH-Gesellschaft aktiv und wurde 1992 in Rostock ins Direktorium gewählt, obwohl ich lieber selbst Ansprüche an ein Direktorium gestellt hätte.



## Thomas Beierlein

Geboren: 1958 in Altenburg bei Leipzig; zehnjährige Schulausbildung, danach Maschinenbauer gelernt und ab 1979 in Mittweida an der Ingenieurhochschule Elektroniktechnologie studiert. 1986 auf mathematischen Simulationen zur Lasermaterialbearbeitung promoviert und nach einer dreijährigen Pause in der Industrie seit 1989 wieder in Mittweida tätig, jetzt auf der anderen Seite des Hörsaales - Fachrichtung "Automatisierungstechnik". Mein persönlicher Computervirus wurde so 1982 herum eingepflanzt und nach einer Anfangszeit in Z80-Assembler und BASIC bin ich dann recht schnell bei FORTH gestrandet und dort auch vor Anker gegangen. Inzwischen hat mich das "harte Leben" noch gezwungen C zu lernen, nach FORTH ein geteiltes Vergnügen. Ab Mitte der achtziger Jahre habe ich für die damaligen FORTH-Hobbyisten der DDR einige FORTH-Versionen für Z80-Homecomputer geschrieben, zum Schluß eine stark gestraffte Version des F83, die auf fast allen bei uns erhältlichen Homecomputern lief (außer ZX81 und Spectrum). Einige Projekte zur Datenauswertung und einige Controllerapplikationen waren weitere Anwendungen. Zur Zeit ist FORTH für mich hauptsächlich eine wunderschöne Umgebung zur hardwarenahen Fehlersuche. Leider bleibt wenig Zeit für größere Projekte.



## Ulrich Hoffmann

Ich wurde 1964 geboren. Meine ersten Computer-Kontakt hatte ich ca. 1977 an einer BS2000-Anlage in Pascal. Später kam der ZX-81 als erster eigener Rechner ins Haus. Neben den vielen anderen Programmen, die sich so ansammelten, gab's da auch eins mit Namen FORTH. Mit Hilfe von Ronald Zechs erstem Buch gelang es mir, das System zu bedienen. In Ermangelung eines Druckers wurde es handschriftliche decompiliert. Heute weiß ich, daß es sich im wesentlichen um ein FIG-Forth gehandelt hat. Ich nahm am 3. Bundeswettbewerb Informatik mit einer Compilerbau-Arbeit teil, in der ich Forth als Werkzeug einsetzte. Ich gewann den 1. Platz. Zu etwa gleicher Zeit formierte sich in Hamburg um Klaus Schleisiek die Forth Gesellschaft. Die lokale Gruppe Hamburg arbeitete an der Entwicklung des volksFORTHs, das auf dem 6502 und dann auch auf dem 68000 lief. Ich entschloß mich, volksFORTH auf dem Z80 unter CP/M zu implementieren. Wenig später konnte ich ein funktionsäquivalentes System demonstrieren. Ich begann 1985 mein Informatik-Studium in Kiel. 1987 gründete ich zusammen mit Marina Kern, Klaus Schleisiek und Jens Storzjohann die heutige Firma DELTA t, von der ich mich jedoch 1990 trennte, um mein Studium zu beenden. Zur Zeit arbeite an der Universität Kiel an meiner Promotion. Ich gehöre nicht zu denen, die mit Forth missionieren wollen, und jeden, den sie treffen, gleich in unendliche Gespräche verwickeln. Forth hat Stärken und Schwächen. Es kann einen hervorragenden Platz dort finden, wo es seine Stärken ausspielen kann, wie etwa im Bereich embedded Control oder der Echtzeitprogrammierung. Forth muß mit realen Projekten, mit fachlichen und sachlich richtigen Argumenten überzeugen. Auf diesem Weg wird die Kenntnis um Forth auch im akademischen Bereich zu verbessern sein, wo es bis dato noch weitgehend unbekannt ist.





# Redaktion

## Arndt Klingenberg

Jahrgang 1949, verheiratet, ein Kind. Diplom Ingenieur Maschinenbau (Werkzeugmaschinen, Meß- und Regeltechnik); bisher Fachjournalist und Leitung eines Testlabors im Bereich HiFi-Elektroakustik; Leitung einer MusiCassetten-Fertigung; nun selbstständig im Bereich Magnetbandtechnik / Qualitätskontrolle / HighSpeedDuplizierung, aber eben auch zunehmend 'embedded controller', Meßsysteme (nicht nur für HiFi-ELA), FUZZY-Auswertesysteme. Gründungsmitglied der FORTH e.V. . Mitglied fig, CCD, UIPRE, Fuzzy-Initiative NRW, VDI, AES, DIN-Normenausschuß. An Forth reizte mich die Interaktivität und mögliche 'Natürlich-Sprachlichkeit' für den Labortisch, sowie Modularität und Komfort. Stack/RPN war 'normal' seit 1972 durch HP35. Anfangs klemmte FORTH entsetzlichst, nirgendwo sinnvolle Hilfe zu public domain Forth Systemen. Die ersten Erfolgserlebnisse: LMI-Forth auf Z80/CPM, dann ASYST (bereits ISA-IBM-DOS), der wirkliche Durchbruch aber erst 1990 mit F-PC. Über das 'fette' F-PC lernte ich Mikroprozessortechnik und 'wirkliche' Programmierung (FORTH, Assembler, System) und eben auch Minimal-Systeme auf 'embedded controllern'.



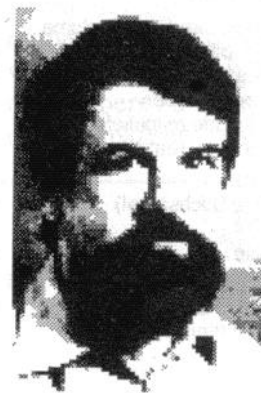
## Rolf Kretzschmar

Jahrgang 1947, Lehrer (OStR, Berufsschule), Bildhauer (Stein), Funkamateur (DF6KR). Verheiratet, ein Kind. Gründungsmitglied der Forth Gesellschaft e.V.. Seit mir 1980 jemand das Rockwell Forth für meinen AIM-65 unterschob, bin ich von Forth nicht mehr losgekommen. Die Möglichkeiten und Freiheiten, die diese Sprache bietet, kommt meiner Art mit Computern zu arbeiten sehr entgegen. Kein anderes Programmierkonzept scheint mir für den Einsatz in Berufsschulen geeigneter, Schülern das Know-How des Computers und dessen Arbeitsweise beizubringen. Forth ist in der Hand des Lehrers, der damit umzugehen weiß, wie Modellierton in der Hand des Bildhauers: Ideen können sehr schnell sichtbar gemacht und auf ihre Wirkung hin untersucht werden. Seit ich Tom Zimmers F-PC in die Hand bekam, konnte ich dann endlich auch meine Suche nach dem schultauglichen Forth-System aufgeben: Bis auf wenige Ausnahmen hatte ich all das, was ich anderen Systemen selber beibiegen mußte.



## Klaus-Peter Schleisiek

Jahrgang 1946, Diplom-Ingenieur der Elektrotechnik, tätig an der TH Aachen. Verheiratet, 2 Kinder. Mitglied der Forth Interest Group. Ich kam vor etwa 10 Jahren zu Forth, als ich es leid war, meinen Rockwell AIM-65 in Assembler zu programmieren; das BASIC war für die anstehenden Aufgaben entschieden zu langsam. Der erste Kontakt war vielversprechend, aber das Handbuch enthielt so viele Ungereimtheiten, daß ich erst mal entnervt aufgab. Schließlich hatte ich ja Zeit sparen wollen. Erst ein neuer Anlauf mit Leo Brodies STARTING FORTH führte dann zu Durchblick und dem damals wohl fast unvermeidlichen Gesellenstück: dem Decompiler. Seit ich Mitch Bradleys Forthmac kenne, bin ich von Forth überzeugt.





# Redaktorial

## Die Checkliste für Autoren

Redaktionen von Mitgliederzeit-schriften freuen sich über jeden ein-trudelnden Beitrag. Wir auch! Aber nicht jeder Artikel kann ohne Ände-rungen sofort in der nächsten Ausga-be erscheinen. Das kann mehrere Gründe haben. Am liebsten ist uns natürlich, wenn der Beitrags-Stack so voll ist, daß wir eine gute Mischung an Artikeln zusammenstellen können. Bei den dabei notwendigen Stack-Operationen kann dann auch schon mal ein Artikel mehrmals auf dem BOS (Bottom Of Stack) landen. Pro-blematisch wird es immer dann, wenn die Zahl der Artikel gerade für die nächste Ausgabe reicht und wir der

Meinung sind, daß ein Artikel noch überarbeitet werden muß. Wie sag' ich es in solchen Fällen meinem Kin-de...? Oft ist ja die Idee, die hinter ei-nem Beitrag zu erkennen ist, ganz pri-ma, doch fehlt es vielleicht an der Umsetzung. Aber wer gute Ideen hat, muß nicht auch ein guter Schriftsteller sein! Einige Umstellungen oder Er-gänzungen im Text können aber Wunder wirken. Die Redakteure wol-len gerne entsprechende Empfehlun-gen weitergeben. Es soll jedoch schon vorgekommen sein, daß der Ange-sprochene sauer reagierte und nie mehr 'was von sich hören ließ... . Das wollen wir gerade nicht!! Aus diesem

Grund veröffentlichen wir hier unse-ren internen Bewertungsbogen und diesen Vorschlag: Checkliste kopie-ren und schon während der Arbeit auf den Text anwenden (Je mehr Kreuze für "ja" bei "Einschätzungen", desto besser ist der Beitrag). Damit stellen Sie selber fest, ob Ihr Text schon rei-sefertig ist. Bitte schicken Sie die Texte als 8-Bit-ASCII Dateien (wenn möglich, ohne Zeilenumbruch, aber mit Absatzumbruch). Auf alle Fälle sollte der Beitrag auch auf Papier aus-gedruckt bei uns vorliegen. Diese Form stellt dann die Referenz für un-sere Formatierungen dar. Also: der ASCII-Text der Datei darf keine For-matanweisungen haben; der gedruck-te Text sollte formatiert sein! Bilder sollten als TIF- oder PCX- Dateien und als Hard-Copy eingeschickt wer-den. Kleinere Zeichnungen (10cm x 10cm) kann ich selber scannen.

rk

## Checkliste für Autoren

### Fakten

Verwendetes System: .....  
 ...ist für den Beitrag ohne Bedeutung  
 ...ist im Text angegeben

Spezielle Hardware: .....  
 ...ist notwendig  
 ...ist angegeben  
 ...ist beschrieben  
 ...Schaltbild(er) sind mitgeliefert

Anpassungshilfen für andere Forth-Systeme...  
 ...sind nicht sinnvoll  
 ...sind notwendig  
 ...sind im Text beschrieben  
 ...sind als Quell-Code mitgeliefert  
 ...sind leicht selber zu realisieren

Vorhandene Zusätze:  
 Stichworte (keine Stichsätze!)  
 Quellenangaben  
 Literaturhinweise

### Einschätzungen

Das Thema ist für die meisten VD-Leser  
 ...nützlich  
 ...interessant  
 ...aktuell

Der Text...  
 ...läßt sich gut lesen  
 ...hat einen angemessenen Umfang  
 ...ist inhaltlich strukturiert  
 ...ist optisch gegliedert (Überschriften!)  
 ...wird dem Thema gerecht  
 ...ist auch für Anfänger geeignet  
 ...muß nicht überarbeitet werden

Die Überschrift...  
 ...ist informativ  
 ...weckt Interesse

Die Kurzbeschreibung...  
 ...hat eine angemessene Länge  
 ...hebt wichtige Textteile hervor  
 ...motiviert, weiter zu lesen

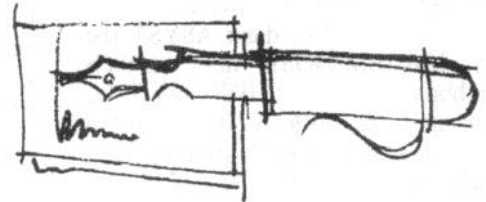
Die Illustration(en) ist (sind)...  
 ...aussagekräftig  
 ...textbezogen  
 ...wichtig für den Text  
 ...Schmuck für den Text  
 ...druckfertig

Der Quelltext...  
 ...scheint vollständig  
 ...scheint lauffähig  
 ...ist gut kommentiert  
 ...ist guter Forth-Stil  
 ...hat exemplarischen Charakter  
 .....ist für mehrere Forth-Systeme geeignet

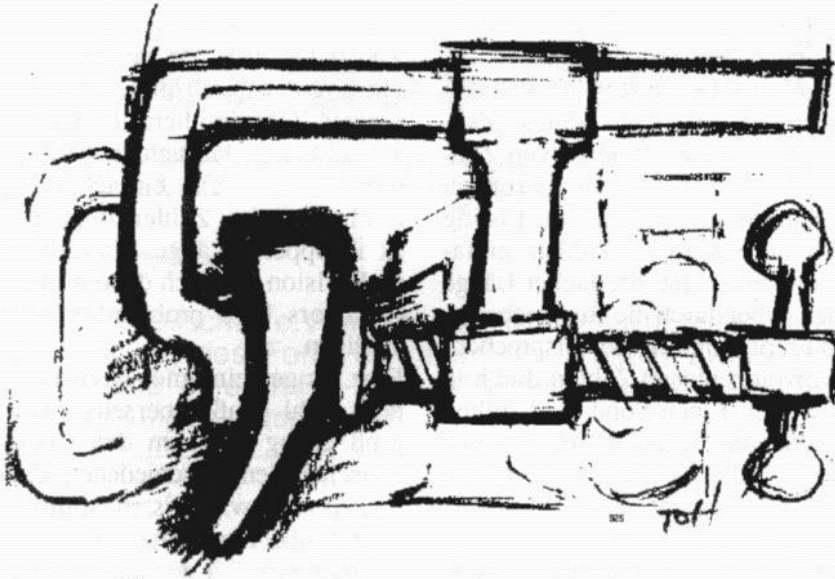
Der Artikel paßt in die Rubrik:

Forum	Briefe
	Info (Alles, was für Forther wichtig scheint)
	Schweinekiste (Code: nicht unbedingt musterhaft, aber nützlich)
	Splitter (kleine Besonderheiten)
	Bausteine (kleiner, musterhafter Forth-Code)
	Wanted (Code gesucht)
	Forth-Labor (Experimente am Forth)
	Forth-Werkstatt (Reparatur an Forth-Systemen)
	Forth in der Klemme (Probleme, Bugs)
	Report oder Interview
Schule (Diese Rubrik ist echten Lehrtexten und Kursen vorbehalten!)	
	Grundschule (für Anfänger)
	Mittelschule (für Fortgeschrittene)
	Hochschule (für weit Fortgeschrittene)

Werk	(Vollständige, lauffähige Applikation)
Produkt	(Marktreifes Produkt)
Projekt/Thema	(Vorschlag:.....)







# Festkommapaket

von Heinrich Hohl

Dieses kleine Festkommapaket erleichtert den Umgang mit 'Festkommazahlen', d.h. Zahlen doppelter Länge mit einem impliziten Dezimalpunkt. Die vorgestellten Routinen können in vielen Fällen helfen, den Einsatz eines Fließkommapaketes zu vermeiden.

FORTH-Compiler besitzen traditionsgemäß keine internen Routinen zum Umgang mit Fließkommazahlen. Stattdessen werden die entsprechenden Routinen durch ein separates Fließkommapaket zur Verfügung gestellt, um erst bei Bedarf ins Lexikon geladen zu werden. In FORTH lassen sich nämlich viele Problemstellungen, welche auf den ersten Blick den Einsatz von Fließkommaroutinen zu erfordern scheinen, bereits durch Zahlen im Festkommaformat lösen. Unter einer Festkommazahl versteht man dabei eine Integerzahl, welche an einer von vornherein festgelegten Stelle einen impliziten, d.h. nicht wirklich vorhandenen Dezimalpunkt aufweist. Um eine ausreichend hohe Rechengenauigkeit zu gewährleisten, werden solche Festkommazahlen üblicherweise durch Zahlen doppelter Länge

dargestellt.

Die Nachteile der Fließkommaarithmetik [1] sind hinlänglich bekannt: Fließkommaroutinen sind kompliziert und aufwendig, ein gutes Fließkommapaket belegt daher relativ viel Platz im Lexikon. Darüberhinaus werden Fließkommaroutinen sehr viel langsamer abgearbeitet als auf Integerarithmetik basierende Routinen. Eine vergleichbare Rechengeschwindigkeit läßt sich dabei allenfalls durch Zuhilfenahme eines numerischen Coprozessors erzielen.

Für FORTH-Programmierer lohnt es

sich somit auf jeden Fall, darüber nachzudenken, ob sich eine Anwendung nicht bereits durch den Einsatz von Integerarithmetik bewerkstelligen läßt. Die Festkommazahlen lassen sich dabei vorteilhaft in Programmen einsetzen, welche dazu gedacht sind, Daten lediglich zu erfassen und zu verwalten, aber nicht umfangreiche mathematische Berechnungen zur Analyse der Daten durchzuführen. Denn hierfür sollte nach wie vor ein Fließkommapaket als die angemessene Lösung betrachtet werden. Der Anwendungsbereich von Festkommaroutinen erstreckt sich somit von der Lagerverwaltung im Supermarkt mit manueller Dateneingabe bis hin zur vollautomatischen Meßdatenerfassung im Labor eines Physikers. Das hier vorgestellte Listing stellt die benötigten Routinen zur Verfügung, welche im folgenden näher besprochen werden sollen.

Die Routinen (FD.), FD. und FD.R dienen zur Ausgabe von Festkommazahlen, wobei an einer vorher festgelegten Stelle der Dezimalpunkt ausgegeben wird. Dessen Position läßt sich durch Aufruf von PLACES mit einem entsprechenden Parameter festsetzen. Die Routine FIXED bringt eine beliebige Zahl doppelter Länge, welche unter Umständen nicht mit der korrekten Anzahl von Nachkommastellen eingegeben wurde, in das mittels PLACES vereinbarte Festkommaformat. Vorausgesetzt wird hierbei, daß die auf dem Stack liegende Zahl unmittelbar zuvor durch NUMBER? konvertiert wurde, da auf den Inhalt der Variablen DPL zugegriffen wird.

Der Einsatz der bisher aufgeführten Worte läßt sich anschaulich anhand

## Stichworte

PC/FORTH  
Festkommazahlen  
Skalierungsoperatoren

## Zum Autor:

**Heinrich Hohl,**

Friedrichstr. 48, 7750 Konstanz Tel: 07531-64284

Geboren: 1962. 1983 Beginn des Physikstudiums an der Uni Konstanz. Seither Kontakt mit mehreren Programmiersprachen. Über Brodies 'Starting FORTH' Zugang zu Forth. Seit dieser Zeit ist Forth die Programmiersprache der Wahl. "Im Rahmen meiner Promotion setze ich Forth unter anderem zur Steuerung von Meßgeräten sowie zur Auswertung der dabei anfallenden Daten ein."

Hobby: Musik machen und hören (Rock'n Roll, HiFi, CD)



eines kleinen Beispiels erklären. Eine Lagerverwaltung z.B. soll von allen vorhandenen Artikeln den Preis in DM und die Masse in Kilogramm registrieren. Es bietet sich hier an, für den Preis zwei und für die Masse drei Nachkommastellen anzusetzen. Angaben wie 22.57 DM, 14 DM, 244.654 kg, 22.6 kg werden also durch die entsprechenden Integerzahlen 2257, 1400, 244654, und 22600 repräsentiert. Mit anderen Worten: der Preis wird in Pfennigen und die Masse in Gramm gespeichert. Zur Ausgabe der gespeicherten Daten könnten dann die Definitionen

```
: .DM ( fd -- ) 2 PLACES FD. ;  
: .kg ( fd -- ) 3 PLACES FD. ;
```

dienen. Eine Eingaberoutine für den Preis der Ware beginnt ebenfalls mit der Anweisung '2 PLACES', darauf folgt die eigentliche Eingaberoutine für eine Zahl doppelter Länge (diese Aufgabe läßt sich mit bereits hier vorgestellten Eingaberoutinen [2] erledigen). Zum Schluß sorgt der Aufruf von FIXED für die korrekte Anzahl von Nachkommastellen.

