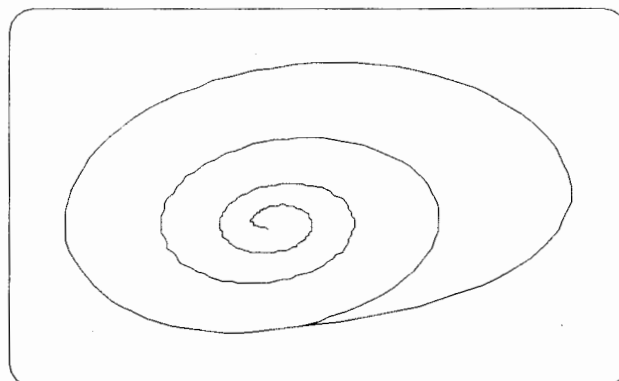


VIERTE DIMENSION

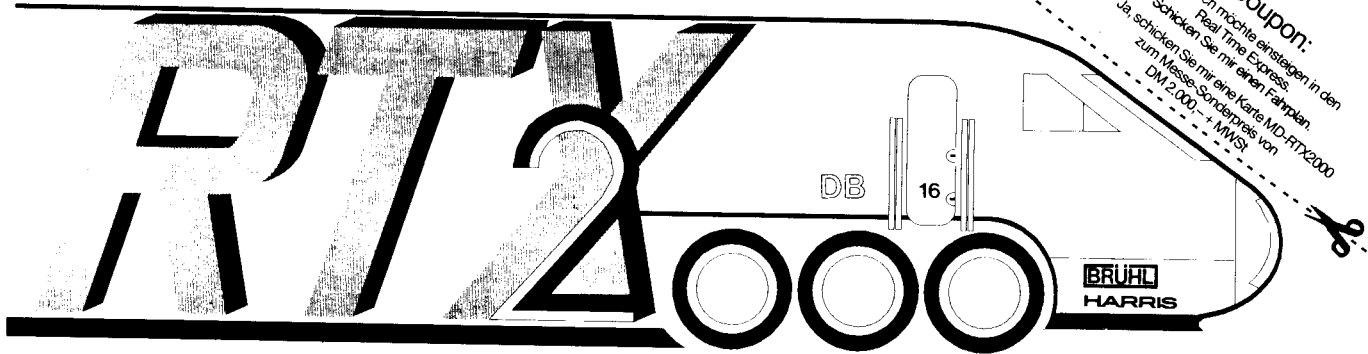
THEMEN

- Maskenerzeugung
- Rössler-Attraktor
- Interview mit John D. Hall
- FORTH-Windows
- ESY-Expertensystem

Einzelpreis DM 7,50



Rössler-Attraktor



Coupon:
 Ja, ich möchte einsteigen in den Real Time Express.
 Ja, schicken Sie mir einen Fahrplan.
 Schicken Sie mir eine Karte MD-RTX2000 zum Messe-Sonderpreis von DM 2.000,- + MWS

Die Zukunft hat schon begonnen

Warten Sie nicht auf Transkarbid oder Grande Tristesse!
 Steigen Sie jetzt ein in den Real Time Express.
 Nur mit dem Real Time Express gewinnen Sie Echt Zeit.
 Sie brauchen Ihr Fahrzeug nicht mehr mit Assembler zu tunen.
 Genießen Sie das Hochgefühl mit Hochsprache und Echt Zeit Ihr Ziel zu erreichen!



SONDERPREIS für Mitglieder der FORTH-Gesellschaft:
 DM 1.777,-
 (incl. Versand, incl. MWSt)
 Hochschulrabatt auf Anfrage!

REAL TIME EXPRESS und RTX ist TradeMark der HARRIS CORPORATION, Palm Bay, Florida

MINIBEE und QuadCard ist TradeMark der BRÜHL EE GmbH, Nürnberg



Karten zum Einsteigen erhalten Sie ab sofort bei:
 BRÜHL ELEKTRONIK ENTWICKLUNGS-GESELLSCHAFT mbH, Hegelstraße 10, 8500 Nürnberg 10, Tel. 0911/359088
 QuadCard 100x100 mit RTX2000, 128 KB RAM, 64 KB PROM,
 40 MHz Quarztakt: DM 2.000,- + MWS (electronica Sonderpreis incl. FG-FORTH-Compiler)

DELTA t

Die Firma mit dem
 FORTH - KNOW - HOW

*Seitdem es
 in Echtzeit geht
 wird der Zufall
 immer greifbarer*

Viele zeichnen Daten schnell auf.
 Wir verarbeiten bis zu einer Million
 Samples/Sekunde mit unserem
 Multiprozessorsystem auf der Basis
 von Forth-RISC-Prozessoren.