Die Wirkung der bisher vorgestellten Worte kann auch interaktiv sofort ausprobiert werden. Nach der Vorgabe von '2 PLACES' z.B. legt '43.6 FIXED' die Zahl 4360 auf den Stack, die von FD. als 43.60 ausgegeben wird. Entsprechend legt '12.345 FIXED' die Zahl 1234 auf den Stack, welche FD. als 12.34 ausgibt.

In einem Festkommapaket dürfen arithmetische Operatoren nicht fehlen. Die hier gebotenen Routinen decken den Bereich der vier Grundrechenarten ab, also Addition, Subtraktion, Multiplikation und Division. Zur Addition und Subtraktion von Festkommazahlen können trivialerweise die entsprechenden Operatoren D+ und D- für Zahlen doppelter Länge eingesetzt werden.

Der Operator D\* erlaubt es, eine Festkommazahl mit einer ganzen Zahl zu multiplizieren, D/ ermöglicht entsprechend die Division einer Festkommazahl durch eine ganze Zahl.

Wie kann nun aber eine Festkommazahl mit einer Zahl multipliziert werden, die sich ebenfalls nur als Festkommazahl darstellen läßt? Derartige

Aufgaben werden von den Skalierungsoperatoren D\*/ und DU\*/ durchgeführt. Der letztere deckt dabei als Operator für vorzeichenlose Zahlen einen größeren Bereich von Zahlen ab. Informationen hierzu können den Stackkommentaren im Listing entnommen werden. Zahlen einfacher, doppelter und dreifacher Länge werden dabei durch die Buchstaben n, d und t repräsentiert, die entsprechenden vorzeichenlosen Zahlen durch u, ud sowie ut. Einen Sonderfall stellen vorzeichenlose Zahlen im Bereich von 0 bis 32767 dar, welche durch den Buchstaben v symbolisiert werden.

Soll nun beispielsweise die Festkommazahl 14800.275 mit dem Faktor 2.235 multipliziert werden, so liefert die Eingabe von '14800.275 2235 1000 D\*/ FD.' mit der Zahl 33078.614 das korrekte Ergebnis. Ein Zwischenergebnis von dreifacher Länge verhindert dabei, daß infolge der ersten Multiplikation der zulässige Zahlenbereich überschritten wird.

Um einen speziellen Skalierungsoperator handelt es sich bei der als DU\*// bezeichneten Routine. Dieser Operator skaliert eine Zahl einfacher Länge entsprechend dem Verhältnis zweier doppelt langen Zahlen. Dadurch ermöglicht er unter anderem die so wichtige graphische Darstellung von Daten auf dem Bildschirm, wo Pixelwerte (einfache Länge) entsprechend den Verhältnissen von Festkommazahlen-Daten (doppelte Länge) skaliert werden müssen.

Besonders beim Einsatz des letztgenannten Skalierungsoperators kann es unter Umständen aufwendig sein, die auf dem Stack liegenden Argumentwerte in eine für den Operator geeignete Anordnung zu bringen. Hier kann es sich als günstiger erweisen, anstelle des Skalierungsoperators die entsprechenden Multiplikations- und Divisionsroutinen zu verwenden, mit deren Hilfe der Operator definiert wurde.

Die solchermaßen einsetzbaren Routinen T\*, T/ , TU\* und TU/ arbeiten analog zu den Verfahren, welche für die schriftliche Multiplikation und Division von Zahlen benützt werden. In TU// dagegen wird die Division einer Zahl dreifacher Länge durch eine Zahl doppelter Länge durch simples

Kürzen möglich gemacht: sowohl der Zähler als auch der Nenner des Bruches werden durch eine bestimmte geeignete Zahl dividiert, die so bemessen sein muß, daß sich im Nenner hinterher eine Zahl einfacher Länge ergibt und im Zähler folglich eine Zahl doppelter Länge. Die verbleibende Division läßt sich dann mittels des Operators UM/ problemlos bewerkstelligen.

Eine als gemeinsamer Divisor geeignete Zahl muß einerseits natürlich groß genug sein, um den Übergang zwischen den verschiedenen Zahlenlängen zu gewährleisten, sollte andererseits aber trotzdem so klein wie nur möglich gewählt werden, um den bei der Umwandlung auftretenden Rundungsfehler gering zu halten. Als ideale Wahl erweist sich damit die um eins vergrößerte, oben auf dem Stack liegende Hälfte der im Nenner stehenden Zahl doppelter Länge. Ein hexadezimal als (178 1E8A 9590)/(335 F250) dargestellter Bruch muß also beispielsweise mit der Zahl 336 gekürzt werden, wodurch er in den äquivalenten Ausdruck (7523 0CA3)/(FFFB) übergeht, und sich damit zu 7525 berechnen läßt.

Selbstverständlich sind die hier vorgestellten Skalierungsoperatoren nicht nur für die Arbeit mit Festkommazahlen von Nutzen, sondern können auch im Zusammenhang mit den üblichen Zahlen doppelter Länge vorteilhaft eingesetzt werden.

Das Festkommapaket wurde mit PC/FORTH 3.2 von der Firma LMI entwickelt. Eine Anpassung des Listings an Tom Zimmers F-PC ist sehr einfach. Hierzu genügt es zu wissen, daß der im Listing auftretende Operator M\* in F-PC unter dem Namen \*D enthalten ist.

### Quellen

[1] Leo Brodie:  
Starting FORTH  
Second Edition 1987  
Prentice Hall, New Jersey

[2] Heinrich Hohl:  
'Ein intelligenter Editor für Eingabefelder' VD 7/4 DEZ. 1991



## Listing zu: Festkommapaket (H. Hohl)

### Screen #00

```
\ FIXED.SCR HH 18:45 08.04.92
written using LMI PC/FORTH 3.2
Heinrich Hohl, W-7750 Konstanz
```

This small package was written to facilitate usage of 'fixed point double length (fd) numbers', i.e. double length numbers containing an implicit decimal point at a known, fixed position. The following words are offered to handle fixed point numbers:

```
PLACES ..... determine position of decimal point
(FD.) FD. FD.R ..... used to display fd numbers
FIXED ..... convert any input d number to fd number
D+ D- D* D/ ..... calculate with d or fd numbers
T* T/ TU* TU/ TU// ..... basic triple length number operators
D*/ DU*/ DU*// ..... scale d or fd numbers; triple length
intermediate results are used
```

### Screen #01

```
\ Load screen HH 18:45 08.04.92
FORTH DEFINITIONS HEX
2 ?SCREENS 1- THRU DECIMAL
```

```
EXCISE Places Places
EXCISE MULTIPLY TRUNCATE
EXCISE UM/ UM/
```

### Screen #02

```
\ Fixed point number output words HH 18:45 08.04.92
VARIABLE Places
```

```
: PLACES ( u -- ) Places ! ;
\ determine number of places to the right of the decimal point
: (FD.) ( fd -- addr len)
TUCK DABS
# Places @ 0 ?DO # LOOP ASCII . HOLD #S ROT SIGN #> ;
\ convert fixed point double length number to formatted string
: FD. ( fd -- ) (FD.) TYPE SPACE ;
\ display fixed point double length number
: FD.R ( fd width -- ) >R (FD.) >R OVER - SPACES TYPE ;
\ display number right justified in a field of specified width
```

### Screen #03

```
\ Fixed point number input converter HH 18:45 08.04.92
: D* ( d v -- d') TUCK * >R UM* R> + ;
: D/ ( d v -- d') TUCK /MOD >R SWAP UM/MOD NIP R> ;
```

```
\ multiply or divide double length number by 0 <= v <= 7FFF
: MULTIPLY ( d u -- d') 0 ?DO 0A D* LOOP ;
: TRUNCATE ( d u -- d') 0 ?DO 0A D/ LOOP ;
\ multiply or divide double length number several times by ten
: FIXED ( d -- fd)
Places @ DPL @ 0 MAX -
DUP 0< IF NEGATE TRUNCATE ELSE MULTIPLY THEN ;
\ convert any input double length number to a fixed point
\ double length number with the specified number of places
```

### Screen #04

```
\ Triple length number arithmetic operators HH 18:45 08.04.92
: UM/ ( ud u -- u') UM/MOD NIP ;
```

```
: T* ( d v -- t) TUCK M* >R >R UM* 0 R> R> D+ ;
: T/ ( t v -- d) DUP >R M/MOD -ROT R> UM/ SWAP ;
\ multiply or divide signed long number by 0 <= v <= 7FFF
: TU* ( ud u -- ut) TUCK UM* >R >R UM* 0 R> R> D+ ;
: TU/ ( ut u -- ud) DUP >R UM/MOD -ROT R> UM/ SWAP ;
\ multiply or divide unsigned long number by unsigned number
: TU// ( ut ud -- u)
DUP 1+
DUP >R UM/ R> SWAP >R TU/ R> UM/ ;
\ divide unsigned triple length number by unsigned double
\ length number
```

### Screen #05

```
\ Scaling operators HH 18:45 08.04.92
: D*/ ( d v1 v2 -- d') >R T* R> T/ ;
```

```
\ scale double length number d according to the unsigned
\ numbers v1 and v2 of range 0 <= v <= 7FFF: (d*v1)/v2 = d'
: DU*/ ( ud u1 u2 -- ud') >R TU* R> TU/ ;
\ scale unsigned double length number ud according to the
\ unsigned numbers u1 and u2: (ud*u1)/u2 = ud'
: DU*// ( u ud1 ud2 -- u') >R >R ROT TU* R> R> TU// ;
\ scale unsigned number u according to the unsigned double
\ length numbers ud1 and ud2: (u*ud1)/ud2 = u'
\ in order to avoid fussy stack operations, it may often be
\ favourable to use basic triple length arithmetic operators
\ for scaling instead of the scaling operators shown here
```

Achtung! Schon jetzt vormerken!  
Die

# EuroForth '92

findet vom

**23. bis 25. Oktober**

in Southampton (GB) statt.

Motto: "Unity, Communication and Exploration"

Kontakt: EuroForth Office: MPE

133 Hill Lane

UK Southampton

SO1 5AF

Tel.: 004 / 703 / 63 14 41

Fax: 004 / 703 / 33 96 91

# Light-Show

Die einzig sinnvolle Anwendung für Ihren 486-50!

von Jörg Staben

Sie wissen nicht, wie Sie Ihren 486-50 sinnvoll beschäftigen können? Nun, dem Grübeln darüber wird mit diesem Artikel endlich ein Ende gesetzt! Zudem lernen Sie Ihr F-PC etwas näher kennen.



Ich weiß nicht, wie oft das Thema "Die AT-Tastatur und ihre drei LOCK-Tasten" die bunten Seiten der Computermagazine gefüllt hat.

Die DOS International hat 1988 den Vorreiter gemacht und ist alleine 1991 fünf oder sechs mal von anderen Magazinen "zitiert" worden oder hat sich selbst wiederholt. Das Thema ist solange immer wieder aufgewärmt worden, bis die ersten BIOSse auftauchen, in deren SETUP man einstellen kann, wie die NUMLOCK-Taste beim Booten geschaltet werden soll (z.B.: AMI486). Das Ganze zeigt, daß man ein Computermagazin sehr schön mit immer denselben Themen füllen kann - es muß nur etwas Zeit dazwischen liegen.

Dennoch ist das eine gute Gelegenheit, mal die Tastatur und deren Leuchtdioden für die Grundlagen der Logik zu mißbrauchen und eine LightShow zu veranstalten.

Die Leuchtdioden der Tasten CAPS-LOCK, NUMLOCK und SCROLL-LOCK werden zufällig an- und wieder ausgeschaltet; das geschieht mit etwas Verzögerung, damit's besser aussieht. Benutzen kann man den Rechner nicht mehr, während das Programm läuft, dazu müßte der Multi-Tasker herhalten.

LIGHTSHOW läuft gut, aber so richtig schön wäre es gewesen, wenn das

Programm über den Unterhaltungswert hinaus noch eine richtige Funktion hätte: Es könnte beliebige LOCK-Tasten ein- oder ausschalten;

dafür muß natürlich eine TCOM-Variante her. Denn wer will schon eine 167K-Datei für solch eine Funktion haben?

## Quelltext zu: LIGHT-SHOW... (Jörg Staben)

```

\\ Lichtorgel mit den LOCK-Tasten                                     J.Staben Mail1992

    Das Programm läuft in dieser Form
    in allen DREI (!) Entwicklungsumgebungen:
    F  LITESHOW.SEQ      OK
    ZZ LITESHOW.SEQ      F5
    NEWZ LITESHOW.SEQ    ALT-F2
}
***** jrg 04/26/92 13:56 *****
TCOM und F-PC müssen unterschiedlich behandelt werden.

Für die ALF-F2 Option von NEWZ:
<tcom liteshowl.seq> \ Bei Problemen evtl. TCOM-Pfad prüfen!
*****
{
\ipc true  #IF
\tcom false #IF
                cr .( F-PC aktiv! )
                anew keyb_lock
                warning on
                #THEN
}
***** jrg 04/26/92 13:51*****
LOCK_STATUS liefert die BIOS-Adresse,
an der der Status der LOCK-Tasten gespeichert ist.

Als 32Bit-Variable:
2Variable lock_status 0000 0417 lock_status 2!

Um das notwendige 2@ zu vermeiden, wird hier 2VALUE
eingeführt
*****
: 2Value:      ( n1 n2 "name" -- )
               Create
               / /
               Does> ( -- n1 n2 )
               2@
               ;

hex
0000 0417 2Value: lock_status
}
***** jrg 04/26/92 14:06 *****
SET und
CLEAR setzen oder löschen des Bit
an der jeweiligen Bitposition
*****
{
decimal
16 Constant scrolllock
32 Constant numlock
64 Constant capslock

: set
  ( d bitpos -- )
  >r
  2dup c@l \ Byte aus SEG:OFF $417 holen
  r> or \ Bitposition OR-verknüpfen
  -rot c!l
  ;

: clear
  ( d bitpos -- )
  >r
  2dup c@l \ Byte aus SEG:OFF $417 holen
  r> xor {siehe Anmerkung von akg}
  -rot c!l
  ;

}
***** jrg 04/26/92 14:09 *****
CLEARED: und
SET: generieren die Setz- und Löschwörter
*****

```

### Stichworte

IBM-AT  
F-PC  
TCOM  
Numlock  
Keyboard-Status  
Command-Line  
TIB



Da man in FORTH säuberlich zwischen dem interaktiven Erweitern und Aufrufen von FORTH-Wörtern und einer eigenständigen Anwendung unterscheiden muß, hat die Applikation für den TCOM einen etwas anderen Algorithmus: Werden dem Programm Parameter -S -N -C übergeben, so führe die entsprechende Ausschaltfunktion durch. Werden keine Parameter übergeben, so zeige LIGHTSHOW, bis eine Taste gedrückt wird. Nur - es funktioniert leider nicht! Kann man im TCOM nicht den Eingabespeicher TIB interpretieren lassen? Warum nicht oder kann man das ändern? Oder wie bringt man die Parameter dazu, auch was zu machen? Während die Antwort noch aussteht - viel Spaß mit LIGHTSHOW!  
 Bis dahin kann man allerdings das F-PC als FORTH-Interpretersystem als Anwendung mißbrauchen und die Syntax von FORTH loben und preisen: Während unter TurboPASCAL die abstruseste Syntax wie

```
IF ParamCount = 1 THEN
BEGIN
Coms := (Hardware2 AND
$06) shr 1;
BEGIN
Parameter := ParamStr(I);
...
```

gepflegt wird, testen wir unter F-PC die neuen Wörter erst mal ganz locker mit `lock_status numlock set` oder mit `caps_on`. Wenn die Lämpchen dann sofort reagieren, wissen wir, daß es geht und ein Turbo-Compiler nicht halb so schnell ist, wie er tut.

Aber zurück zur Syntax: Während die Compiler-Besitzer (hat eigentlich mit PASCAL gar nichts zu tun) mühsam die DOS-Kommandozeile übernehmen und in Einzelteile zerlegen müssen, kann FORTH die Kommandozeile sofort interpretieren: `dos>tib interpret` - das war's.

Gegenüber einem anderen FORTH kehrt sich hier der Komfort des F-PC in's Gegenteil um; man muß ihm erst seine "normale" Interpretation der Kommandozeile abgewöhnen, die da ist:

`f-pc <filename> <forth words>`  
 Ein fehlender Dateiname wird nach

## Fortsetzung des Listings zu: LIGHT-SHOW...

```
{
: cleared:      ( n "name" -- )
                Create
                /
                Does @ ( -- n )
                lock_status rot
                clear
                ;

: set:          ( n "name" -- )
                Create
                /
                Does @ ( -- n )
                lock_status rot
                set
                ;

}
***** jrg 04/26/92 15:42 *****
                LOCK-Taste einzeln ausschalten
*****
{
scrolllock cleared: scr_off
numlock   cleared: num_off
capslock  cleared: caps_off

\fpc true #IF
\tcom true #IF      \ Kommandozeilen für TCOM
: -s   scr_off ;
: -n   num_off ;
: -c   caps_off ;
#THEN

}
***** jrg 04/26/92 15:43 *****
                LOCK-Tasten einzeln einschalten
*****
{
scrolllock set: scr_on
numlock   set: num_on
capslock  set: caps_on

\fpc false #IF
\tcom true #IF      \ Kommandozeilen für TCOM
: +s   scr_on ;
: +n   num_on ;
: +c   caps_on ;
#THEN

}
***** jrg 04/26/92 14:56 *****
                ALL OFF
                schaltet alle LOCK-Tasten aus
*****
{
: all_off      ( -- )
                0 lock_status c!l
                ;

}
***** jrg 04/26/92 14:13 *****
                Die Lichtorgel
*****
{
needs rand3.seq
3 Constant #lock_keys

Variable scr_state

: scr_toggle   ( -- )
                scr_state @
                IF     scr_off
                ELSE   scr_on
                THEN
                scr_state @
                true xor
                scr_state !
                ;

Variable num_state

: num_toggle   ( -- )
                num_state @
                IF     num_off
                ELSE   num_on
                THEN
                num_state @
                true xor
                num_state !
                ;

Variable caps_state

: caps_toggle  ( -- )
                caps_state @
                IF     caps_off
                ELSE   caps_on
                THEN
                caps_state @
                true xor
                caps_state !
                ;

}
***** jrg 04/26/92 14:21 *****
                Los geht's
*****
```

dieser Konvention mit einem Minus "-" angedeutet, so daß lockoff --n-c die CAPSLOCK- und die NUMLOCK-Taste ausschalten würde. Aber weil das nicht schön ist, enthüllen wir mit B HELLO und B HDEFAULT, wie man's macht. Darauf wird das Programm sofort um eine Abfrage "Anwendung NEIN oder DOCH nicht?" zusammen mit einem ' lock\_off is boot fsave lockoff ergänzt und schon schaltet lockoff -n-s-c von DOS-Ebene aus alle Lämpchen! Das testen Sie natürlich aus dem F-PC heraus:  
sys lockoff -n-s-c  
Nichtsdestotrotz - die TCOM-Lösung steht noch aus! executes> tib wäre mir als Schreibweise am liebsten.

P.S.1:  
Ich habe wirklich nichts dagegen, wenn jemand Stil und Inhalt meiner Programme kommentiert oder korrigiert; ich habe aus solchen Leserzuschriften schon viel Elegantes gelernt.

P.S.2:  
Sollte es jemand interessieren: Laut TIMEIT, einem Tool aus dem Lieferumfang des F-PC/TCOM, dauert das ausschalten mit LOCKOFF.EXE (167000 Byte) unendliche 0.8 Sek., mit NUMOFF.COM (36 Byte) winzige 0.2 Sek.!

P.S.3:  
Sollte jemand gerade seinen Beilstein wg. der Logiktafel von AND, OR, XOR verlegt haben, einfach die Datei LOGICS.SEQ compilieren.

### Fortsetzung des Listings zu: LIGHT-SHOW...

```
{
: lightshow ( -- )
BEGIN
#lock_keys rnd
CASE
0 OF scr_toggle 1 tenths ENDOF
1 OF num_toggle 1 tenths ENDOF
2 OF caps_toggle 1 tenths ENDOF
drop
ENDCASE
key? IF key drop true
ELSE false THEN
UNTIL all_off
;
\ fpc cr lightshow

\ fpc true #IF
\ tcom false #IF
cr .( Soll eine Anwendung generiert werden? J = Ja )
key upc
ascii J = not
#IF \s #THEN
#THEN
}
***** jrg 04/26/92 14:21 *****
ANWENDUNG ERZEUGEN mit F-PC/TCOM
Ab hier geht's im TCOM schief,
im F-PC um so eleganter
*****
\ fpc false #IF
\ tcom true #IF
: do_options ( <options> )
bl word dup c@ '- ' =
IF 1+ c@ upc
CASE 'S' OF scr_off ENDOF
'N' OF num_off ENDOF
'C' OF caps_off ENDOF
drop
ENDCASE
ELSE drop
THEN
;

: main ( <options> -- )
\ dos_to_tib \ default?
#tib @
IF do_options \ Das funktioniert
\ executes> tib \ nicht?!
cr ." LockKey OFF " \ Warum geht das nicht?
cr
ELSE cr ." --- LIGHTSHOW --- "
cr ." Jede Taste beendet!"
cr lightshow
THEN
;

#ELSE
\ Direkt aus HELLO und HDEFAULT kopiert!
: lock_off ( --- )
SP0 @ 'TIB !
IN OFF
SPAN OFF
#TIB OFF
LOADING OFF
\ u NOSETCUR NOSETCUR OFF
ONLY FORTH ALSO DEFINITIONS
DEFAULTSTATE
dos-line c@ 0=
IF cr
cr ." --- LIGHTSHOW --- "
cr ." Jede Taste beendet!"
cr lightshow
ELSE
THEN dostib interpret
bye
;

' lock_off is boot fsave lockoff

#THEN
\ fpc cr .( Anwendung LOCKOFF generiert! )
\s
```

#### Zur Kompatibilität mit anderen Versionen:

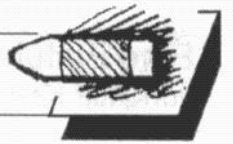
\\ : ak-version: vor der ersten geschweiften Klammer-zu '}' muß der durch '\\ ausgelöste Kommentar durch '-\\' beendet werden.

\FPC \TCOM: v. < 3.54 oder ak-version: in F-PC sind alle \FPC zu löschen und \TCOM durch \ zu ersetzen, in TCOM umgekehrt.

Ein kleiner Bug: CLEAR ist überall in TOGGLE umzubenennen oder das XOR ist durch NOT AND zu ersetzen. Das NOT hat die Funktion des ANS INVERT, dreht also im Gegensatz zu 0= alle Bits einzeln um. Wie es so geht, das Programm funktioniert auch so, aber NUM\_OFF kann auch einschalten.

Siehe auch Anmerkung auf Seite 19

akg



# Eine Einführung in die Arbeit mit Forth

von Rolf Kretzschmar

F-PC ist inzwischen weltweit wohl das am stärksten verbreitete Public-Domain-Forth-System. Zudem zählt es zu den größten Forth-Systemen überhaupt. Der vorliegende Artikel wendet sich an Computerlaien, die gerne programmieren möchten und dies mit der faszinierenden Sprache Forth versuchen wollen.

Programmieren heißt, dem Computer etwas beizubringen, das er vorher nicht konnte. Da Sie von einem Computer aber keine Fantasie erwarten können, müssen Sie bei Ihren Lehrversuchen bis aufs letzte Leerzeichen korrekt die Anweisungen eingeben.

Wie man dies in Forth machen kann, soll nun Schritt für Schritt an kleinen Beispielen kurz gezeigt werden. Dazu sollten Sie unbedingt Ihren Computer eingeschaltet und Forth gestartet haben. "Learning by doing" ist auch hier die beste aller denkbaren Lernmethoden.

## Der Dialog mit Forth

Anders als in Pascal - aber ähnlich wie in Basic - können Sie Forth benutzen, ohne vorher ein Programm geschrieben zu haben. Vergleichbar ist dies mit der Benutzung eines Taschenrechners. Sie können mit dem Forth-System einen *Dialog* führen: Sie geben etwas über die Tastatur ein und bekommen (oft) eine Antwort.

Viel interessanter ist allerdings die Fähigkeit des Computers, etwas Neues zu lernen. Dies geschieht üblicher Weise mit Hilfe einer Programmiersprache. Denken Sie sich die Programmiersprache Forth als ein recht intelligentes Kind, das zwar von Ihren Problemen nicht die geringste Vorstellung hat, das aber wegen seines umfangreichen Allgemeinwissens, seiner Vorkenntnisse und seines erstaunlichen Wortschatzes sehr lernfähig ist.

hig ist.

Spätestens an dieser Stelle sollten Sie Forth installiert und gestartet haben, damit Sie die folgenden kleinen Beispiele nachvollziehen können. Unmittelbar nach dem Starten des Forth können Sie in einen Dialog eintreten: Sie befinden sich nämlich auf der *Dialogebene* des Forth. Überzeugen Sie sich zunächst von dem enormen Wortschatz des "Forth-Kindes", indem Sie einen Blick in sein Wörterbuch werfen.

Um dies zu erreichen, "sagen" Sie einfach `words` `□ ↵`.

(Mit `□ ↵` ist die Return-Taste - das ist die Taste mit dem geknickten Linkspfeil `□ ↵` oder *Enter* gemeint. Durch Drücken dieser Taste übergeben Sie das Wort auf dem Bildschirm dem *Forth-Interpreter* zur Auswertung. Bevor Sie `□ ↵` betätigen, können Sie noch Schreibfehler korrigieren.)

Erschrecken Sie jetzt nicht! Sie müssen keineswegs die meisten dieser Worte auswendig lernen, um in Forth programmieren zu können. Wie in natürlichen Sprachen sind Sie bereits mit einem kleinen Grundwortschatz in der Lage, einen brauchbaren Dialog zu führen.

Vermutlich haben Sie sowieso die wenigsten Worte aufnehmen können, weil sie viel zu schnell über den Bildschirm rasten. Geben Sie einfach erneut `words` ein und unterbrechen Sie die Ausgabe mit der Leertaste (Abk.: `□ space`); weiter geht es wieder mit `□ space`. Mit `□ Esc` können Sie vorzeitig die Ausgabe beenden.

Irgendwo im Wörterbuch des Forth steht das Wort `words`, das Sie gerade benutzt haben. Als Antwort auf die Eingabe dieses Wortes bekamen Sie alle vorhandenen Worte gezeigt.

Warnung: Erliegen Sie nicht der verständlichen Versuchung, interessant klingende Worte zu benutzen! Wenn man sich ins Ausland begibt, kann der falsche Gebrauch der Sprache gefährliche Reaktionen hervorrufen. So kann es Ihnen auch in Forth ergehen.

Testen Sie nun die Reaktion von Forth auf die folgenden Worte, indem Sie diese einzeln eingeben und mit `□ ↵` die Eingabezeile abschließen:

```
cls cr decimal test 11 22
```

Wie Sie sehen, gibt es Worte, die Forth veranlassen, ohne Rückmeldung etwas zu tun oder aber Ihnen eine Mitteilung zu machen (z.B. sagt das `ok` nach einer Eingabe, daß Forth alles verstanden hat). Nach `cls` wurde der Bildschirm gelöscht und mit `cr` sprang der *Cursor* an den Anfang einer neuen Zeile.

Auf `test` antwortete F-PC mit `<- what?`. Das heißt soviel wie: "Was soll das denn? Das kenn' ich nicht!" Dieses Wort steht somit nirgends in seinem Wörterbuch.

Nach `decimal`, `11` und `22` tut sich anscheinend (abgesehen von der Mel-

---

Ein Computer tut immer das, was Sie ihm sagen, aber häufig nicht das, was Sie von ihm wollen!

---

dung `ok`) nichts. Geben Sie jetzt noch folgendes ein:

```
+ . □ ↵
```

**Achten Sie bei allen Eingaben unbedingt auf die Leerzeichen! Leerzeichen sind in Forth als Trennzeichen zwischen den Operatoren und Parametern zwingend notwendig!!**

Forth antwortet sofort mit: `33 ok`. Vielleicht haben Sie schon richtig vermutet: Forth merkt sich eingegebene Zahlen, bis sie verarbeitet werden (hier mit dem Operator `+`) oder ausgegeben werden (hier als Ergebnis der Rechnung `11 + 22`). Der Punkt ist ein

### Stichworte:

F-PC  
Schule  
Anfänger  
Einführung

**Forth-Wort.** Er veranlaßt die Ausgabe des Ergebnisses auf dem Bildschirm. (In Basic wird dies z.B. durch den Befehl PRINT erreicht.)

Wenn Sie jetzt neugierig geworden sind, wie man im Dialog mit Forth rechnet, so möchten wir Sie an dieser Stelle ermuntern, das Buch "Programmieren in Forth" von Leo Brodie (Hanser Verlag, München) zur Hand zu nehmen und dort die entsprechenden Beispiele auszuprobieren.

Wenn Sie wollen, können Sie aber auch unserer kleinen Einweisung folgen, um zu lernen, wie man dem Forth-Kind etwas Neues beibringt.

### Das erste neue Wort

Bis hierher haben Sie nicht in Forth programmiert, sondern sich mit Forth unterhalten: Sie haben es veranlaßt Dinge zu tun, die seine Schöpfer ihm beigebracht haben. Doch ist es nicht ärgerlich, daß Forth nicht einmal Ihren Namen kennt?! Wenn Sie ihn probeweise eingeben, antwortet er mit <- what? als Ausdruck seines Unverständnisses. Wenn Sie zufällig *Mayer* heißen, geben Sie nun den Doppelpunkt (:), ein Leerzeichen, den Namen Mayer, wieder ein Leerzeichen, ein Semikolon (;) und zuletzt   ein! Auf Ihrem Bildschirm sollte nun folgendes zu sehen sein:

```
: Mayer ; ok
```

Wenn Sie nun *Mayer* eingeben, kommt keine Fehlermeldung mehr, sondern ein ok. (Die Unterstreichung bedeutet, daß dies eine Ausgabe von Forth ist.)

```
Mayer ok
```

Ihr Computer kennt Sie jetzt. (Erwarten Sie bitte nicht, daß Forth Sie nun auch immer versteht...) Forth hat Ihren Namen an oberste Stelle in sein Wörterbuch eingetragen (Schauen Sie ruhig nach...!). Bevor Sie nun die Namen aller Freunde/innen eintragen, müssen wir Sie warnen: wenn Sie morgen Ihren Computer erneut einschalten, dürfen Sie nicht enttäuscht sein, wenn Forth Sie wieder vergessen hat....

Was haben die drei Worte (:, Mayer und ;) nun bewirkt?

Das Wort ':' (Bitte gewöhnen Sie sich schon jetzt daran, daß auch Satzzeichen - wie in diesem Fall - für Forth *Worte* sein können!) heißt so-

viel wie: "Merke Dir bitte das folgende Wort, indem Du es in Dein Wörterbuch einträgst!" Das abschließende ';' heißt soviel wie: "Das war's, was Du Dir merken solltest!"

Achtung: Nachdem Sie den Doppelpunkt eingegeben haben, ist Forth derart mit dem Lernen beschäftigt, daß es für einen Dialog keine Zeit mehr hat. Erst wenn der Lernvorgang mit dem Semikolon abgeschlossen wurde, ist ein Dialog wieder möglich, und Sie können das eben Gelernte sofort als bekannt voraussetzen.

Sie haben natürlich recht: es macht wenig Sinn, dem Computer Worte beizubringen, wenn diese nichts bewirken. Was halten Sie davon, wenn nach Eingabe des Namens *Mayer* Ihre Adresse samt Telefonnummer und - wenn Sie wollen- zusätzlich Kontonummer, Sternzeichen usw. auf dem Bildschirm ausgegeben werden?

Nichts ist einfacher, als dies Ihrem Forth beizubringen. Geben Sie nun die folgenden Zeilen ein. Aber vorsicht! Wenn Sie sich verschrieben haben, können Sie nur innerhalb einer Zeile noch korrigieren. Sobald Sie eine Zeile mit   abgeschlossen haben, können Sie (vorläufig) keine Korrekturen mehr darin vornehmen. Sollte nach   die Fehlermeldung <- what? erscheinen, müssen Sie sogar wieder mit dem Doppelpunkt beginnen! (Den unterstrichenen Text *nicht* eingeben!)

```
: Mayer  
```

```
MAYER isn't unique
```

```
cr  
```

```
." Peter Mayer" cr  
```

```
." Kölnstr. 22" cr  
```

```
." 4711 Wasserstadt" cr cr  
```

```
;   ok
```

An der Tatsache, daß nach Betätigung von   kein ok erscheint, erkennen

Sie, daß Forth "voll beim Lernen" ist, also für einen Dialog keine Zeit hat. Hoffentlich haben Sie sich nicht dadurch irritieren lassen, daß Forth Ihnen nach Eingabe von Mayer den freundlich gemeinten Hinweis gibt, daß es das Wort Mayer schon kennt (MAYER isn't unique); Grundsätzlich haben Sie nämlich die Freiheit, dem Forth die gleichen Worte beliebig oft erneut beizubringen. Forth "benutzt" dann allerdings immer das zuletzt gelernte. Und da Sie (siehe oben) bereits Mayer definiert hatten, kommt die genannte Meldung. Sobald nach Eingabe des Semikolons Forth mit ok antwortet, können Sie das neue Forth-Wort sofort anwenden. Geben Sie also jetzt (...Sie sind wieder im Dialogmodus!) Mayer   ein! Versuchen Sie es auch mit *mayer* und *MAYER*!

Als neues Wort haben Sie jetzt das ." kennengelernt, das als untrennbaren Teil immer das später folgende " benötigt. Mit ." wird während des Lernvorganges der Text einleitend markiert, der zum Ausgabegerät (hier: zum Bildschirm) geschickt werden soll. Mit dem abschließenden " wird lediglich das Ende des betreffenden Textes markiert.

Dringende Warnung: Sie sollten an dieser Stelle mit der Eingabe eines Telefonverzeichnis all Ihrer Bekannten warten, bis wir Ihnen verraten haben, wie man diese Daten dauerhaft abspeichern kann!

In Abb. 1 ist noch einmal schematisch der Aufbau einer Wortdefinition dargestellt.

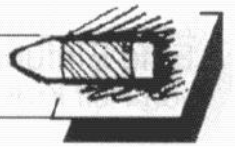
☞ **Da Forth das Ende eines Wortes am folgenden Leerzeichen erkennt, darf innerhalb des Wortnamens kein Leerzeichen stehen.** Dies ist - abgesehen von einer maximalen Län-

### Aufbau eines Forth-Wortes

: Neu-Wort	Der Doppelpunkt schaltet den Lernmodus ein. Das erste Wort nach dem Doppelpunkt ist das Wort, das neu gelernt und in das Wörterbuch eingetragen wird.
Anweisung1	Es folgen beliebig viele Anweisungen, die dann ausgeführt werden, wenn Neu-Wort später aufgerufen wird.
Anweisung2	
Anweisung3	
;	Das Semikolon beendet den Lernmodus und schaltet zurück auf die Dialogebene.

Abb. 1





ge- aber auch die einzige Einschränkung, die bezüglich der Wahl des Namens zu beachten ist.

An dieser Stelle muß noch einmal auf Eigenschaften der Fehlermeldung <- what? hingewiesen werden. Diese Meldung kommt dann, wenn Sie ein Wort eingeben, das Forth nicht kennt. Wenn Sie sich nun im Programmiermodus befinden (also nach Eingabe des Doppelpunktes), dann wird Forth die Programmierung von sich aus abbrechen!

Das ist auch gut so, denn das Wort, das Sie gerade programmieren, würde ja sowieso nicht richtig arbeiten können, da es ein nicht definiertes Wort enthielte. Leider müssen Sie in diesem Fall alle Eingaben seit dem letzten Doppelpunkt wiederholen. Ein kleiner Tip: Betätigen Sie mal die beiden Pfeiltasten für rauf und runter! Sie bekommen in F-PC damit die letzten Eingaben zurück und können auf diese Weise sehr einfach Eingaben wiederholen!

## Die erste Applikation

Sicher ist Ihnen aufgefallen, wie sparsam wir bisher mit dem Begriff Programmieren umgegangen sind, obwohl wir -verglichen mit anderen Programmiersprachen- genau dies gemacht haben: wir haben erste kleine Programme geschrieben. Nur nennen wir diese nicht Programme, sondern Worte.

Ziel Ihrer Arbeit mit Forth wird es sein, die richtigen Worte zu finden, mit denen Sie Ihr Problem beschreiben können. Wenn Sie gelernt haben werden, gute Worte zu finden, versteht Ihr Computer (wenn er Forth

Unter einer Forth-Applikation versteht man die Gesamtheit der Worte, die zur Lösung spezieller Probleme erzeugt und sinnvoll zusammengefügt wurden.

spricht) diese so gut wie der Mensch. Der gesamte Wortschatz, den Sie dem Forth-System programmierend bei-

## Vier Programmierschritte

### Erster Schritt: Worte definieren

Geben Sie ein:

```
: der      ." the " ;       ↵
: das      der ;           ↵ (das führt der aus!)
: den      der ;           ↵
: Hund     ." dog " ;      ↵
: beisst   ." bites " ;   ↵
: Kind     ." child " ;   ↵
```

### Zweiter Schritt: Worte testen

Eingabe: der Hund beisst das Kind  ↵

Ausgabe durch Forth: (Das wird nicht verraten. Testen Sie selbst!)

Ein weiterer Test:

Eingabe: das Kind beisst den Hund  ↵

Ausgabe: (...die haben Sie jetzt hoffentlich erwartet!)

### Dritter Schritt: Worte in neuen Worten verwenden

Geben Sie nun ein:

```
: oft      cr der Hund beisst das Kind cr ;  ↵
: selten   cr das Kind beisst den Hund cr ;  ↵
: beides   oft selten ;  ↵
```

(**cr** setzt den Cursor an den Anfang der folgenden Zeile)

### Vierter Schritt: Worte anwenden

Geben Sie ein:

```
oft       ↵ Forth antwortet: (...na was wohl?)
selten    ↵ Forth antwortet: ??
```

bringen und der Ihr spezielles Problem zu beschreiben in der Lage ist, sollte nicht Programm, sondern treffender Applikation genannt werden. Wie Sie mit einer Handvoll Legosteinen nahezu beliebige Gebäude stecken können, können Sie in Forth aus selbstgefertigten Bausteinen die vielfältigsten Applikationen erzeugen.

Als kleine Übung wollen wir nun als Applikation einen Übersetzungsautomaten Deutsch/Knüp-pel-Englisch programmieren. So soll auf die Eingabe des Satzes:

Der Hund beisst das Kind der Computer mit the dog bites the child antworten. (Programmierer,

die sich in anderen Sprachen auskennen, empfehlen wir, dieses Problem als Vorübung in "ihrer" Sprache zu programmieren.) Die typische Vorgehensweise in vier Schritten für die Programmierung in Forth ist in Abb. 2 dargestellt.

Sie sollten die wenigen Zeilen des Beispiels in Abb. 2 nun eingeben, um sich mit den einzelnen Schritten vertraut zu machen.

Die Worte, die Sie damit zum bestehenden Wortschatz hinzugefügt haben, Programme zu nennen, wäre übertrieben. Alle zum Problembereich des Übersetzungsautomaten gehörenden Worte bilden aber eine Applikation.