DELTA t Ulrich Hoffmann Marina Kern Klaus Schleisiek
 Entwicklungsgesellschaft für computergesteuerte Systeme mbH
 Telefon 040 / 644 57 82 · Roter Hahn 72 · D - 2000 Hamburg 72

EDITORIAL

Weihnachten steht vor der Tür und pünktlich zum Fest erscheint die zweite "Vierte Dimension" mit neuem Gesicht. Die anfänglichen organisatorischen Schwierigkeiten sind nun weitestgehend überwunden, so daß der vierteljährliche Erscheinungsrhythmus wieder gewährleistet ist.

Herzlichen Dank für alle Beiträge und Zuschriften. Es wäre schön, wenn sich noch mehr unserer Leser für eine aktive Beteiligung entscheiden könnten. Dadurch wird die "Vierte Dimension" noch interessanter und facettenreicher. Sie brauchen wirklich keine Angst zu haben, daß Ihr Beitrag unseren Ansprüchen nicht genügt. Wir freuen uns wirklich über jede Zuschrift!

In diesem Heft finden Sie u.a. einen Artikel über die "Erzeugung von Bildschirmmasken", die Auswertung unserer Umfrageaktion vom Herbst, ein Interview mit John D. Hall, einem der Direktoren der FORTH Interest Group, sowie vieles mehr.

Wir hoffen, daß das vorliegende Exemplar Ihnen ein paar informative und unterhaltsame Stunden bereitet und wollen auf diesem Wege allen Lesern ein frohes Weihnachtsfest und ein glückliches neues Jahr wünschen!

Rainer Aumiller

Denise Luda

IMPRESSUM

Titel:
FORTH MAGAZIN 'Vierte Dimension'
Zeitschrift der Mitglieder der FORTH-Gesellschaft e.V. © 1988

Herausgeber:
FORTH-Gesellschaft e.V.

Redaktion:
D. LUDA Software, Staudingerstr. 65, 8000 München 83, Tel. 089/670 83 55

Kontaktadresse:
Entweder direkt die Redaktion anrufen bzw. anschreiben oder das FORTH-Büro in München, Postfach 1110, 8044 Unterschleißheim kontaktieren.

Autoren dieser Ausgabe:
Johannes Reilhofer, Ulrike Schnitter, Claus Kühnel, Christoph Kringinger, Ekkehard Flögel, G. Göttle, R.W. Kobenter. Übersetzungen: D. Luda. Graphiken teilweise Rolf Kretzschmar.

Erscheinungsweise:
Vierteljährlich

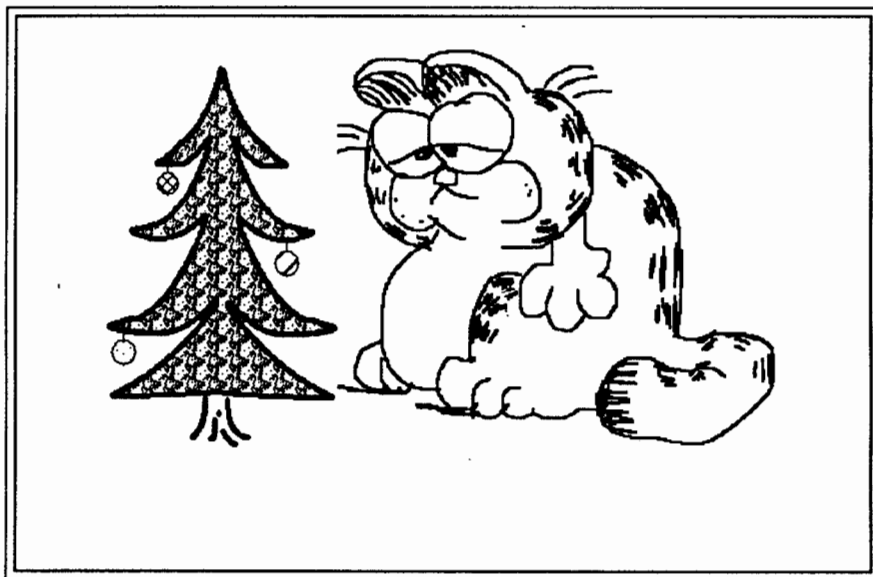
Redaktionsschluß:
Die zweite Woche im mittleren Quartalsmonat

Auflage:
ca. 1000 Stück

Druck:
Druckservice Ost GmbH, Pfanztelplatz 4, 8 München 83

Bezugspreis:
Einzelheft DM 7,50, Abonnement 4 Hefte DM 40,- inklusive Versand.

Für jedes eingesandte Manuskript sind wir sehr dankbar. Für die mit Namen oder Signatur des Verfassers gekennzeichneten Beiträge übernimmt die Redaktion lediglich die presserechtliche Verantwortung. Die in dieser Zeitschrift veröffentlichten Beiträge sind urheberrechtlich geschützt. Übersetzung, Vervielfältigung, Nachdruck sowie Speicherung auf beliebigen Medien ist allerdings auszugswise mit genauer Quellenangabe erlaubt. Freie Mitarbeit ist erwünscht. Die Beiträge müssen frei von Ansprüchen Dritter sein. Veröffentlichte Programme gehen, sofern nicht anders vermerkt, in die Public Domain über. Für Fehler im Text, in Schaltbildern, Aufbauskizzen usw., die zum Nichtfunktionieren oder evtl. Schadhafwerden von Bauelementen führen, kann keine Haftung übernommen werden. Sämtliche Veröffentlichungen erfolgen ohne Berücksichtigung eines eventuellen Patentschutzes, auch werden Warennamen ohne Gewährleistung einer freien Verwendung benutzt.



Schon wieder Weihnachten ohne Lasagne

Vierte Dimension

Inhalt

Erzeugung von Bildschirmmasken Dieser Artikel zeigt anhand eines kurzen Programmes, wie einfach Bildschirmmasken erzeugt werden können.	<i>von Johannes Reilhofer</i>Seite 13
ADMINISTRATION Allgemeine Hinweise für die Mitglieder der FORTH-Gesellschaft e.V.	<i>vom FORTH-Büro</i>Seite 18
FORTH-Profil(e): John D. Hall Ein Interview mit John D. Hall.	<i>von Mike Ham</i>Seite 21
FORTH-WINDOWS Dieser Artikel zeigt, wie man für die Schneider CPCs eine Fenstertechnik programmiert.	<i>von Claus Kühnel</i>Seite 27
Kopierservice Ein Service der FORTH-Gesellschaft e.V.Seite 32
Wie das Titelbild entstand Mit dem in dieser Abhandlung vorgestellten Programm kann der Rössler-Attraktor berechnet und dargestellt werden.	<i>von Christoph Krinninger</i>Seite 34
ESY - Expertensystem Ekkehard Flögel stellt die Wissensbasis für das Expertensystem ESY vor.	<i>von Ekkehard Flögel</i>Seite 41
MARC4-Mikrocontroller Beschreibung des FORTH-programmierbaren Mikrocontrollers MARC4 der Firma EUROSIL.	<i>von G.Göttle, R.W. Kobenter</i>Seite 43
Auswertung der Umfrageaktion Auswertung der im September vorgenommenen Umfrageaktion unter den Mitgliedern der FORTH-Gesellschaft e.V.	<i>von Christoph Krinninger</i>Seite 46

Inhalt

FORTH-Bibliothek, Teil 1

.....Seite 48

Diese Übersicht zeigt einen Ausschnitt aus der umfangreichen Bibliothek der Münchner Gruppe.

EDITORIAL, Impressum

.....Seite 3

Nachrichten

.....Seite 9

Anleitung für Autoren

.....Seite 8

Gruppen

.....Seite 50

Insertenverzeichnis

.....Seite 8

ZUSCHRIFTEN

.....Seite 6

Kleinanzeigen

.....Seite 49

LESERBRIEFE und Zuschriften

F83-FORTH für uP 80C96 bzw. 80C196

Als begeisterter Forth-Anwender möchte ich auf dem uP 80C96 bzw. 80C196 ein F83 installieren. Kennt jemand eine Implementation oder wer weiß weiter?

Eckert A.
Bachtelstr. 3
CH-8107 Buchs
Schweiz

Handbuch zu PC/FORTH 3.10 gesucht

Aus den USA bekam ich folgendes Forth-System: PC/FORTH 3.10 (83 Standard) © 84 Laboratory Microsystems Inc., welches wohl in Deutschland nicht vertrieben wird. Leider fehlt mir das Handbuch dazu. Hat jemand Unterlagen über dieses System?