## Eine Applikation bearbeiten

Wenn Sie sich beim Abschreiben der kleinen Beispiele verschrieben haben sollten und nach Betätigung von  ↵ eine Fehlermeldung erhalten, oder

wenn das Wort nicht das macht, was Sie erwarteten, können Sie das bereits dem Wörterbuch zugeführte Wort prinzipiell nicht mehr bearbeiten oder korrigieren. Mit `see` und dem folgenden Namen eines Wortes können Sie sich nur anschauen, wie das "Rezept" dieses Wortes aussieht. Sie können aber ein fehlerhaftes Wort wieder aus dem Wörterbuch entfernen, indem Sie `forget` und den betreffenden Namen eingeben.

Sollten Sie die o.g. Vokabeln eingegeben haben, schreiben sie nun:

`see selten`   um zu sehen, wie das zuletzt beigebrachte Wort definiert wurde. Danach "sagen" Sie:

`forget mayer`  
  (...oder wie immer das *erste Wort* hieß, das Sie Forth beigebracht haben.),

dann werden Sie sehen, daß das Wort *Mayer* und alle folgenden Worte aus der Wörterliste verschwunden sind. (Wenn das nicht der Fall ist, waren Sie nicht folgsam und haben irgendetwas anderes als erstes eigenes Wort eingegeben! Dann wenden Sie jetzt auf dieses Wort das `forget` an!)

Was damit erreicht ist? Nun, jetzt können Sie alle Worte erneut eintippen.....!

Halt, schlagen Sie jetzt nicht aus Wut Ihren Computer. Wir wollen Ihnen einen anderen Weg zeigen, wie Sie auch in umfangreichen Applikationen Korrekturen vornehmen können, ohne jedesmal den Text neu eingeben zu müssen.

Den Vorgang des Korrigierens oder Bearbeitens eines Programmtextes (Quellcode) nennt man *Editieren*, und das dazu verwendete Hilfsprogramm nennt man *Editor*.

Was jetzt kommt, gilt nur für eine spezielle Forth-Version (F-PC)! (In anderen Forth-Versionen müssen Sie in das Handbuch schauen und unter EDITOR nachschlagen!)

Betätigen Sie die Tasten  ESC und  ↵. Es erscheint ein Menü, aus dem Sie mit den Pfeiltasten oder dem großgeschriebenen Buchstaben auswählen können, was Sie machen wollen. Ge-

ben Sie nun `N` ein, um eine *neue Datei* anzulegen! Forth antwortet mit:

`filename to create?`

Geben Sie hinter diesem Satz ein:

`uebung`  ↵.

Der Editor startet und sie befinden sich auf der ersten Seite einer leeren Datei. Nun können Sie die oben besprochenen Beispiele eingeben.

## Eine Applikation speichern

Wenn Sie nun die kleinen Programmierbeispiele mit Hilfe des Editors in die Datei `uebung.seq` (Die Namens-erweiterung `.seq` wird durch Forth

automatisch angehängt!) schreiben und dann den Editor durch Betätigung von  F10 wieder verlassen, wird Ihre erste Forth-Applikation auf der Diskette, bzw.

auf der Festplatte unter dem angegebenen Dateinamen (`uebung.seq`) gespeichert.

## Eine Applikation laden

Wenn Sie allerdings danach in das Wörterverzeichnis schauen (mit `words`), werden Sie die im Editor definierten Worte nicht finden. Sie müssen Forth noch dazu bewegen, das, was Sie in die Datei `uebung.seq` geschrieben haben, zu "lesen" und alles genau zu befolgen; also auch die neuen Worte zu lernen, als hätten Sie ihm dies im Dialog mitgeteilt.

Diesen *Lesevorgang* starten Sie bitte mit `load`. Sollte nun Forth während des Ladens (Lesens) einen Fehler entdecken, d.h etwas nicht verstehen, dann startet es selbständig den Editor und zeigt Ihnen mit dem Cursor an, wo es über einen Fehler gestolpert ist. Der Cursor steht immer unmittelbar vor der Stolperstelle.

Logische Fehler kann Ihnen Forth auf diese Weise natürlich nicht zeigen. Sie wissen ja: Ein Computer macht immer das, was Sie ihm sagen, aber häufig nicht das, was Sie von ihm wollen.

Stellen Sie also fest, daß ein Wort nicht das leistet, was Sie erwartet haben, so können Sie einfach mit `ed` aus

der Dialogebene des Forth heraus wieder in den Editor gelangen. Sie finden sich häufig an der Stelle Ihres Textes wieder, an der sie sich beim Verlassen befanden.

Berichtigen (debuggen) Sie nun Ihren *Quellcode* (so nennt man Text, der ein Programm für eine Programmiersprache enthält) und lassen Sie ihn erneut durch Forth laden.

So, an dieser Stelle sollten Sie sich nun Beispiele aus dem Buch "Programmieren in Forth" vornehmen und sich gleichzeitig mit der Programmiersprache Forth und Ihrem Forth-System vertraut machen.

## Eine Applikation starten

In den meisten anderen Programmiersprachen ist eine Applikation immer ein *geschlossenes Programm*, das durch Eingabe des Programmnamens aufgerufen wird. In Basic können sie das Programm durch `RUN` starten. *In Forth sind alle Worte gleichberechtigt!* Es gibt nur ältere und jüngere Worte, aber i.a. keine Worte, die man nicht sofort im Dialogmodus ausprobieren könnte. Somit ist auch das Wort, das unsere Applikation startet, nichts anderes, als ein Wort unter anderen Worten.

Nutzen Sie immer die Möglichkeit, ein Wort sofort auszutesten, bevor Sie es in weiteren Worten verwenden. Die Fehlersuche wird durch diese Methode drastisch verkürzt!

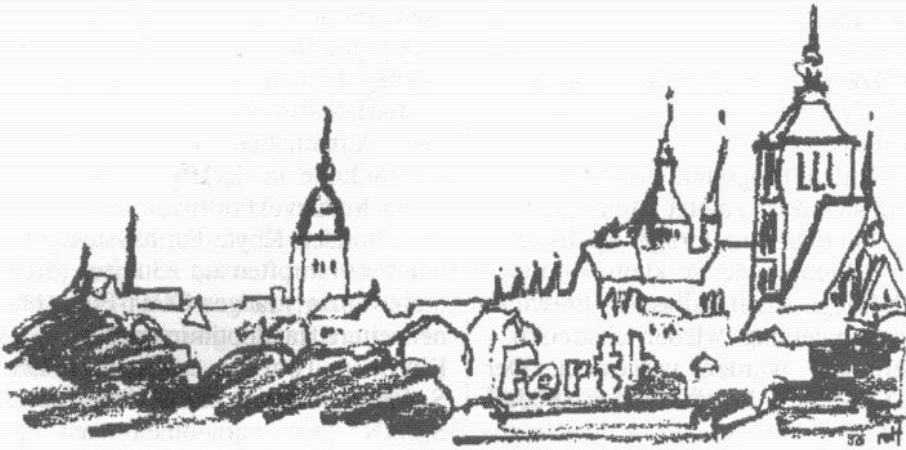
Nun können Sie den Laufstall verlassen und sich in der weiten Forth-Welt umsehen.



### Bezugsquellen für F-PC

**F-PC(original)**  
mit Erweiterungen und deutscher Anleitung:  
Jörg Staben  
Hagelkreuz 23  
4010 Hilden  
☎ 02103-240609

**F-PCakg**  
mit umfangreichen Modifikationen und erweiterten Hilfen z. T. in deutsch:  
Arndt Klingenberg,  
Straßburger Straße 12  
5110 Alsdorf  
☎ 02404-61648



## Tagung '92 in Rostock

von Michael Kalus

Das Motto der Tagung war zwar nicht offiziell vergeben worden, lag aber offenbar in der Luft: "Forth optimieren und spezialisieren". Der Bericht nennt Einzelheiten.

Wochenende, 27. März, 14:00 Uhr, leichte Hektik in Malente, Schleswig-Holstein, denn zu dieser Zeit begann die Tagung in Rostock bereits. Doch ich konnte mich nicht eher von den beruflichen Pflichten befreien. Die Strecke nach Rostock mit dem Auto über Lübeck und Wismar war gut westlich verstopft mit zäh fließendem Straßenverkehr, und so dauerte es doch recht lange bis wir ankamen - gerade noch rechtzeitig zum Vortrag von Ulrich Hoffmann, der bei mir mitgefahren war.

Rostock zeigte sich regnerisch-be-deckt, kalt und windig. Das machte die Trabantenstadt, in der unser Tagungshotel lag, noch trister. Reihenweise langgestreckte klotzige Wohnblocks aus Betonkästchen, reizlos aufgereit auf der Wiese. Urbanes Wohnen - genauso scheußlich wie manche Vorstadt im Westen und ebenso zerfallend. Gut, wenn schlechter Beton so schnell zerbröselt.

Das Congresshotel war da besser in Schuß, wenn auch aus gleicher Zeit und in ähnlichem Stil erbaut. Im Inneren dann Komfort in großzügigen Räumen und Hallen - gute Unterbringung. Die Zimmer waren offenbar

kürzlich renoviert worden und in recht anständiger Verfassung, ausgestattet mit allem was man so braucht und darüber hinaus mit allem, was man nicht braucht, wie Video, Fernseh, Telefon und Minibar. Na bitteschön!

Alle Teilnehmer waren im Hause untergebracht - bis auf die aus Rostock natürlich - so daß die Leute gut zusammen fanden. Man merkte dies am Sonntag bei der Mitgliederversammlung. Die Gruppe war gut gelaunt und

harmonisch gestimmt. Überhaupt war die Tagung gut vorbereitet. Insgesamt eine runde Sache. Ein Lob dem Rostocker Team von FORTECH, unseren Gastgebern.

Die Tagung verlief wie geplant, der Zeitplan wurde gut eingehalten, und bot genug Pausen. Schon Freitag um Zwei ging es los. Vortragsblock mit drei Beiträgen, Pause usw. Die Mahlzeiten konnten mit genügend Zeit gemeinsam eingenommen werden und waren ausgezeichnet - mit einem üppigen Frühstücksbuffet und mittags wie abends warmer Küche wurden wir verwöhnt.

Die Themen der Tagung und die Aussteller habe ich nachstehend übersichtlich aufgelistet. Einzelheiten können im Tagungsband nachgelesen werden, den es beim Forthbüro gibt. Er enthält die gekennzeichneten Beiträge.

Das Motto der Tagung war zwar nicht offiziell vergeben worden, lag aber offenbar in der Luft. "Forth optimieren und spezialisieren" war vielen ein Anliegen. Schon die Ausstellung zeigte Spezielles: Meßdatenerfassung, RTX-Workstation, Fuzzyregler und KI, SMAN, Big-Forth, F68K und F-PC und zwischendurch den FRP1600 - sozusagen ofenfrisch.

In den Vorträgen dazu war Erstaunliches herauszuhören. Alle schienen sich einig zu sein darüber, daß nur spezialisierte Forthsysteme gute Forthsysteme seien. Und so gab es friedlich nebeneinander eine Vielfalt an Systemen und Themen: Kleine



schlanke Kilobyter und dicke große Megabyte, Microcontroller und Datenbanken, Systeminnereien und Programmierumgebungen. Jedes für sich selbstverständlich optimiert und spezialisiert für den besonderen Einsatzbereich - ANS sei Dank. Von einengenden Standards war nicht mehr die Rede.

Forth und seine Benutzer wirkten ausgesprochen selbstbewußt, dabei selbstkritisch und klar. So ließen etwa die Leute von FORTech, Rostock, ihre comFORTH-Muskeln spielen - in der Ausstellung und mit einem dicken KI-Paket. Fuzzytools und KI - die Leidenschaft von *Birgit Steffenhagen*: "Die modular aufgebaute KI-Toolbox für das System comFORTH2 ermöglicht den gleichberechtigten Gebrauch prozeduraler und deklarativer Programmierverfahren. Es ist ein Werkzeugkasten mit konfigurierbaren Modulen für Listenverarbeitung, Backtrackingmechanismen für unbestimmte prozedurale Verzweigungen, Mustervariablen und entsprechende Unifikationsalgorithmen, Regelcompiler, Inferenzmechanismen, Verarbeitung von Prädikatsymbolen, Erklärungskomponente, Darstellung und Verarbeitung von unscharfem Wissen. Dabei wurden Erfordernisse für den Einsatz von wissensbasierten Komponenten bei der Lösung von Steueraufgaben wie Online Kopplung mit dem Prozeß, Zeitabhängigkeit von Inferenzprozessen und Wissensbasis, Verarbeitung unterschiedlichster Wissensarten, Interaktivität, Lernfähigkeit, geringe Systemgröße und hohe Arbeitsgeschwindigkeit berücksichtigt. Geschaffen für Steuerungsaufgaben gestatten die Tools es, konventionelle Steuerungssoftware und heuristische Methoden sinnvoll zu kombinieren."

Ebenso beeindruckend fand ich das Paket ums KK-Forth von *Klaus Kohl*, im praktischen Einsatz zu sehen im Meßdatenerfassungssystem im MC-32. In seinem Vortrag hat er dieses Forth näher vorgestellt.

Die Themen gingen dann mehr zur Spezialisierung eines Forthsystems auf verschiedene Prozessoren über. Auch *Egmont Woizel* und *Klaus Gungl* gingen näher auf Fragen der

Portierung und Microcontrollerprogrammierung ein.

Messen, regeln, automatisieren war ein Themenbereich. Systemfragen ein anderer. Optimierung war das Schlagwort.

In großen Programmierumgebungen stellt sich doch zuletzt immer die Frage, wie man eine fertige Applikation schließlich wieder so klein wie möglich kriegt. Allen Ballast der Entwicklungsumgebung wieder abstreifen - "Stripper" wurden vorgestellt. Der TCOM zum F-PC, Methoden im comForth, big-Forth und F68k wurden gezeigt oder diskutiert. Zwei Aspekte wurden näher beleuchtet. Zum einen Methoden, um zu niedrigem Spei-

cherplatzbedarf der Applikation zu kommen und zum anderen die Laufzeitoptimierung. TCOM von *Tom Zimmer* brachte in einem spontanen Testprogramm ein sehr gutes Ergebnis: Ausschalten der NumLock-Leuchtdiode in der PC-Tastatur mit einem highlevel Forthwort und so etlichen hundert Kbyte Forthsystem darunter schrumpften auf 23 Bytes (dreiundzwanzig!) langes COM-File - ohne weitere Handoptimierung!

Ein neues Tool gab es zu sehen: Die Softwaremanagement-Umgebung SMAN. Dem vernehmen nach geschrieben in Forth zur Verwaltung und Handhabung umfangreicher Quelltexte und ihrer verschiedenen

### Tagungsprogramm der Forth-Tagung 92 in Rostock

(\*) Beitrag ist im Tagungsband enthalten

#### Freitag, 27.03.1992 ab 14:00

Begrüßung

- \* Forth in der Meßtechnik (Klaus Kohl, Ing.büro)
- \* Portable Generierung von Forth-Systemen (Egmont Woitzel, FORTech Software GmbH)

Pause

- \* KK-Forth - Ein einheitliches Forth für verschiedene Microcontroller (Klaus Kohl, Ingenieurbüro)
- \* Erfahrungen bei der Entwicklung einer Super 8 Applikation (Thomas Beierlein, IH Mittweida)
- \* Der Controller Modul TDS 2020 mit Hitachi H8/532 (Arndt Klingelberg)
- \* Fuzzyregler für Anfahrsteuerung (Birgit Steffenhagen)

Pause

Workshop: Applikations-Compiler (Ulrich Hoffmann)

Abendbrot

#### Sonnabend, 28.03.1992 ab 09:00

F-PC - Stand der Dinge (Jörg Staben)

F-PC/TCOM als universelle Forth-Entwicklungsumgebung (Arndt Klingelberg)

Die sprachunabhängige Programmierumgebung (Frank Horst Raschke)

Pause

FRP 1600 (Klaus Schleisiek-Kern)

Pause

- \* Forth als Programmiersprache für Geschäftsprogramme (Ekkehard Flögel)
- \* 68k-Forth (Bernd Paysan, 6p Soft)

Mittag

Ausflug nach Warnemünde (Seeluft schnuppern)

- \* Non-Forthmaschinen (Ulrich Hoffmann, Universität Kiel)
- \* Optimierende Compilation von Fadencode (Udo Schütz, Universität Rostock)
- Microcontrollerprogrammierung mit Forth (Klaus Gungl)

Abendbrot

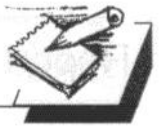
- \* Automation eines Sägewerks mit Forth (Ralf, Neuthe)

#### Sonntag 29.03.1992 ab 09:00

Mitgliederversammlung der Forthgesellschaft

Mittag

Abreise



### Aussteller der Forthtagung '92 in Rostock

#### Meßdatenerfassungssystem MC-32,

Graphische Aufbereitung (PC)

Klaus Kohl

Firma Dr.Schettler BMC, Puchheim/München

#### Forth-Workstation (RTX2000)

Dr. Olaf Haeggquist

Wickede/Ruhr

#### Schlupfkontrolle

Beispiel für eine Fuzzyregelung mit comFORTH (PC)

Birgitt Steffenhagen

Firma FORTEch, Rostock

#### SMAN

Softwaremanager, Quellenverwaltung (PC)

Frank Stüß

Frankfurt

#### BigForth

Forthsystem für 68000 (Atari)

Bernd Paysan

München

#### F68k

Forthsystem für 68000 (Atari)

Jörg Plewe

Mühlheim/Ruhr

#### Forthprozessorkarte (FRP1600)

Klaus Schleisiek-Kern

Firma Delta-T, Hamburg

Und schließlich ging es auch diesmal wieder runter bis in die Maschinenebene. Erfahrungen wurden ausgetauscht über die Programmierung des Super-8 und des H8/532 von Hitachi. Und noch tiefer hinein in Fragen des Prozessor-Design ging es: *Klaus Schleisiek-Kern* stellte den 16Bit Forth Risc Processor FRP1600 vor, der jüngste Sproß in der Familie der echten Forthprozessoren. Optimiert für schnelle Gleitkommaverarbeitung, Bildverarbeitung und Interruptbehandlung scheint dies ein vielversprechender Forthchip zu werden. Wir werden davon sicher noch einiges hören. Die Einführung des Produktes erfolgte kürzlich auf der Hannovermesse.

Und das genaue Gegenteil dazu - Prozessoren, auf denen kein Forth implementiert werden kann - stellte *Ulrich Hoffmann* vor: Die Microcontroller ST6x der Firma SGS-Thomson. Zu guter letzt gabe *Ralf Neuthe* und *Dr. Woizel* Einblicke in ein größeres Forthprojekt, das sie kürzlich betreut haben: Die Automation eines Säge-

werkes. Alle, die wissen möchte, warum man so etwas am besten in Forth macht, ist dieser Beitrag wärmstens zu empfehlen; nachzulesen im Tagungsband (demnächst auch in der VD). Kurz aber humorvoll lernt man darin die "Feinheiten" bei groben Holzarbeiten kennen und was sich hinter so einer harmlosen Aussage wie "asynchrones Ereignis" verbirgt, wenn kurzes Holz im Spiel ist!

Zu guter letzt dann gehörte der Sonntag wieder dem Verein. Wichtige Entscheidungen standen zur Abstimmung. Die Beschlußfassung ging diesmal erstaunlich glatt. Nach dem Finanzbericht und der Aussprache wurde das Direktorium neu gewählt: *Thomas Beierlein*, *Ulrich Hoffmann*, *Jörg Staben*. Ferner wechselte die Redaktion der 'Vierten Dimension' nach Baesweiler / Alsdorf / Aachen in die Hände von *Rolf Kretzschmar*. Die Diskussion über die Zeitung zeigt überwiegend hohe Erwartungen der Mitglieder an "ihr" Blatt. Nachdem Vortrag von *Rolf Kretzschmar* über sein Konzept für die VD und seine technischen Möglichkeiten hatte ich das Gefühl, vorsichtiger Optimismus ist angebracht: Vielleicht ist die Redaktion jetzt endlich in den richtigen Händen.