Essen, den 5.8.88

Detlef Bunk
Dipl.-Psychologe
Norderneyweg 5
D-4300 Essen 1
Tel: (0201) 718533

Maschinensteuerung

Ich möchte mit Forth-Programmen Maschinen bzw. Schrittmotorgetriebene Geräte ansteuern. Ich erhoffe mir über die Mitgliedschaft

in der FORTH-Gesellschaft Anregungen und Erfahrungsaustausch auf diesem Gebiet.

Berlin, den 2.10.88

Robert Dorfner
Finsterwalder Str. 7
1000 Berlin 26

Abonnement Ja/Nein?

Vielen Dank für die Zusendung der in neuem Gewand erschienen FORTH-Zeitschrift 'Vierte Dimension'. Ich bin von der Aufmachung sehr begeistert, man sieht gleich, es sind Profis am Werk. Die alte 'VD' sah mehr nach Club-Nachrichten aus, was jedoch kein Tadel sein soll. Da ich selber FIG-Mitglied bin und die FORTH DIMENSIONS bekomme, sehe ich viel Ähnlichkeit in beiden. Dieses finde ich sehr positiv, auch der Anzeigenteil ist gut, vor allem sehr informativ. Ihrer Anfrage über ein Abonnement der 'Vierten Dimension' kann ich nur zustimmen. Es gibt bestimmt viele FORTH-Benutzer, die gerne eine FORTH-Zeitschrift hätten, jedoch nicht unbedingt bereit sind, der FORTH-Gesellschaft beizutreten.

Hilden, den 23.10.88

K.H. Peilert
Tucherweg 8
4010 Hilden

(Die 'Vierte Dimension' gibt es jetzt auch im Abonnement für DM 40,- inklusive Versand. Die Red.)

Prototypenentwicklung

Ich interessiere mich im Moment vor allem für die interaktive Prototypenentwicklung von Neuronalen Netzwerksimulatoren auf STs, Macs, ATs und PS/2. Da es schwierig ist, 'Korrektheitskriterien' für die Implementation derartiger komplexer Strukturen auszumachen, bin ich daran interessiert, in FORTH zu arbeiten, da ich hier ständige interaktive Kontrolle darüber habe, was tatsächlich im

RAM vorliegt. (Offenbar simuliert man am M.I.T. Zellular-Automaten ebenfalls in FORTH).

Wer kann weiterhelfen ?

Zürich, den 20.10.88

Ulrich H. Bauer
Hofstr.55
CH-8032 Zürich
Schweiz

Ansprechpartner für FORTH

Ich bin seit einer Woche bei Euch Mitglied und möchte mich für den Bereich 'Lüneburg' als Ansprechpartner zur Verfügung stellen.

Ich besitze einen PC mit MS-DOS und kann für diesen Computertyp als Berater tätig werden.

Das Fachwissen und die Beratung wird sich auf FORTH, GRAFIK und FLOATING beziehen, wobei ich bei den beiden letztgenannten Gruppen noch aufbauen möchte. Der Bereich kann auch darüber hinausgehen, wenn andere regionale Bereiche keinen Ansprechpartner haben.

Lüneburg, den 26.9.88

Peter Bierbach
Schomakerstr. 65
2120 Lüneburg
Tel: 04131/65031

Turbo-FORTH: Teil 2

Die in meinem Leserbrief (siehe Doppelheft 2-3/88) aufgeworfenen Fragen sind zu meiner Zufriedenheit erledigt. Ich stehe seit Juni 1988 in Briefkontakt mit Marc Petremann von JEDI und bin seither mit dessen Unterstützung mit der Eindeutschung von Turbo-FORTH beschäftigt. Marc Petremann schickt mir laufend sämtliche neueren Errungenschaften und Zusatzinformationen über Turbo-FORTH. Zu erwähnen

ist das Buch 'FORTH83-STANDARD' von Marc Petremann, Jean-Marie Premesnil und Michael Zupan, das im Verlag Editions Loisitech erschien und FF 140,- kostet. Es erklärt (recht gut) die F83-Worte des Systems von Laxen and Perry und berücksichtigt auch einen Großteil der in Turbo-FORTH aufgenommenen Zusatz-Worte. Das eigentliche Turbo-FORTH-Handbuch ist in Bearbeitung. Die ersten 20 Seiten des Entwurfs liegen mir zur Unterstützung meiner Übersetzungsarbeiten vor.