Sicher in den richtigen Händen ist seit Jahren das Forth Büro bei *Ulrike Schnitter* in München. Daher hat es mich ganz besonders gefreut, daß sie in diesem Jahr mit dem SWAP-Drahten ausgezeichnet worden ist - Standing Ovations und unser aller Dank an Dich, Ulrike!



Versionen. Ich hab es leider nur kurz zu sehen bekommen und die Demo ging mir zu schnell. Aber das war wohl Zweck der Übung. So flink ist man damit eben in der Erstellung von Applikationen aus seinen Toolboxes. *Frank Stüß* und *Frank Horst Raschke*, Frankfurt, geben sicher gern nähere Auskünfte.

#### Anmerkung zu 'Light-Show' (Seiten 10-12)

Dem Target-Kompiler TCOM fehlt vernünftigerweise der Interpreter, sonst hätten wir ja wieder ein F-PC. Allerdings kann man auch ein (wesentlich größeres) Target mit Interpreter erzeugen. Es passiert allerdings wenig Sinnvolles, wenn die Worte auf der commandline dem Interpreter gar nicht bekannt sind.

Ein Beispiel wie TCOM (und F-PC) die DOS-Kommandozeile übernehmen können und dann auswerten, werde ich im Folgeheft am Beispiel meines Programmes NOW bzw. NUN erklären. DOS-übliche, mehrfache Optionen mit vorgestelltem '/' werden ausgewertet und das Datum und/oder Uhrzeit in allen möglichen und unmöglichen Spielarten dargestellt. Es ersetzte ein C-Programm, das ohne Optionsvielfalt und ohne Helpscreen ähnlich groß war.

akg

# Programmier-Wettbewerb '92

Bericht von Klaus-Peter Schleisiek

Zum 3. Male fand die jährliche Messe ECHTZEIT in Sindelfingen statt, und - wie jedesmal - gab es dabei einen Programmier-Wettbewerb. Bei diesen Wettbewerben geht es darum, per Computer ein elektromechanisches Modell zu einem bestimmten Verhalten zu bringen, wobei nicht nur der Computer seine Echtzeit-Fähigkeit zu beweisen hat, sondern auch die Programmier-Teams die Eignung ihrer Werkzeuge und - nicht zuletzt - ihre persönliche Fitness demonstrieren sollen.

Die Sieger des 1. Wettbewerbs - FORTECH Software GmbH mit ihrem comFORTH - hatten auch beim 2. Mal gewonnen. Dieses Mal hatten sie das Modell gebaut; darum konnten sie nicht mitmachen.

## Das Modell

Das Modell dieses Jahres heißt BUNTE BOX (Abb. 1); es ist dargestellt durch seine Vorder- und Rückwand, nämlich zwei rechteckige Platten mit vier Abstands-Bolzen dazwischen. Bunt ist eine Kreis-Scheibe im Inneren mit drei gleichgroßen Farbsektoren: rot, grün und gelb. Durch ein Loch in der Vorderwand ist jeweils eine dieser Farben zu sehen, sofern die außerdem vorhandene Lochblende es gestattet. Die Lochblende wird gebildet durch eine schwarze Kreis-Scheibe mit vier kreuzweise angeordneten runden Löchern. Gedreht wird diese Scheibe durch einen 2-Phasen Schrittmotor, für den 2 Steuer-Leitungen an den Schnittstellen-Stecker geführt sind. Die bunte Scheibe rotiert dank eines unregelmäßigen Gleichstrom-Motors; eine Kerbe am Umfang gibt bei jeder Umdrehung einen Impuls auf die einzige Signal-Leitung der Schnittstelle. Die BUNTE BOX zeigt

also dann eine der drei Farben allein, wenn ein Viertel Blenden-Drehzahl gleich der Farbscheiben-Drehzahl ist. Die Auswahl der gezeigten Farbe wird durch den Phasenwinkel zwischen beiden Drehungen bestimmt.

## Die Aufgabe

Die Aufgabe bestand darin, mit der BUNTEN BOX eine Verkehrsampel mit folgendem Farbwechsel nachzu-

bilden: gelb 2 sek., rot 8 sek., gelb 2 sek., grün 8 sek. .

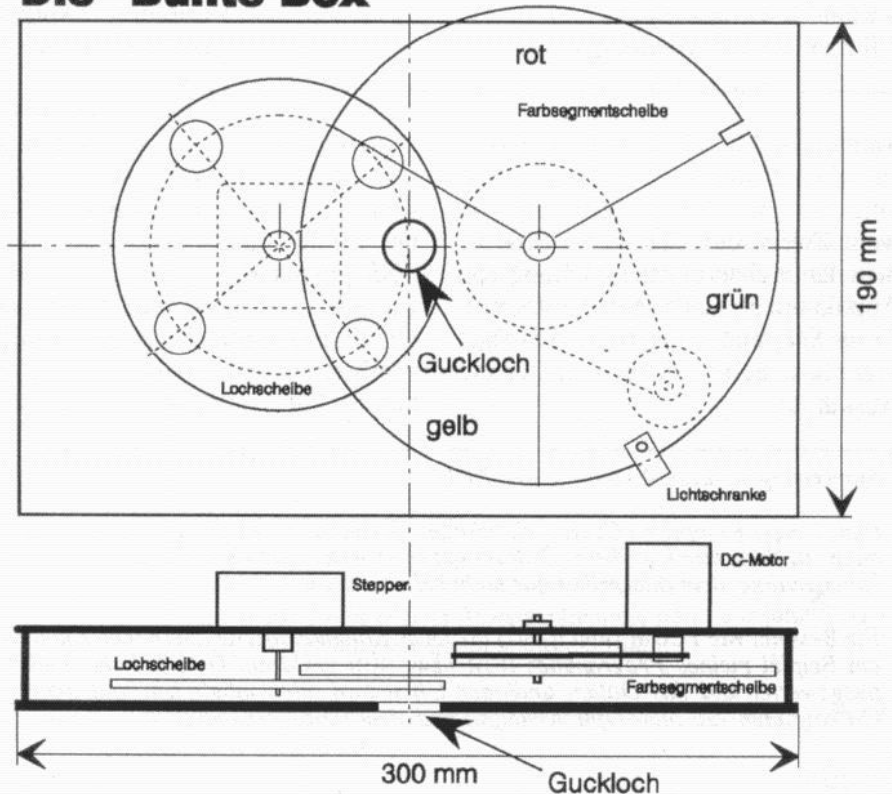
Die erforderlichen Teil-Aufgaben sind also:

- zwei Rechteck-Schwingungen mit Tastverhältnis nahe 50% und Phasenlage bei 90 Grad zueinander auf die beiden Steuerleitungen zu geben,
- Die Frequenz dieser Schwingungen nur so sanft zu ändern, daß der mit großem Trägheits-Moment belastete Schrittmotor folgen kann,
- eine bestimmte Phasendifferenz zwischen Schrittmotor und Farbscheibe zu erreichen und für vorgegebene Zeit konstant zu halten. Die in der BUNTEN BOX erreichten Drehzahlen stellen hinsichtlich der Geschwindigkeit keine hohen Anforderungen.

## Die Teams

Die meisten der 10 Teams hatten offenbar andere Sorgen, als die auffällige Darstellung ihrer gewählten Sprache und Ausrüstung. Immerhin zeigte die Forth-Gesellschafts-Alt direktoren-Riege Flagge: Ein Blatt Papier in DIN A4 auf der Monitorrückseite klärte die Beobachter kurz über Soft-

## Die "Bunte Box"



### Stichworte

Echtzeit-Messe,  
Programmierwettbewerb



und Hardware auf. Arndt Klingelberg hatte seinen Tisch mit Hinweisen beklebt und achtete darauf, daß keine nassen Jacken darauf landeten. Bei anderen konnte man das Glück haben, auf gelegentlich zugeklappten Büchern C++ oder RTOS zu lesen. Ansonsten waren die Beobachter aufs Raten angewiesen, da störendes Fragen den gestreßten Wettkämpfern wohl kaum zuzumuten war.

Die 10 Teams traten mit folgenden Sprachen an:

- 2 mal Assembler
- 2 mal C
- 3 mal Forth
- 1 mal Pearl
- 1 mal Turbo Pascal

Für Forth fochten:

- F1: J. Reilhofer, K. Schleisiek-Kern, H. Schnitter mit FRP 1600
- F2: A. Klingelberg mit F-PC auf 386er-Laptop
- F3: Heinz, Weiggaard, Thie auf IBM-AT

## Der Verlauf

### Nach der 1. Stunde:

Ein Team hatte schon zwei BUNTE BOXEN verbraucht, jedoch gelang an einer davon die Reparatur. Bei zwei Forth-Teams bewegten sich hin und wieder die Schrittmotoren.

### Nach der 2. Stunde:

Wohl jede der Lochblenden hatte sich

schon unzufällig bewegt. Bei dem Team F1 (s.o.) war bereits flotte Rotation der Lochblende zu beobachten, wenn auch noch nicht mit stabilen Farben. Ich hätte wetten mögen ...

### Nach der 4. Stunde

war ich ganz froh, doch nicht gewettet zu haben. Die vorgesehene Wettkampf-Dauer war erfolglos verstrichen; die Fast-hab'-ich'-s-Stimmung war einer vertiefenden physikalischen Betrachtung mit entsprechenden Debatten und Papierverbrauch gewichen. Wo sich die Lochscheiben noch drehten, waren häufig die Finger im Spiel. Ist Forth vielleicht doch nicht so enorm überlegen? Schließlich, nach viereinhalb Stunden:

## Die Sieger

Das Team von *Hubert Jäger* mit C auf Tranputer-bestückter PC- Erweiterungs-Karte bittet um Aufmerksamkeit für eine Vorführung; kann das als Lösung gelten? Es kann! Nach der Vergabe des 1. Platzes wird die nächstbeste Annäherung an die Lösung gesucht. *Arndt Klingelberg* bietet eine, die sich vom Vorbild effektiv nur durch schwächere Farben infolge langsamer Lochblenden-Drehung unterscheidet. Damit erringt er - als Ein-Mann-Team mit Forth - den 2. Platz. Der 3. Platz wird dem zugebilligt, der wenigstens eine Farbe stabil zeigen kann. Bis zum 6. Platz mußte der gestreßte Wettkampfleiter sich durchkämpfen; denn es gab tatsächlich nennenswerte Gewinne zu verteilen, die von verschiedenen Seiten gestiftet worden waren! Darunter war auch eine BUNTE BOX. Die übrigen BUNTEN BOXEN konnten anschließend käuflich erworben werden.

## Fazit

Lange hat dieser Wettkampf gedauert; wesentlich länger als veranschlagt. Das gab Zeit zu interessanten Unterhaltungen, u.a. zum Thema: "War die Aufgabe vielleicht zu schwer?" Nach Ansicht des Aufgabenstellers hätte die gestellte Aufgabe - bei geeigneter Vorbereitung - in vier Stunden zumindest mit Forth gelöst werden können. Nach der den Teams bekannten Schnittstellen-Beschreibung war eine Schrittmotor-Steuerung mit gleichzei-



tiger Zählung des Eingangs-Signals durchaus nicht überraschend; jedenfalls nicht für Echtzeit-Programmierer. Geeignete Vorbereitung wäre demnach, außer Netzteil und Stecker auch ein paar passende Worte ins Gepäck zu stauen. Daran aber fehlte es den Teams offensichtlich.

Darum, liebe Wettkampf-Aspiranten, nehmt euch die Erkenntnis von FORTECH zu Herzen: Forth allein reicht nicht; man muß sich auch richtig vorbereiten! Und Flagge zeigen, finde ich!

Die nächste Gelegenheit ist auf der ECHTZEIT 1993 vom 15. bis 17. Juni in Karlsruhe. Es wird noch dringend ein Ausrichter und Modellentwickler gesucht. Welches Institut oder welches Unternehmen möchte sich dieser publicityträchtigen Aufgaben annehmen? Bitte melden Sie sich möglichst zeitig beim Veranstalter (L. Drebingner GmbH, Destouchesstr. 16, W-8000 München, Tel. 089-33 30 33) oder beim Forth-Büro.



# Echtzeit '92

von Arndt Klingelberg

Echtzeit mit Public Domain und Industrie Standard ?  
Auf der Echtzeit'92 reichte es für den 2. Platz.  
Die Vorbereitung dazu: Digital I/O und Timing.

Die bereits vor dem Programmierwettbewerb bekannten Anforderungen waren: Die 'Bunte Box' hat zwei digitale Eingänge und einen Ausgang. Die Folgefrequenz des abgegebenen Low-Pulses sollte max. 20 Hz betragen.

Bestritten habe ich den Wettbewerb mit Forth und zwar mit F-PC-ak9 (der von mir korrigierten und erweiterten Version des F-PC). Als Hardware diente ein ISA Rechner mit 386SX 20 MHz in Lunch-Box-Portable Format. Fast alle ATs und die besseren PCs sollten sich für die Bunte Box eignen, exakte Verzögerungen über Software verbieten ganz langsame Krücken. Die RTC (Real-Time-Clock) war für meine Lösung zwingend erforderlich. Ich bereitete ein paar Worte für den Centronics-Printer Port vor. Zusätzlich erstellte ich noch ein etwas günstigeres Wort für kurze, feinstufigere Zeitverzögerungen. Andere sinnvolle Erweiterungen sind in F-PC-ak bereits enthalten. Und wie das so geht, hatte ich zwar ein Kabel für die BunteBox möglichst sorgfältig 'bereits' in der Nacht So.-Mo. erstellt, die I/O- und Delay Worte entstanden dann allerdings bereits nach Mitternacht Mo.-Di und die letzten Modifikationen tippte ich Mi. 8:15 auf der Bettkante sitzend, also knapp 5 h vor dem Start.

## Stichworte

F-PC  
IBM-AT  
BIOS  
Port  
Centronics IBM  
LPT  
Digital I/O  
Timer  
Echtzeit

## Printer Port - das unbekannte Wesen

Den 'Centronics'-Port hat IBM auf ein 25 poliges Sub-D Weibchen gelegt, das ist billiger, kleiner, besser zu

verarbeiten als der Centronics Stecker. Und da der serielle Port bei IBM -- immerhin konsequent -- männlich war, funktioniert das sogar in der Praxis. Nur, die exakte Steckerbelegung usw. findet man nirgendwo. Nachfolgende Informationen ( siehe auch Quelltext ) entstammen verschiedensten Quellen und eigener Printboard-Erforschung. Wer weiß mehr?

Der Printer-Port hat wohl bis zu 12 Ausgänge und bis zu 9 Eingänge. Ich habe mich nachfolgend auf 2 u. 1 beschränkt, dazu kannte ich auch die Pinbelegung.

Die Programmierung erfolgt entweder über MS-DOS, IBM-PC-BIOS ( es ist

### Listing zu: Echtzeit '92 (A. Klingelberg)

```
\ E292-VD.seq  Quellcode zum Programmierwettbewerb Echtzeit 92 \ ak92jun15
comment: -----
2 digital Outputs and 1 digital Input.
The input signal supplies a low-pulse, f < 20 Hz ,
The parallel-'CENTRONICS'-printer-port of an 'IBM'-alike will be used.
Written in F-PC ak-version
for other Forthes modify:
    TITLE:  VALUE &> ...  number ( binary )
    assembler notation
-----
comment;
ONLY FORTH ALSO DEFINITIONS  decimal
title: -ez92-vd-                \ this is an debugged ANEW
comment: -----
Centronics Printer Port IBM-style

interrupt
LPT1      7      3BC (378) ((278))      3BC usually for Hercules-Adapt.
LPT2      AT:5   378 (278) ((---))
LPT3      278 (---) ((---))

-----in Forth we use THESE numbers !!!
25p sub-D
pin      data      v
OUT      bit
$0378    2          1      0
          3          2      1
          4          3      2
          5          4      3
          6          5      4
          7          6      5
          8          7      6
          9          8      7

IN
$0379    15         3      -ERROR
          13         4      +SeLeCT
          12         5      +Paper Error
          10         6      -ACK
          11         7      -BUSY

Ground 18..25
-----
BIOS-call
INT $17   func 0      emit      \ all fetch 5 bit input
          1          init      \ " ^^^^^
          2          get status \ "

DOS-call
INT $21   func $40    string out
          $3D         init
          $05         emit
          Reading of inputs NOT possible !!!

\ sample from kernel2.seq
CODE PR-STATUS ( n1 -- b1 )
\ n1 is the printer number.      Return the printer status byte.
    POP DX                        \ PRINTER NUMBER
    MOV AH, # 2
    PUSH SI                       PUSH BP   \ 2 \
    INT $17                       \ SI BP should be preserved
    POP BP                         POP SI   \ 2 / by INT $17 \ ak92apr26
    MOV AL, AH
```





also nicht das wesentlich verbesserte PS/2 BIOS gemeint) oder direkt über die Port-Adresse. Da BIOS/DOS (?) hier um-map-pen, muß die jeweils gültige Portadresse ermittelt werden. Ich benutze hierzu SYSID. Die Adresse kann jeder allerdings auch selbst in FORTH aus dem BIOS-Variablen-Segment (\$0040), und zwar an 0040:0008 bzw. 0000:0408 (über 0 \$408 @L) abfragen. Hierzu half mir die Tech-Help Software.

Es gibt vier Möglichkeiten bits an den Centronics Ausgang zu schicken:

□ 1. HighLevel Forth mit EMIT redirected auf PRN-handle statt CON.

□ 2. DOS INT \$21

func \$40 string out  
\$3D init

\$05 emit

(Eingänge NICHT abfragbar!)

□ 3. BIOS INT \$17

func 0 emit

1 init

2 get status

alle diese Funktionen liefern 5 Bit zurück (Eingänge).

□ 4. direktes Lesen und Schreiben an die PORT Adresse, dieses wiederum in HighLevelForth (pC@ pC!) oder Assembler (IN OUT).

Das Lesen und Schreiben des Ports in Assembler war für mich die bevorzugte vorgehensweise, da ich den Assembler in F-PC-ak mit den von mir geschaffenen Online//Context-Hilfen sehr gerne nutze. Die Port-Assembler Lösung ist zudem die schnellste und universellste. DOS erlaubt ausschließlich ein Lesen des Ports, BIOS zusätzlich das Lesen von immerhin 5 bit. Die direkte Portansprache erlaubt 'alles'. Für Portabilität und Lerneffekt werden unterschiedliche Versionen vorgestellt. Die weitgehend am Standard orientierte Syntax im F-PC Assembler sollte eine leichte Umsetzung in andere Assembler erlauben.

Da -- so hatte ich angenommen -- die zwei Bits völlig unabhängig geschrieben werden mußten, wurde über eine Maske und AND und OR erreicht, daß nur das jeweilige Bit angesprochen wurde. Dazu wird das komplette Ausgabe-Byte in OUT-BYTE abgespeichert. Der Einfachheit halber wird als Stack-Vorgabe

TRUE == -1 == \$FFFF

gefordert, der flag-Ausgang ist dage-

Fortsetzung des Listing zu: Echtzeit '92 (A. Klingelberg)

```

MOV AH, # 0
LPUSH END-CODE
-----
comment;
\ ----- using BIOS
0 VALUE LPT1- \ use ( DOS number -1 ) LPT2 ==> 1
CODE pr-init ( -- f1 ) \ get flag (false=successful)
\ set value LPT-1 to the appropriate value
MOV AX, # $0100
MOV DX, &> LPT1-
INT $17
AND AX, -1111100000000000
lpush end-code
CODE pr-C!@ ( 8b -- 5b ) \ send 8bit to parallel port
\ get 5bit back with physical value
\ INT 17 inverts some of the bits
\ you have ot set LPT1- to the appropriate value
POP AX
XOR AH, AH \ zero AH
MOV DX, &> LPT1-
INT $17
XOR AX, # -0000000011001000 \ a '1' toggles bit
\ - ERROR
\ + printer selected
\ + out of paper
\ - printer acknowledge
\ - printer NOT busy
\ mask
AND AX, # -0000000011111000
lpush end-code
CODE pr-5b@ ( -- 5b ) \ get 5bit back
\ set value LPT1- to the appropriate value
MOV AX, # $0200
MOV DX, &> LPT1-
INT $17
MOV AL, AH
XOR AX, # -0000000011001000 \ a '1' toggles bit
\ - ERROR
\ + printer selected
\ + out of paper
\ - printer acknowledge
\ - printer NOT busy
\ mask
AND AX, # -0000000011111000
lpush end-code
: pr! pr-C!@ drop ; ( 8b -- ) \ send 8bit to parallel port
' pr-5b@ ALIAS pr@ ( -- 5b ) \ get 5bit from parallel port
\ ---- using port directly \ compile on the specific computer you are using
VARIABLE prn-addr
$40 8 @L ( $03BC ) ( $0378 ) ( $0278 ) prn-addr ! \ lpt1
\ we compile the address found during compiling
\ ----- poll INput bits \ bit 4 is SeLeCT pin 13
: in4-F ( -- f1 ) \ input bit 4 , Forth version
[ prn-addr @ 1+ ] Literal pC@ \ fast but fixed version
-00010000 AND ;
code in4 ( -- f1 ) \ input bit 4 , Code version
MOV DX, prn-addr \ you may change the address
INC DX \ prn-addr+1
IN AL, DX
AND AX, # -00010000
lpush end-code
\ ----- set OUTput bits
0 value out-byte \ remember the status of other bits
code out0 ( f -- ) \ serves pin 2
\ use -1==TRUE or -xx1x as a HIGH input
POP AX
AND AX, # -1
MOV BX, &> out-byte
AND BX, # -10
OR AX, BX \ overlay other bits
MOV &> out-byte AX \ remember new byte-value
MOV DX, prn-addr
OUT DX, AL
next end-code
code out1 ( f -- ) \ serves pin 3
\ use -1==TRUE or -xx1x as a HIGH input
POP AX
AND AX, # -10
MOV BX, &> out-byte
AND BX, # -1
OR AX, BX \ overlay other bits
MOV &> out-byte AX \ remember new byte-value
MOV DX, prn-addr
OUT DX, AL
next end-code
\ or to set both bit simultaneously without using a mask ( out-byte )
: PPORT-C!-F ( 8b -- ) \ Forth Version, -00011 will set both bit to HIGH
prn-addr @ pC! ;
code PPORT-C! ( 8b -- ) \ 00011 to set both bit to HIGH
POP AX
MOV DX, prn-addr
OUT DX, AL
next end-code
\ -----

```

sind nur einzelne Bits gesetzt. Wie das Leben so spielt, war allerdings ein Schrittmotor anzusteuern. Die beiden Bits mußten synchron geschrieben werden. So wurde dann all dieses im Wettbewerb neu geschrieben.

## Zeitverzögerungen

Längere Zeitverzögerungen  $> 0.1$  s sind einfach über die RTC ( RealTimeClock ) realisierbar. Sollen mehrere Zeischleifen überlagert werden hilft F-PC's ELAPSED allerdings nicht, hier sollte man auf TIMER: (ak-Version) oder P. Koziar's WAIT zurückgreifen.

Für kurze Verzögerungen helfen im üblichen IBM-PC nur DO...LOOP oder besser (weil weniger Balast) FOR...NEXT Schleifen. Im IBM-AT kann die RTC 'verbogen' werden und wesentlich kürzere 'Ticker' als -wie üblich- in 55  $\mu$ s Abständen liefern. Dieses hätte mir im Wettbewerb geholfen, da ich in F-PC-ak sehr einfach einen Interrupt auf dem TICKER installieren kann.

Über die Verbiegung der RTC als Kurzzeittimer hoffe ich demnächst in der VD mehr schreiben zu können. Da mir eine MilliSekunde zu grob erschien, wurde als zusätzliches Wort 50US (  $n * 50 \mu$ s Verzögerung ) geschaffen. Tom-Zimmer bestimmt während der Systeminitialisierung über INITSTUFF den individuellen Schleifenparameter FUDGE für die jeweilige Maschine, und zwar anhand der RTC.

Die original MS (= "Millisekunde) im F-PC taugt nicht viel, sie wurde daher in der ak-Version stark modifiziert. Einmal wurde sie genauer; allerdings erfüllt sie nun nicht immer die ANS-Forderung. Dies fordert, daß MS MINDESTENS  $n * 1$  ms verzögern muß (wieviel länger ist dagegen egal). Die original Version wird extrem ungenau bei Multitasking und -- was mich im Wettbewerb zur Verzweiflung gebracht hätte -- ist gänzlich unbrauchbar nach Aufruf des DEBUG-ers ( NEXT | wird langsamer ).


In der AK-Version wird FUDGE genauer ermittelt, und für DEBUG UNBUG umgeschaltet, bleibt also auch nach dem debuggen nutzbar ( wird der

## Fortsetzung des Listing zu: Echtzeit '92 (A. Klingelberg)

```
\ trigger-delay
: trigger+ ( -- )
  BEGIN pause-func in4 UNTIL ;
  \ doing something usefull while waiting

: trigger- ( -- )
  BEGIN pause-func in4 0= UNTIL ;

defer t-junk ' noop is t-junk \ not waisting all the time inbetween

: trigger++ ( -- )
  trigger- t-junk trigger+ ;
  \ this one: 

\ fast delay-loop
40 VALUE us-fudge \ FUDGE is a VARIABLE , US-FUDGE is a VALUE

: us-init ( -- )
  @ fudge 25 / !> us-fudge ;
  \ fudge changes when debugging !!! use US-INIT

us-init

\ us-init has to be chained into INITSTUFF ADEBUG UNBUG

: 50us ( n -- )
  \ 2DUP 0= ?EXIT 1- FOR us-fudge FOR NEXT NEXT ;
  FOR us-fudge FOR NEXT NEXT ;
  \ native version , delays one unit more than expected

\ not as precise as MS when using DEBUG UNBUG
```

debugger deaktiviert muß allerdings explizit UNBUG aufgerufen werden). Das 50US Wort bzw. US-FUDGE paßt sich nicht automatisch an, muß also über US-INIT jeweils angepasst werden. Entsprechend der FOR...NEXT Struktur und zur Vermeidung weiteren OverHeads wird auch bei 0 50US ca. 50  $\mu$ s verzögert. Später stellte sich noch heraus, daß die Initialisierungsroutine für MS bei Prozessoren  $\geq 386$  (386SX ?) falsche Werte liefert. ( Wahrscheinlich hat mich das zusammen mit einem nicht vorhandenen Oszilloskop die Zeit für den ersten Platz gekostet ). FUDGE muß also mit mehr Aufwand bedacht werden.

## Noch ein paar Tips

Forth hängt sich gerne auf ( exakt: ich hänge den Rechner gerne damit auf ). Bei nicht zu extremem Hängen hilft mir der Tastaturreiber K3 : über Ctrl-Alt-Break sende ich an DOS 'Programm kontrolliert beenden' (int \$21 func \$4C). Das hat mir im Wettbewerb kostbare BOOT-Zeit gespart. Wieder zu Hause probierte ich die BunteBox wieder aus und - oh Schreck - nichts funktionierte so wie gedacht. Ich verzweifelte an meinen Programier-'Künsten'. Streng nach Murphy ist immer die Komponente die Ursache, die über jeden Verdacht völlig erhaben ist: es war das große

102-er Keyboard. Das Programm NUMCLICK (gibt Tönchen beim Tippen auf dem NUMBLOCK) brachte bei der Bedienung des Sound-Portes das gesamte Timing durcheinander. Ich wußte nun endlich was ECHZEIT ist.

## Hinweise

SYSID K3 sind PublicDomain und finden sich im Komplettpaket F-PC-ak

Tech-HELP!  
Flambeaux Software, darf wohl in Zukunft zum Komplettpaket F-PC-ak dazukommen.

Um etwas mehr Experimentieren zu ermöglichen, wird in der Folgeausgabe der VD die Pervertierung des Centronics Port zu einem 8bit AnalogAusgang folgen (PulsWeitenModulation).

akg



# Neues aus der FG

## Projektförderung durch die FG

von Jörg Staben

Die Forth-Gesellschaft e.V. will stärker Aktivitäten unterstützen, die zur Verbreitung des Forth beitragen. Zu diesem Zweck sollen Forth-Systeme eine Art FG-Stempel erhalten, wenn sie bestimmten Anforderungen entsprechen.

Die FG ist eine Gemeinschaft von Leuten, die sich mit Forth beschäftigen. Leider tun sie das nicht gemeinsam. So kommt es, daß es in der fast 10jährigen Geschichte der FG erstaunlicherweise kein einziges vollendetes und nur zwei begonnene Projekte gegeben hat (1. Leibniz; ein Forth in deutsch; nicht zu verwechseln mit dem Leibniz von A. Goppold! 2. RTX2000-Karte; fertig, aber nicht als FG-Produkt! rk).

Sogar das volksFORTH als das übertragbare Forth der FG erweist sich bei näherer Betrachtung nicht als geplantes Projekt, sondern als ein Bündel nachträglich aufeinander abgestimmter Einzelleistungen. Daraus könnte man schließen, daß es zwar nicht unmöglich, aber zumindest sehr schwierig ist, gemeinschaftliche Projekte durchzuführen.

Erstaunlicherweise gibt es aber zumindest zwei umfassende Aktivitäten, die zwar nie als Projekte konzipiert wurden, denen es aber gelang, eine Vielzahl von Leuten für die Mitarbeit zu begeistern. Bezeichnenderweise sind beides Forth-Systementwicklungen,

da nur hier die Erweiterung in jede beliebige Richtung möglich ist. Im Anwendungsbereich dagegen wäre die unkoordinierte Zusammenarbeit verschiedener Leute wohl gar nicht möglich.

Das F-PC ist unzweifelhaft auf dem PC zum Wegbereiter für Forth geworden. Kein anderes System hat bisher eine vergleichbare Ausbreitungsgeschwindigkeit innerhalb der Forth-Welt und/oder eine solche Aufmerksamkeit außerhalb erreicht. F-PC ist Gesprächsthema.

Auch über das volksFORTH ist gesprochen worden, zumindest eine zeitlang. Viele Leute haben Programme für und mit diesem System geschrieben und diese auch veröffentlicht. Trotzdem ist das volksFORTH auf dem Stand von April 1988 stehen geblieben. Das F-PC dagegen entwickelt sich stürmisch weiter.

Beides sind Forth83-Systeme, beide sind public domain, gleichermaßen vollständig, dokumentiert und leicht verfügbar. Wie konnten sich zwei so gleiche Systeme so unterschiedlich entwickeln?

Nun - beim volksFORTH ging die eigentliche Projektarbeit, die Beiträge zum System, verloren. Sie wurden geschrieben, getestet, veröffentlicht, vergessen. Was hat gefehlt? Ein einziger engagierter Jemand. Jemand, der anspricht, sammelt, einfügt, ordnet, beschreibt. Eine zentrale, integrierende Instanz. Für das F-PC ist dies Tom Zimmer. Beiträge, die zu ihm gelangen und gewissen qualitativen Maßstäben genügen, finden Eingang ins System. Das bedeutet, im nächsten Release gehören sie zum LIEFERUMFANG. Dadurch entwickelt sich das System weiter und bleibt für die Benutzer und Projektteilnehmer sichtbar lebendig.

Das gleiche Ziel hat sich das System F68K gesetzt. Es deckt mit den 680x0- Prozessoren die zweite starke Prozessorlinie ab, die im Gegensatz zu den Intel- Maschinen in einer größeren Anzahl verschiedener, gängiger Rechner eingesetzt werden. Wie F-PC ist F68K public domain, vollständig, dokumentiert und besitzt mit dem Autor das besagte integrierende Element. Deshalb wurde vom Direktorium beschlossen, das F68K durch die Forth-Gesellschaft zu unterstützen. Damit kann das System ab sofort über das Forth-Büro ( aber auch weiterhin direkt beim Autor *Jörg Plewe*) bezogen werden. F68K ist für Mitglieder der FG außer einem geringen Bearbeitungsobulus kostenlos. Nichtmitglieder zahlen zusätzlich einen Betrag in Höhe des entsprechenden Jahresbeitrages der Forth-Gesellschaft. Das Direktorium arbeitet noch an einem Kriterienkatalog für vergleichbare Forth-Systeme, die sich um das FG-Förder-Siegel bemühen.

Anzeige

### Verkaufe:

deutschsprachiges integriertes Datenbank-Pseudo-Forth "Amber" incl. Buch "Amber im Überblick", Heise Verlag - 49,- DM  
 "LaFORTH" experimentelles Forth für den PC plus "Guide to LaFORTH" (Ting) 49,- DM  
 Jörg Staben, Hagelkreuz 23  
 W-4010 HILDEN

### Vorschau VD 8/3

- **F68K-das erste Jahr** von Jörg Plewe
- **LINKED ACTIONS und kein Ende** von Jörg Staben
- **Fourieranalyse in FORTH** von Klaus Kohl
- **PRINT SCREEN in F-PC** von Jörg Staben
- **Internationaler Zeichensatz in F-PC** von Claus Vogt
- **Forth RISC-Prozessor FRP 1600** von Arndt Klingelberg
- **Automation eines Sägewerks** von Ralf Neuthe



Angeschlagen von Jörg Staben

Hagelkreuz- 23, W-4010 Hilden, Tel.: 02103 - 240609

Unter WANTED sollen Ideen und Wünsche geäußert werden können. Als Dankeschön für einlaufende Lösungen oder Anregungen bekommen die Einsender die aktuelle Version des F-PC.

### **SYSINFO** - Hardwareauskunft über den PC

Bei vielen Programmen wüßte man sehr gerne, welche Schätze in Form von Hardware in den ach so kompatiblen PC's verborgen sind. Was da fehlt, ist ein SYSINFO wie bei Norton und in PCtools. Videokarte gibt's schon, wer nimmt sich auch einen der Bereiche: Prozessor/Coprozessor, Schnittstellen, RAM, CMOS, Laufwerke... Da könnte man auch mal sehen, wie FORTH so im Projekt aussieht.

### **Forth in der Klemme** - wo ist die Brechstange?

Viel häufiger, als die anderen denken - Fehlersituationen!

Wie haben Sie die Situation wieder in den Griff bekommen? Mit welchen Werkzeugen aus dem prall gefüllten Werkzeugkasten des F-PC (SEE, DEBU(G), DUMP, DIS TRACE(CODEBUG)) oder des TCOM (BUG, bxDEBUG)?

### **Interrupts** oder warum in Ihrem Rechner die Milch nicht anbrennt.

1a) FORTH macht ja bekanntlich die KLEINSTEN Programme, ist also ein ideales Werkzeug für die TSR - Programmierung. Da gab's mal so einen tollen Beitrag über hardware/software interrupts in der Zeitschrift PASCAL. Wer macht's in der VD, indem er uns zeigt, wie die residente Uhr links oben auf dem Schirm erscheint??

1b) Wer zeigt's vielleicht an dem 100.000sten residenten SCREEN-Saver und Dokumenten-Wächter mit der Maus?

### **KEYBOARD-Treiber**

Der DOS-KBD braucht 15 KB, als SYS-Treiber sogar erschütternde 36 KB. Der Treiber von Koehlmann C'T kommt mit ca. 2 KB aus, ebenfalls der JJA-NOKBD der TOOL. Wer macht's in FORTH (TCOM und/oder FORTH-COMP) genauso oder kleiner und schreibt einen kurzen Überblick über das KEYBOARD als Eingabegerät?

### **Geräte-Treiber**

TCOM ist lernfähig! Wer bringt ihm bei, auf die OPTION /device hin einen Treiber zu generieren?

### **WANTED**

Für F-PC (Editor) wird ein Buffer gesucht, um - ähnlich wie in VolksForth - einzelne Worte zu entlöschen (nach Ctrl-T) oder zu verschieben (copy>buffer buffer>line) und ein Wort Delete/Move-Stack in LIFO (statt FIFO)

(vorhanden: FIFO buffer, in der ak version LINE STACK verbessert, Funktionstasten vorbereitet)

Kontakt:

Arndt Klingelberg  
Straßburger Straße 12  
W-5110 Alsdorf  
Tel.: 0 24 04 / 6 16 48  
Fax: 0 24 04 / 6 30 39

### **WANTED**

Thomas Prinz sucht Aktivisten für eine Fachgruppe 8051 (und 68HC11)

Kontakt:

Thomas Prinz  
Tel. 06221 - 701.509 g  
Fax 06221 - 70 06 12 g  
Tel. 06271 - 28 30 p  
Adalbert-Stifter-Str. 1  
D-W 6930 Eberbach aN

vorhanden:

Europakarte mit CPU 8052-AH-Basic, ROM 8 kB (Basic intern), RAM 8 kB für Basic, EPROM 16 kB (on Board programmierbar), I/O 2\* 8255 (je 3\* 8bit Port), 1\* 8255 (für RTC), 1\* MAX232EPE (RS 232).

verfügbar:

e-FORTH (original-Fassung) und AH-BASIC als EPROM (27C128)

geplant:

eFORTH (ca. 8kB) oder tCOM (ca. 16 kB), Forth zusätzlich (!) zu BASIC nutzbar (wegen Kompatibilität), Programmieren des EPROM (unter Hilfe von BASIC ?), Berichte für die Vierte Dimension.



## Kein PRINZ FORTH !!

Leserbrief von Friederich Prinz,  
Homburgerstraße 335, 4130 Moers 1 Tel.: (02841) 58398

Liebe neue VD Redaktion!

Ich bin mir bewußt, daß es sehr frustrierend für Euch sein muß, die Kritik für 'alte Sünden' präsentiert zu bekommen. Deshalb bitte ich Euch sofort, die folgende Kritik nicht persönlich zu nehmen, sondern als konstruktive Kritik eines nach wie vor an der VD und an der FG interessierten Menschen anzusehen.

Zunächst war ich sehr überrascht, als ich Auszüge aus der VD bereits in 'meiner' Mailbox lesen konnte - zwei Tage bevor ich die Zeitschrift auf dem Tisch hatte. Ich will einfach annehmen, daß hier unsere Post einmal mehr verantwortlich ist. Ich kann aber nicht umhin zuzugeben, daß mich das ein wenig ärgert, umso mehr, als ich zu der gleichen Zeit einen ersten Brief eines Lesers der VD auf dem Tisch hatte, der mich annehmen läßt, daß die ersten Exemplare der Zeitschrift noch deutlich eher ausgeliefert wurden. Ein Anruf, den mein Anrufbeantworter aufgezeichnet hat und der sich ebenfalls auf Inhalte der VD bezog, die ich bis dahin selbst noch nicht hatte, hat mich dann endgültig 'sauer' gemacht. Ich mag halt einfach nicht als Letzter informiert sein... Vielleicht läßt es sich in Zukunft organisieren, daß die Verteilung der VD zeitlich ein wenig gleichmäßiger erfolgt...

Als ich die VD endlich selbst auf meinem Schreibtisch liegen hatte, hat mir bereits das Titelblatt den nächsten Adrenalinstoß verpaßt. Dort prangt in schönen, fetten Buchstaben "PRINZ FORTH". Was soll denn das ? (!!!) Es gibt kein FORTH-System das von mir stammt oder an dem ich irgendwelche Rechte besäße! Ich arbeite seit Jahren daran, die 'Vergötterung' bestimmter 'Gurus' aus den Köpfen meiner Mitmenschen heraus zu bekommen und zumindest den Mitgliedern unserer FORTH-Gruppe begreiflich zu machen, daß im Prinzip jeder Mensch mit seinem Computer alles das tun kann, was auch die Mannen um Bill

Gates tun. Wenn ich bisher mit der Redaktion der VD korrespondiert habe (...mit den 'alten' Redaktionen), dann habe ich immer im Namen und im Sinn der Moerser Gruppe geschrieben. Ich war -und bin es noch immer- bemüht, unsere Gruppe als Gruppe darzustellen und deutlich zu machen, daß wir die Gemeinsamkeit ebenso pflegen, wie das FORTH. Mitunter ist uns die Gemeinsamkeit der Gruppe sogar deutlich wichtiger, was z.B. dann sichtbar wird, wenn wir, bei schönem Wetter, einen Kurstag ausfallen lassen und uns lieber geschlossen in 'unseren' Biergarten am Rhein begeben.