Inzwischen vertreibt JEDI mit Modul 5 eine Version ihres Systems, bei dem man in Bezug auf die Bildschirmmeldungen zwischen Französisch, Englisch und Deutsch wählen kann. Der deutsche Teil der Formulierungen stammt von mir.

ROOT.VOC habe ich JEDI ebenfalls in Übersetzung zugeschickt. Außerdem ein paar Kleinigkeiten, wie SOUNDS.FTH von Michael Zupan mit deutschen Erklärungen. Von den über 600 Turbo-FORTH-Worten in der Auto-Dokumentationsdatei FORTH.VOC habe ich bis auf 30 noch fehlende alle übersetzt.

München, den 17.10.88

Fred Behringer
Straßbergerstr. 9c/519
8000 München 40

FORTH PD-Pool

Herzlichen Glückwunsch zur neuesten VD! Das neue Layout gefällt mir außerordentlich. Auch die Vielfalt der Artikel gefällt mir gut, sehr gut. Übrigens sollte die VD immer die Dicke eines Sonderheftes haben, oder nicht?

Grund dieses Schreibens ist aber nicht nur meine (gute) Kritik an der VD. Vielmehr möchte ich Ihnen mitteilen, daß ich seit dem 20. Oktober - also seit gestern - einen FORTH PD-Pool unterhalte.

Anbei mein vorläufiges Info-schreiben.

FORTH PD-Pool

FORTH gehört sicher zu den Sprachen, die man zu einem eigenen Programmiersystem umbaut. Man baut dieses und jenes zusätzlich ein, der eine schwört auf SWOPPERATOREN, der nächste zieht eine deutsche Umgebung vor usw. Leider müssen die meisten allein basteln und können nur selten auf Altbewährtes zurückgreifen. Das hat ein Ende! Ich biete Ihnen jetzt einen Pool an, der so funktioniert:

- Haben Sie selbst ein neues PD-Programm, das in den Pool aufgenommen werden soll, erhalten Sie als Gegenleistung 1 Diskette aus dem Pool gratis (Bitte anrufen)! Sollte für Sie im Moment nichts dabei sein, erhalten Sie selbstverständlich einen Gutschein!
- Ansonsten zahlen Mitglieder der Forth-Gesellschaft eine Gebühr von 6.50 pro Disk und eine Versandkostenpauschale von DM 2.
- Nichtmitglieder zahlen DM 8.00 pro Disk. Die Versandkostenpauschale beträgt ebenfalls DM 2.

Der am 20. Oktober gegründete Pool enthält momentan folgende Disketten:

ST1 Das bekannte
ST2 volksFORTH 83 V3.8
ST3 der FORTH-Gesellschaft

Weitere Disketten sind in Vorbereitung!

Erkelenz, den 21.10.88

Sven Lütke Meyer
Köln Heerweg 11
5140 Erkelenz 1
Tel.: 02431/2344

FORTH-Gruppe im Raum Moers

Wir haben hier in Moers vor einigen Monaten eine lokale Forthgruppe gegründet, die es sich zum Ziel gesetzt hat, FORTH in den hiesigen Schulen, Betrieben

und Institutionen zu verbreiten, soweit dies den einzelnen Mitgliedern der Gruppe möglich ist. Wir wollen versuchen, Anwender anderer Sprachsysteme von FORTH zu überzeugen und uns darüber hinaus gegenseitig bei den verschiedensten Fragen zu FORTH, aber auch zu anderen Problemen der Informatik, zu helfen. Wir sind ein relativ 'loser' und 'bunter' Haufen, der sich ebenso aus Ingenieuren, Technikern und Dipl. Kaufleuten zusammensetzt, wie auch aus Bergleuten und Handwerkern.

Wir treffen uns im Rahmen der gesamten Gruppe regelmäßig an jedem zweiten und vierten Samstag eines Monats und im Rahmen unserer Anfängergruppe separat an jedem ersten und dritten Samstag eines Monats. In den Räumen des Moerser Arbeitslosen Zentrums, das von der Stadt getragen wird, stehen uns an den Samstagen ab 14.30 Uhr einige Räume zur Verfügung. Das Zentrum liegt nahe der Stadtmitte in der Donaustr.1. Wir laden auf diesem Wege alle interessierten FORTH'ler aus der Umgebung (Duisburg, Essen, Krefeld, etc.) ein, sich anzuschließen.

Für unsere Anfängergruppe benötigen wir eine FORTH-Version für den Commodore Amiga. Falls jemand FORTH bereits auf dieser Maschine implementiert hat, bitten wir um Nachricht und Hilfe.