Die 'Schlagzeile' auf der Titelzeile könnte den Eindruck erwecken, alle FORTH-Aktivitäten in Moers und Umgebung seien allein auf mich zurückzuführen. Dem ist absolut nicht so. Im Gegenteil - ich möchte diese Gelegenheit dazu benutzen Euch einmal ein wenig mehr über die 'Interns' der Forthgruppe zu berichten, bzw. über die für uns wirklich wichtigen Aktivitäten einiger Mitglieder.

Da wären zunächst Bernd Feuerer und Ulrich Prinz zu erwähnen. Die Beiden haben einige Zeit hauptamtlich in der Geschäftsführung des Moerser Arbeitslosenzentrums gearbeitet - und sind seither schon seit vielen Jahren

ehrenamtlich weiter dort tätig. Ihren guten Beziehungen zu dieser Einrichtung verdanken wir, die Forthgruppe, es, daß wir die Räume der Einrichtung seit Jahren kostenlos an allen uns liebenden Wochenenden nutzen können.

Michael Major 'macht' zur Zeit einen Kurs der ebenfalls Samstag stattfindet. Eine Einführung in MSDOS und den 'Umgang' mit dem PC weckt bei seinen 12 Kursteilnehmern Interesse und Verständnis für das Werkzeug Computer und bringt uns -der FORTHgruppe- vielleicht das eine oder andere neue Mitglied. Wenn nicht, dann werden zukünftige Kurse, die wir zur Zeit planen und gemeinsam mit dem Arbeitslosenzentrum initialisieren und organisieren wollen, vielleicht auch diesen Nebenerfolg bringen. Wir arbeiten daran. Michael steckt in die Vorbereitung und Durchführung seiner Kurse mindestens ebenso viel Zeit und Arbeit, wie ich das für unsere FORTH Kurse tue. Ohne seinen Einsatz wäre diese zusätzliche Arbeit gar nicht zu leisten.

Und was ist eine Gruppe ohne Mitglieder? Die 'Familie' Wissusek lebt in Heiden, einem kleinen Städtchen in Westfalen, circa 70 km von Moers entfernt. Martin Wissusek arbeitet als Grubensteiger (technische Aufsicht) in einem Steinkohlenbergwerk der RAG. Der anstrengende Schichtdienst, der teilweise erst am Samstagmorgen beendet ist, hält ihn nicht davon ab, sowohl an den Kurstagen nach Moers zu kommen, als auch an den Samstagen 'zwischen den Kur-



Zum Autor:  
**Friederich Prinz,**  
Homburgerstraße 335, 4130 Moers 1  
Tel.: 02841-58398

Geb. 1955, Realschule, Zimmermanns Lehre, Betonbaumeister, Bergmann, Bergfachschule, Bergfach-Techniker (staatl. gepr.), Fachhochschulreife. Arbeitgeber: RAG (Bergwerk Friedrich-Heinrich, Stabstelle) Nebenberuflich Lehrer an der Bergfachschule für EDV-Grundlagen und -Praxis.

Computerkontakt seit 10 Jahren: Z80-CP/M, PC/XT-DOS, PC386-DOS und OS/2. Sprachen: BASIC, Pascal, C, Clipper. "FORTH habe ich schon sehr früh kennen, aber erst dann schätzen gelernt, als der BlackBox Charakter anderer Systeme deutlich wurde. Forth ist heute für mich das ideale 'Vehikel' zum Transport von Wissen rund um den Computer"

Hobbies: Sportschießen, Biergärten, Spaziergänge unter Tage, die Moerser Forthgruppe, FORTH und EDV, LERNEN.

sen'. Martin ist eben ein echter FORTHer. Ähnliches gilt für Willi Wissusek, den auch Schnee und Eis oder Nebel kaum jemals davon abbringen, die insgesamt 140 km für 3 bis 4 Stunden FORTH im Auto zu verbringen. Detlef Malbrecht, ein Dr. Ing an der hiesigen Bergschule (Moers ist auch eine Bergbaustadt) ist seit mehr als einem Jahr Mitglied unserer Gruppe und hat uns die Alternative eröffnet, bei Bedarf auf die Räume und Geräte der Bergschule auszuweichen. Dort stünde uns ein vollständig eingerichteter Computer-Lehrraum zu Verfügung, den wir bis heute nicht genutzt haben, weil wir z.Z. noch mit unseren derzeitigen Ressourcen auskommen. Es tut aber gut zu wissen, daß wir diese Optionen haben.

Ich will Euch nicht alle Mitglieder aufzählen und deren jeweiligen Einsatz für die Gruppe schildern. Ich denke aber, daß jetzt deutlicher wird, daß eine funktionierende Gruppe eben nicht nur 'eines Mannes Sache allein' ist. Viele müssen etwas geben, um aus dem Ganzen etwas Brauchbares zu gewinnen.

Etwas ganz anderes habe ich nicht so recht verstanden. Ich habe der VD die Kurstexte zu Verfügung gestellt, die im Laufe mehrerer Jahre hier in Moers entstanden sind. Zu meiner großen Freude tauchen einige Kapitel daraus in der jüngsten VD auf. Auf Anfängerkurse in 'ihrer' Zeitschrift haben die Niederrheinischen FORTHer lange vergeblich gewartet. Besser spät als nie - aber warum druckt die VD die ersten beiden Kapitel und das Kapitel 10 ab? Sollten die Inhalte der dazwischen liegenden Kapitel für Anfänger irrelevant sein? Das kann ich mir kaum vorstellen. Wenn ein 'Nachschieben' der fehlenden Texte geplant ist, dann sollte man das auch den Lesern mitteilen! Daß auch innerhalb der Leserschaft der VD ein echter Bedarf an Anfängerkursen besteht, zeigt mir ein Brief den ich dazu erhalten habe:

Ein Mitglied der FG aus Pforzheim hatte den Eindruck, daß wir hier in Moers einen 19-tägigen Kurs quasi 'an einem Stück' abhalten würden. Dem ist natürlich nicht so. Das könnten wir gar nicht leisten. Unsere Beru-

fe ließen das nicht zu. Der Kurs der auf der Seite -34- vorgestellt wird, ist zudem bereits vor zwei Jahren hier angeboten und vorgetragen worden. Das war der Redaktion der VD bekannt. Ich denke, daß man das den Informationen hätte begeben müssen! Dann werde ich immer wieder nach den Kosten für unsere Kurse und 'unser' ZF System gefragt. Auch dazu möchte ich an dieser Stelle noch einmal klarstellen, daß wir weder Gebühren für unsere Kurse erheben, noch Kosten anderer Art den Kursteilnehmern auferbürden. ZF ist ein Forthsystem von Tom Zimmer und Public Domain! Wir haben dieses System 'eingedeutscht' und ein wenig an unsere 'Bedürfnisse' angepaßt. Das ändert aber nichts an der Tatsache, daß ausschließlich Tom Zimmer Rechte an seinem ZF besitzt. Mr. Zimmer hat ZF aber in die PD gegeben. Daran werden wir nicht rütteln. ZF, im Original ebenso wie in der von uns veränderten Version, kann Jedermann kostenlos von uns beziehen. Wir nehmen die ursprünglichen Grundgedanken der Forthgesellschaft (Verbreitung von FORTH) so ernst, daß wir sogar den Versand gerne übernehmen. Selbstverständlich sind unsere Kurstexte ebenfalls umsonst. Wer im-

mer diese Texte haben will kann sie bei uns anfordern!

Ich hoffe, daß die neue Redaktion in Zukunft Einiges besser machen wird. Rolf Kretzschmar hat mir telefonisch zugesagt, daß die VD zukünftig wesentlich mehr für die Anfänger tun will. Darauf warten wir! Wir wissen hier in Moers aber auch, daß die Redaktion der VD nur dann ordentliche Arbeit leisten kann, wenn sie Leser der Zeitschrift ihren Teil dazu tun. Wir sind gerne dazu bereit. Und wir möchten alle anderen Leser dazu aufrufen, sich ebenfalls bei der Redaktion zu melden. Schreibt Briefe, sendet Beiträge und/oder sagt ganz einfach etwas dazu, was ihr gerne lesen wollt. Helft mit dabei, daß unsere Zeitschrift möglichst vielen Lesern etwas bieten kann. Damit erhöht ihr die Attraktivität der VD ebenso wie die Attraktivität der Forthgesellschaft selbst! Letztlich hilft ihr damit FORTH!

In diesem Sinne verbleibe ich mit einem herzlichen Gruß vom linken Niederrhein

Euer  
Friederich Prinz

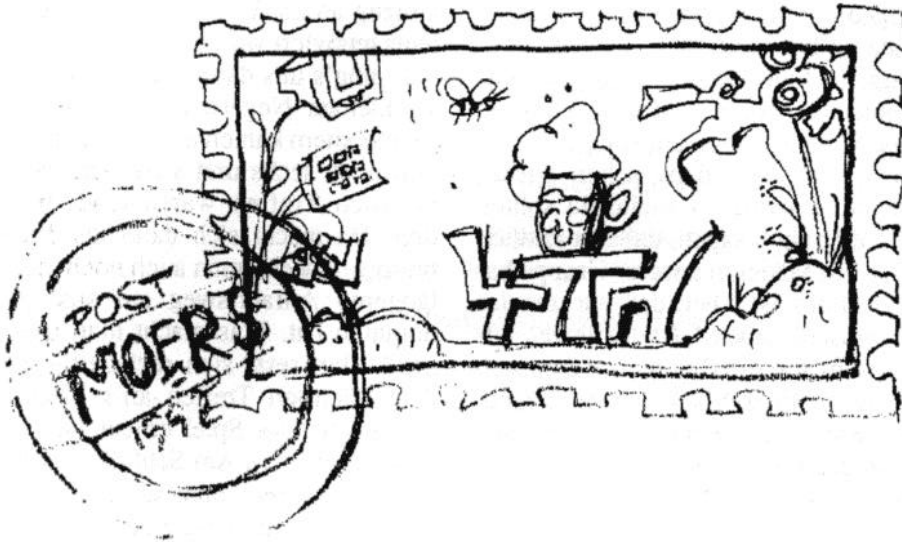
Lübbecke, 26.4.92

*Selbstgehester Herr Prinz!*

*Zum ersten Mal, daß ich über Einsteiger in FORTH lese und daß vor allem für sie etwas getan wird (VD-März 92). Für mich im äußersten Osten Westfalens ist es bis Moers leider zu weit. - Ich traue, was FORTH angeht, schon seit langer Zeit auf der Stelle. Eigentlich sollte ich alles weggeworfen haben! In weitaus den meisten Veröffentlichungen über FORTH wird für Kömmer geschrieben. In den wenigen FORTH-Büchern werden nur die einfachsten Operationen behandelt. Was fehlt, ist eine Darstellung in einem FORTH-Dialekt, die von den Grundlagen ausgeht und in einer zusammenhängenden Anwendung endet, so etwas wie ein Projekt. --- Ich wollte mich, mehr aus einer Laune heraus, mit FORTH anfreunden. Das ist von der Sache her schwierig, von meinem Alter her, -ich bin 55-, nicht mehr leicht und von der Unterstützung her unmöglich. Man hat den Eindruck, FORTH ler wollen sich alleinkalten. Die klagen aber über Desinteressel (...)*

*Mit freundlichen  
Grüßen E. Smetan*

Diese Karte von Herrn Smetan ist eine von mehreren Reaktionen auf den Prinz-Artikel in der vorigen VD. Zusätzlich erhielt Herr Prinz noch mehrere Anrufe.



## Brief aus der Provinz

Leserbrief der Forth-Gruppe Moers

Liebe (neue) Redaktion der VD,  
liebe Forthfreunde 'da draußen'....

Der Mai ist schon 'mitten drin', bis zum Sommer ist es nicht mehr weit und mit dem Sommer kommt auch für die Forthgruppe Moers die 'Sauere Gurken Zeit'. Zwar lassen wir uns, als echte Rheinländer, von den (hoffentlich) hitzeträgen Tagen und Wochen nicht negativ beeinflussen, aber eine 'wohlgefüllte' VD auf den Knien zu haben ist auch am Urlaubsort, am Badestrand und selbst in 'unserem' Biergarten nicht zu verschmähen. Zu einer wohlgefüllten VD wollen wir, die Forthgruppe Moers, ein wenig mit diesem Brief beitragen.

Laßt uns mit einigen Reaktionen auf die letzte VD beginnen. Friederich Prinz hat sich sehr über die Titelseite geärgert, weil die seiner Meinung nach irreführend und einfach unrichtig war. Das hat er der Redaktion aber schon deutlich geschrieben. Ebenso irreführend fand er die Seite 34 der letzten VD. Da stand das Inhaltsverzeichnis eines unserer älteren Kurse so abgedruckt, als würde unsere Gruppe diesen Kurs aktuell anbieten. Zunächst waren die meisten Mitglieder der Forthgruppe der Meinung, daß diese Seite gar nicht so mißverständlich sei. Aber spätestens als die erste

schriftliche Anfrage einging, in welcher ein eifriger FORTHer um die Beschaffung einer Übernachtungsmöglichkeit bat, weil er an allen Kurstagen dabei sein wollte, war schon klar, daß diese Seite schlicht 'geschlampt' worden ist. Friederich's Ärger ist verständlich. Schließlich mußte er insgesamt 10 (!) Interessenten an diesem Kurs mitteilen, daß die ganze Sache schon vor gut zwei Jahren hier angeboten und durchgeführt wurde. Wer mag schon gerne die Enttäuschung dieser Menschen auf sich nehmen?

Das scheint aber mittlerweile ausgestanden zu sein. Allerdings gehen noch immer vereinzelte Anfragen nach 'unserem' FORTH ein. Bisher hatten wir den Eindruck, daß sich außer unserer Gruppe Niemand für das ZF interessiert. Dem ist aber anscheinend gar nicht so. Das Interesse an ZF und an den in unserer Gruppe entstandenen Kurstexten und Übersetzungen steigt. Uns freut das, denn das zeigt, daß wir hier einen vernünftigen Mittelweg 'zwischen F83 und LMI' gefunden haben.

Zu den laufenden Kursen wollen wir noch einmal deutlich sagen, daß wir heute auf ein anderes 'Konzept' setzen. Der in der VD (leider nur teilweise) angefangene Kurs war gut, als wir noch 'pure' Anfänger in der Gruppe

hatten. Heute 'fahren' wir Kurssemester in denen wir jeweils ein spezielles Thema aufarbeiten. Das Frühjahr '92 hat uns dabei im Clinch mit den Eigenheiten des ZF als vollständiges Entwicklungssystem gesehen. Dieses Thema haben wir aber so gut wie abgehandelt. Zur Zeit diskutieren wir in der Gruppe deshalb das Thema des Herbstes. Im September, unmittelbar nach den Sommerferien, wollen wir uns mit den 'Innereien' des ZF-Systems im Besonderen und FORTH im Allgemeinen auseinander setzen. Wir wollen uns damit ein wenig darüber informieren, wie Forth-Worte von innen aussehen, bzw. wie sie strukturiert sind, wie das Dictionary aufgebaut ist, was bei einem Suchlauf im Dictionary passiert und wie VIEW die Angaben zum Quellfile einer Definition findet. Aber bis dahin ist noch etwas Zeit. Die warmen Monate kommen erst noch - und damit die Nachmittage und Abende im Biergarten (unser Lieblingsthema). Im Biergarten werden wir die Themen der letzten Kurse aufarbeiten und überlegen, ob wir als Gruppe den Moersern weitere Angebote machen können und wollen. (...)

Unser allemächstes Ziel ist jetzt erst einmal unser Sommerfest. Die Moerser Forthgruppe feiert einmal im Jahr ein kleines Fest - mit den Familienangehörigen! In diesem Jahr wollen wir bei einer kleinen Radtour (mit entsprechenden Pausen, selbstverständlich) (Friederich will nicht mit) ein wenig gemeinsames Tun erleben. Nach der Radtour geht's dann, wie im Vorjahr, in den Garten der Familie Purmann. Dort wird gegrillt, getrunken und geschwätzt - bis dem letzten die Augendeckel tonnenschwer werden.

Wir können nicht sagen, daß FORTH uns unwichtiger würde, aber mit jedem Jahr wird die Gemeinschaft uns wichtiger. (...) Wir sind halt keine 'Gurus' oder 'Köner' oder Spezialisten, sondern nur ein paar fortgeschrittene Anfänger die unter anderem auch Spaß an FORTH haben.

Daß FORTH für uns nicht Alles ist, zeigt ein 'neues Konzept' mit dem wir ebenfalls im Herbst beginnen wollen. Um den normalen 'Kursalltag' ein wenig aufzulockern, sollen und wol-

len die einzelnen Mitglieder der Gruppe, die ja alle gemeinsam versuchen zu FORTHen, Vorträge zu ganz unterschiedlichen Themen ausarbeiten. Als wir die möglichen Themen, die alle etwas mit dem Computer zu tun haben sollten, diskutierten, waren wir selbst überrascht, mit welcher unterschiedlichen Dingen die Einzelnen sich nebenher auseinandersetzen. Wir werden also neben FORTH ab dem Herbst Vorträge über Graphikstandards, über die IEEE Zahlenspezifikation, über die Speichersegmentierung in INTEL Maschinen, über Statistik, über Benchmarks zwischen FORTH versus PASCAL, C und BASIC, über Pascal selbst und über den Unter-

schied der Werteübergaben (by reference vs by value) und deren Bedeutung für FORTH hören. Die Vortragenden haben damit sicherlich jeweils eine Menge 'Arbeit am Hals'.

Naja, das wär's dann schon wieder aus der Provinz :-). Vielleicht sollten wir aber noch sagen, daß wir endlich auch ein Mädels in unserer Gruppe haben! Lang ist's her, daß wir uns das gewünscht haben! Jetzt drücken wir uns ganz fest die Daumen, daß diesem ersten, mutigen Mädels weitere folgen. In diesem Sinne wünscht die Moerser Forthgruppe dem Rest der forthenden Republik einen schönen Sommer.



Umzug an die Uni, weil das Jugendhaus renoviert wurde. Und so allmählich ging's uns dann ähnlich wie den 10 kleinen Negerlein: Von den 10 Teilnehmern kamen am nächsten Termin noch sechs und wiederum beim nächsten Treffen waren's nur noch drei. Da macht sich dann der Frust breit, v. a. wenn man auch noch einen längeren Anfahrtsweg auf sich genommen hat. Was macht man dagegen? Man setzt sich rechtzeitig vor dem nächsten Treffen an's Telefon und startet das Spiel erneut (s. Programmbeispiel). Am Schluß steht die Erkenntnis, daß man es eben läßt, wenn man den Hund zum Jagen tragen muß...

Seitdem gibt's also keine Treffen mehr, aber gelegentlich ruft mal jemand an, den ich dann mit ein paar VD-Heften versorge. Momentan habe ich aber keine Zeit, irgendwelche Treffen zu organisieren. Ich könnte mir jedoch vorstellen, daß genügend Leute mitmachen würden, wenn jemand wieder etwas auf die Beine stellen wollte. Zum Beispiel würde ein Einführungskurs in FORTH sicher gut ankommen. Das müßte ja nicht vorlesungsmäßig von statten gehen, sondern man könnte sich FORTH gemeinsam aneignen, indem man die Schwierigkeiten, die einzelne haben, gemeinsam löst. Nur ohne Computer klappt das natürlich nicht.



## Die Forth-Gruppe Stuttgart Sein - oder nicht ?

Leserbrief von Wolf-Helge Neumann, Huttenstraße 27, 7000 Stuttgart 31, Tel.: (0711) 88 26 38

"Wie sieht's denn so bei der Stuttgarter Gruppe aus", fragte mich neulich Rolf Kretschmar am Telefon. Am einfachsten ließe sich diese Frage mit der Gegenfrage "Welche Gruppe?!" beantworten. Warum das so ist, will ich jetzt mal erzählen:

Entstanden ist die Gruppe, weil ich bei meinen ersten Gehversuchen mit FORTH immer wieder mit Problemen konfrontiert war, die ich auf dem Atari bei GFA-Basic bestimmt nicht gehabt hätte. Fragen konnte ich keinen, weil ich niemand kannte, der sich außer mir mit FORTH beschäftigte, und so war das immer wieder etwas frustrierend, ein tolles Werkzeug zu besitzen, mit dem man angeblich "alles" machen kann, aber eben nur theoretisch. Weil ich dachte, daß es anderen sicher auch noch so geht, wollte ich unbedingt auch in Stuttgart Leute zusammenbringen, die sich mit FORTH beschäftigen. Nach einigen Inseraten in ein paar Kleinanzeigen-Zeitungen und nachdem ich von der FORTH-Gesellschaft die Mitgliedsadressen im Raum 7000 bekommen hatte, habe ich 58 Leute, die entweder bereits Mitglieder in der FORTH-Gesellschaft waren oder sich für FORTH interes-

sierten, zu einem ersten Treffen am 8.11.1989 eingeladen. Von diesen sind dann immerhin 16 in's Jugendhaus nach Heschlach gekommen. Getroffen haben wir uns dann einmal im Monat mit ca. 10 ständigen Teilnehmern. Ungünstig war natürlich, daß wir nie einen Computer zur Verfügung hatten, auch nicht nach unserem

```

: FORTH-Gruppe ( genügend Teilnehmer -- Spaß an FORTH)
  #Versuche 0 DO
    BEGIN
      Treffen
      #Teilnehmer @ Begeisterung @ *
      genügend? UNTIL
        telefonieren
        motivieren
        Termin_ausmachen
    LOOP
      Begräbnis ;
  
```



*Vielen Dank für den Bericht und den beigelegten Artikel. Daß ein Begräbnis Spaß an Forth zum Ergebnis hat, wollen wir hier unkommentiert lassen....! Stattdessen unser Apell: Liebe Stuttgarter! Laßt Euch nicht von Wolf-Helge Neumann zum Jagen tragen! Klingelt mal wieder bei ihm an! Wer einen Schlepp-Top hat, bringt ihn mit!* rk





## Prügel für das Schwein in der Kiste

Leserbrief zum Schweinekistenprogramm in 8/1 von Jörg Plewe, Mülheim an der Ruhr, Tel: 0208-423514

Ha, Ha! Das in der Schweinekiste im letzten Heft abgedruckte Programm muß natürlich so heißen:

```
crash_when_depth2
```

Das war leicht zu sehen, da die Rekursion über die Stacktiefe gesteuert wird, diese sich aber an keiner Stelle ändert. Also Rekursion bis zum Bersten des Returnstacks. Ich rief Uli Hoffmann an, um ihm zu diesem famosen Beitrag zu gratulieren. Er aber zerstörte mein Siegergefühl, indem er mich auf geringfügige Druckfehler hinwies.

Das Programm sah wohl einmal so aus:

```
: mystery ( ?? -- ?? )
depth 0=
IF
ELSE
  depth 1 =
  IF
  ELSE
    depth 2 =
    IF
    swap
    ELSE
      >r recurse r>
      swap
      >r >r recurse r>
      recurse r>
    THEN
  THEN
THEN
THEN
;
```

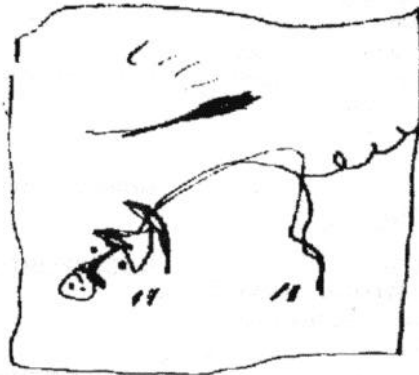
Daraufhin habe ich mich an den Tip gehalten, den Namen des Wortes zu raten und damit weiterzumachen. Bei bis zu drei Stackeinträgen erkennt man schnell durch gedankliches Programmverfolgen, wie der Stackkommentar aussieht:

```
( n1 -- n1 )
( n1 n2 -- n2 n1 )
( n1 n2 n3 -- n3 n2 n1 )
```

Der geratene Name muß also RE-

VERSE heißen. Um diesen Namen auch für größere Stacktiefen zu rechtefertigen, darf man AUF KEINEN FALL das Programm auch nur in die erste Rekursion verfolgen. Dann nämlich ist man sofort verloren. Einfach dem Stackkommentar glauben!!

Also: rekursiert wird erst, wenn die



Stacktiefe 2 ist. Man nehme einmal an, am Eingang der Wortes wären die Stackelemente der Größe nach sortiert. Das kleinste Element liegt unten, das größte oben. Dann wird zunächst das größte Element auf den Returnstack gesichert und der Rest des Stacks in seiner Reihenfolge umgekehrt (das Wort heißt schließlich REVERSE! Glaube dem Stackkommentar!) Das kleinste Element liegt jetzt oben. Nun wird das größte Element vom Returnstack zurückgeholt und mit dem kleinsten, das vorhin ja noch oben lag, vertauscht. Dann liegt jetzt das kleinste wieder oben und das größte darunter. Beide werden erneut auf den Returnstack gesichert und die Stackreihenfolge wiederum umgekehrt. Damit hat der Stack wieder seine ursprüngliche Reihenfolge, nur daß das größte und das kleinste Element nicht mehr dabei sind. Das erste gesicherte Element wird jetzt zurückgeholt -das war das große- und auf dem Datenstack abgelegt, wo es jetzt nach zweimaliger Umkehr der Reihenfolge auch wieder ganz gut hinpaßt. Durch erneutes REVERSE landet das größte Element jetzt also ganz unten im

Stack und das zweitkleinste steht ganz oben. Jetzt muß nur das kleinste Element, das vor der letzten Rekursion ebenfalls gesichert wurde, restauriert werden, und schon hat sich die Stackreihenfolge im Ganzen umgedreht.

Ich glaube, diese Erklärung ist einem Schweinekistenprogramm angemessen. Wahrscheinlich könnte sie auch ohne Änderung im zugehörigen Handbuch stehen (oder selbst in der Schweinekiste!).

Die Lösung des Problems läßt sich viel einfacher erklären, wenn man sich die Ausgangsposition klar macht: man hat ein Wort, das einen Stack mit maximal n Elementen umkehrt, und soll damit einen Stack mit n+1 Elementen umkehren.

Also kehrt man die n Elemente um, nimmt den kleinsten weg, kehrt die verbleibenden n-1 Elemente zurück und fügt das neue größte Element zu. Damit hat man wieder nur n Elemente. Dann wieder einmal 'rum (Größtes jetzt schon an seinem Zielplatz ganz unten) und das kleinste Element oben drauf. Bingo! Ich glaube, mit dieser Erklärung hätte jeder das Programm so geschrieben, oder??

□

### VORSCHAU:

Messen - Kongresse - Seminare

#### Meßcomp (Wiesbaden)

September 1992

#### Interkama (Düsseldorf)

5. - 10. Oktober 1992

#### EuroForth (Southampton)

23. - 25. Oktober 1992

#### Electronia (München)

10. - 14. November 1992



## Forth-Gruppen regional

<b>W-1000 Berlin</b>	Claus Vogt Tel. 0+30 - 2 16 89 38 p Treffen: nach Absprache
<b>W-4XXX Rhein-Ruhr</b>	Jörg Plewe Tel. 0+208 - 42 35 14 p Treffen: jeden 1. Samstag im Monat im S-Bahnhof Derendorf Münsterstr. 199, 4000 Düsseldorf
<b>W-4130 Moers</b>	Friedrich Prinz Tel. 0+2841 - 5 83 98 p Treffen: jeden Samstag 14:00 Arbeitslosenzentrum, Donaustr. 1 4130 Moers
<b>W-51XX Aachen</b>	Arndt Klingelberg Tel. 0+2404 - 6 16 48 agp Treffen: jeden 1. Montag im Monat als Gruppe des Computer-Club der RWTH, Seminargebäude Raum 214, Nähe Wüllnerstraße 5110 Aachen
<b>W-6800 Mannheim</b>	Thomas Prinz Tel. 0+6271 - 28 30 p Ewald Rieger Tel. 0+6239 - 86 32 p Treffen: jeden 1. Mittwoch im Mo- nat, Vereinslokal Segelverein Mannheim e.V., Flugplatz, 6800 Mannheim-Neustheim

## Forth-Fachgruppen

z. Zt. keine Gruppe

## Gruppengründungen, Kontakte

### Regional

**W-7000 Stuttgart** Wolf-Helge Neumann  
Tel. 0+711- 8 87 26 38 p

### Fachbezogen

**8051 ... (Forth statt Basic,e-FORTH)**  
Thomas Prinz  
Tel. 0+6271 - 28 30 p

## Forth-Hilfe für Ratsuchende:

### Forth allgemein

Jörg Plewe  
Tel. 0+208 - 42 35 14 p

Karl Schroer  
Tel. 0+2845 - 2 89 51 p

Jörg Staben  
Tel. 0+2103 - 24 06 09 p  
Di. und Fr. 20:00 - 22:00

Frank Stüss  
Tel. 0+6187 - 9 15 03 ap

## Spezielle Fachgebiete

**32FORTH (Atari)** Rainer Aumiller  
Tel. 0+89 - 6 70 83 55 gp

**FORTHchips (FRP1600, RTX, Novix ...)**  
Klaus Schleisiek-Kern  
Tel. 0+40 - 2 20 25 39 gp

**F-PC & tCOM, ASYST (Meßtechnik), embedded controller  
(H8/5xx//TDS2020, 8051 ... eFORTH...)**  
Arndt Klingelberg  
Tel. 0+2404 - 6 16 48 agp

**Gleitkomma-Arithmetik**  
Andreas Döring  
Tel. 0+721- 59 39 35 p

**HS/Forth (Harvard Softworks)**  
Wigand Gawenda  
Tel. 0+30- 44 69 41 p

**KI (Künstliche Intelligenz), OOF (Object Oriented Forth)**  
Ulrich Hoffmann  
Tel. 0+431 - 80 12 14 p

**Unterricht mit FORTH** Rolf Kretzschmar  
Tel./Fax 0+2401 - 8 88 91 ap

**UUCP (FORTH ... per eMAIL)**  
Andreas Jennen  
Tel. 0+30 - 3 96 52 27 ap

**volksFORTH/ultraFORTH**  
Klaus Kohl, Tel. 0+8233 - 3 05 24 p

## Forth-Forum: Forth-Mailbox

Jens Wilke (SysOp)  
Modem 0+89 - 8 71 45 48,  
300-2400 baud (8N1)

## Hinweise

a == Anrufbeantworter, hier können Sie Ihren Ansprechpartner  
eventuell vorinformieren, erwarten Sie bitte keinen (kostspieli-  
gen) Rückruf

g == geschäftlich, zu erreichen innerhalb typischer Arbeitszeiten  
p == privat, zu erreichen außerhalb typischer Arbeitszeiten

Für unsere Leser aus den neuen Bundesländern sind die Ortsnetz-  
kennzahlen auch nach 91jun01 nicht eindeutig. Bitte informieren Sie  
sich über örtliche Besonderheiten.

Es werden weitere Kontakte / Ansprechpartner gesucht, insbesonde-  
re aber auch im Osten. Bitte wenden Sie sich an die Redaktion.

## Redaktionsadresse

**Rolf Kretzschmar, Rote Gasse 7, W-5112 Baesweiler**  
Tel./Fax 0 24 01-8 88 91

## Forth-Büro

**Postfach 1110, W-8044 Unterschleißheim**  
Tel./Fax 089-3 17 37 84  
Mailbox 089-8 71 45 48 (300 - 2400 8N1)

Als international tätiger EDV-Hersteller mit extremen Zuwachsraten suchen wir einen

# Entwicklungsingenieur

## für hardwarenahe Softwareentwicklung

Sie sollten als Diplom-Ingenieur oder Physiker neben den fachlichen Fähigkeiten (Assembler, höhere Programmiersprachen, Betriebssysteme) über eine ausgeprägte Kooperationsbereitschaft, die selbständiges Arbeiten jedoch einschließt, und gute Englischkenntnisse verfügen.

Nun ist die Reihe an Ihnen. Wenn Sie bei uns starten wollen, sollten Sie uns Ihre Bewerbungsunterlagen zusenden. Wir möchten mehr über Sie erfahren und Ihnen mitteilen, was GFC für Sie in Zukunft bereit hält.



**mbH, Vogelsanger Weg 39, 4000 Düsseldorf 30**  
**Telefon (0211) 61895-0, Telefax (0211) 61895-74**

Hamburg · Bielefeld · Halle · Düsseldorf · Frankfurt · Stuttgart · München

# KK-FORTH

## hatein Herz für Mikrocontroller

- Verfügbar für:
- IBM-PC/XT/AT oder Kompatible
  - mc-PC-EMUF (V20)
  - mc-Z80-Mini-EMUF (84C015)
  - mc-RISC-EMUF oder FG-RTX-Board (RTX2000/1A)
- Benötigt:
- mindestens 16KByte EPROM; ab 8KByte RAM (optional: 32K/32K)
  - eine serielle Schnittstelle (PC-Terminalprogramm wird mitgeliefert)
- Features:
- über 400 Befehle (FORTH-83 mit einigen ANSI-Erweiterungen)
  - ROM-Fähig, für Interrupts vorbereitet
  - Komfortabler Zeileneditor, Fileinterface über PC-Terminalprogramm
  - Viele Kernroutinen umleitbar (z.B. Ein-/Ausgabe)
- Preise (inkl. MwSt):
- |           |  |
|-----------|--|
| DM 65.00  | Handbuch allein (für alle Versionen gleich)        |
| DM 70.00  | Eine Version (Disketten und Zusatzbeschreibung)    |
| DM 110.00 | Vollversion (Disketten und vollständiges Handbuch) |
- Zilog Super8-Chip mit ROM-FORTH
- |           |   |
|-----------|---|
| DM 45.60  | Super8-Chip mit FORTH im Masken-ROM                 |
| DM 34.20  | Beschreibung und Disketten zum S8-FORTH             |
| DM 51.30  | Bausatz zum S8-Projekt VD3/91 (Platine, GAL, SMD's) |
| DM 228.00 | Vollständig bestückte Platine mit Beschreibungen    |

**Ingenieurbüro Klaus Kohl · Pestalozzistraße 69 · W-8905 Mering 1**



# FORTH-SYSTEME GMBH

Postfach 1103,  
7814 Breisach

Telefon (0 76 67) 5 51  
Telefax (0 76 67) 5 55

Telefon Schweiz:  
**(055) 53 65 55**

## UR/FORTH

- FORTH-83 Standard
- Für MS-DOS, OS/2, 80386
- Direkt gefädelt Code Implementationen mit dem obersten Stackwert im Register um größtmögliche Ausführungsgeschwindigkeit zu erreichen
- Segmentiertes Speichermodell mit Programm, Daten, Headers und Dictionary Hash Table jeweils in einem getrennten Segment
- Komplett gehashtes Dictionary führt zu extrem schneller Übersetzung
- Mächtige neue String Operatoren (Suche, Extraktion, Vergleich und Addition) sowie einen dynamischen String-, Speichermanager
- Kann mit Objektmodulen, die in Assembler oder anderen Hochsprachen erzeugt wurden, gelinkt werden
- Native Code Optimizer zur direkten Umsetzung in 80 x 86 Code im Lieferumfang

## WinFORTH

- UR/FORTH kompatibel
- Windows Funktionen werden voll unterstützt
- Erweiterte Debugging-Hilfsmittel
- Online Windows Hilfe
- Coprozessor Unterstützung möglich
- Software-Gleitkomma-Paket
- Viele Beispielprogramme
- Upgrades von UR/FORTH Systemen auf WinFORTH sind preisgünstig zu erhalten

## SRS II

- Serieller ROM Emulator
- Unterstützung folgender Bausteine:  
27256, 27512, 271000, 27010, 27020, 27040
- Minimale Zugriffszeit 100 ns
- Maximale Baudrate 115.200 bits/s
- Highspeed Interface als Option
- Gleichzeitiger Zugriff von Host und Zielprozessor
- Zusätzliche serielle Schnittstelle über den ROM-Sockel
- Intel-Hex, Motorola-S oder ASCII/binär Formate werden unterstützt
- Der SRS II ist nur 157 x 94 x 36 mm groß
- SRS63 kompatibel

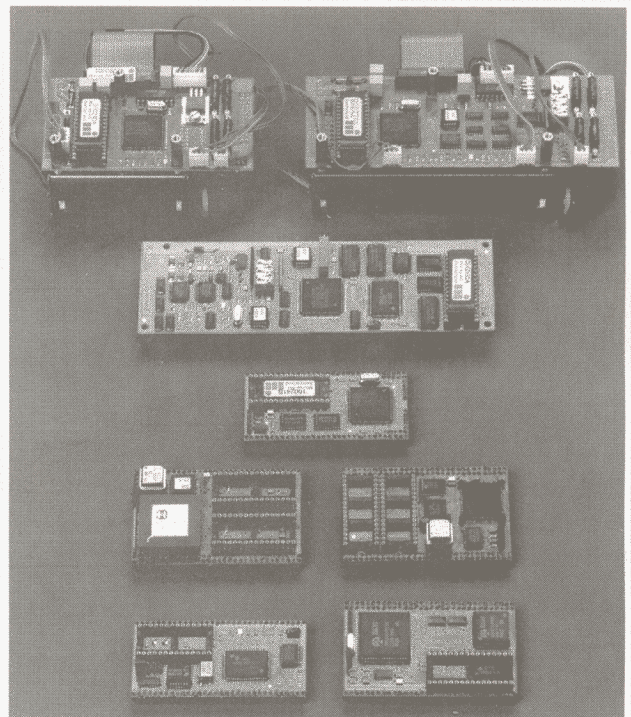
## LMI FORTH-83 Metacompiler

Der LMI FORTH Metacompiler wird mit komplettem Quellcode für ein ausführlich ausgetestetes, Hochgeschwindigkeits FORTH 83 Kern ausgeliefert, wobei Sie die Auswahl aus folgenden Zielprozessoren haben:

- |               |               |
|---------------|---------------|
| • 8086/8088   | • 78310       |
| • Z80/HD64180 | • 8031/32/535 |
| • 8080/8085   | • 6303        |
| • 68000       | • 6502        |
| • Z8          | • V25         |
| • 1802        | • 68HC11      |
| • 6809        | • RTX 2000    |
| • 8096/97     | • 80C166      |

Sie erzeugen schnelle und kompakte Anwendungen, indem Sie Ihre Quellprogramme mit unserem Forth Nucleus zusammenstellen und ihn mit dem LMI FORTH Metacompiler übersetzen.

## ModuNORM



CPU-Steck-Module im Scheckkartenformat:

- |                        |  |
|------------------------|--|
| • 8 Bit z.B. 6303      | • Softwareunterstützung durch SwissFORTH |
| • 16 Bit z.B. V25      | • Thermodrucker und Controller           |
| • Highspeed RTX-2000/1 | • LCD Grafik-Controller                  |
| • 80C166               |  |

Bitte fordern Sie unseren Produktkatalog und die Preisliste an. FORTH-Gesellschaftsmitglieder erhalten bis zu 10% Rabatt (artikelabhängig).