Einige unserer 'Fortgeschrittenen' beschäftigen sich augenblicklich mit dem FORTH-Assembler. Hierzu vermissen wir eine wirklich gute, d.h. umfangreiche Dokumentation mit Beispielen usw.. Falls uns jemand zum FORTH-Assembler entsprechende Literatur empfehlen kann, würden wir uns über einen Anruf oder eine entsprechende schriftliche Nachricht sehr freuen.

Friedrich Prinz
Homburgerstr. 335
4130 Moers 1
Tel. 02841/58398

Zuschriften

Dr.Dobb's Toolbook of FORTH Vol.II gesucht

Im letzten Heft der VD wurde das 'Dr.Dobb's Toolbook of FORTH' vorgestellt. Mich interessiert da vor allem die FFT. Die von mir derzeit verwendete, fast ausschließlich

in Hochsprache definierte FFT läßt hinsichtlich der Laufzeit erwartungsgemäß einige Wünsche offen.

Ich wäre sehr dankbar, wenn mir jemand aus der FORTH-Gemeinde dieses Buch für kurze Zeit leihweise zum Schnuppern zur Verfügung stellen würde.

Dr. Claus Kühnel
Zschertnitzer Str. 52
DDR-8020 Dresden 1

Hinweise für Autoren

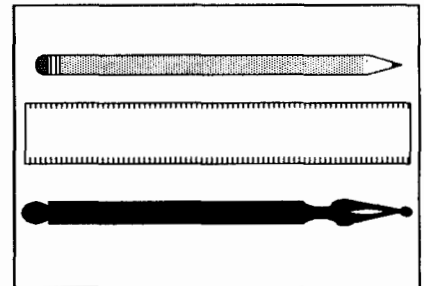
Auch in Zukunft möchten wir Beiträge veröffentlichen, die Sie uns hoffentlich in großer Zahl liefern werden. Schicken Sie Ihre Manuskripte bitte an die Redaktion der 'Vierten Dimension' D.LUDA Software Staudingerstr. 65, 8000 München 83, Tel. 089/6708355.

Am liebsten hätten wir die Manuskripte auf einer Diskette 5 1/4" (360 Kbyte oder 1,2 Mbyte) im IBM-Format oder einer 3 1/2" Diskette (Atari-Format oder 720 Kbyte IBM-Format). Ist Ihnen das nicht möglich, können Sie auch normale Texte auf Papier einse-

den. Bei Bildern sollte allerdings darauf geachtet werden, daß ein möglich guter Kontrast vorliegt. Die Arbeiten sollten in dieser Reihenfolge enthalten:

- Kurzer Titel,
- Autor,
- Zusammenfassung (ca. 50 Worte),
- Schlüsselworte (ca. 5), Text,
- Quellenangaben,
- Illustrationen,
- Tabellen,
- Quellcode.

Die Beiträge werden überarbeitet. Falls ein ausführliches Lektorieren erforderlich ist, erhalten Sie vor der Wiedergabe den Beitrag zur Korrektur und Zustimmung zurück. Layouts werden nicht mehr zur Prüfung durch die Autoren vorgelegt. Autoren erhalten auf Wunsch ein kostenloses Exemplar der 'Vierten Dimension' mit ihrem Artikel.



Inserentenverzeichnis:

Firma _____	Seite der Anzeige
DELTA † Entwicklungsgesellschaft für computergesteuerte Systeme mbH, Hamburg _____	2
RSO Gesellschaft für technische Kybernetik mbH, München _____	33
FORTH-Gesellschaft e.V. _____	49
Angelika Flesch , FORTH-Systeme, Breisach _____	52
Johannes Teich, Echtzeit-Applikationen Murnau _____	47
BRÜHL Elektronik Entwicklungsgesellschaft mbH Nürnberg _____	51,2
SIG - ACM, New York _____	17,40

Die Zukunft hat schon begonnen ...

bis zum Jahresende zum *electronica*-Sonderpreis von DM 2000,- + MwSt bei der Fa. BRÜHL in Nürnberg bestellen, noch ist Lieferung ab Lager möglich.

Für Mitglieder der FORTH-Gesellschaft gibt es einen ermäßigten Preis von DM 1.550,- zuzüglich Versandkosten und Mehrwertsteuer für eine Karte. Das FG-FORTH ist als Betriebssystem im Preis enthalten.

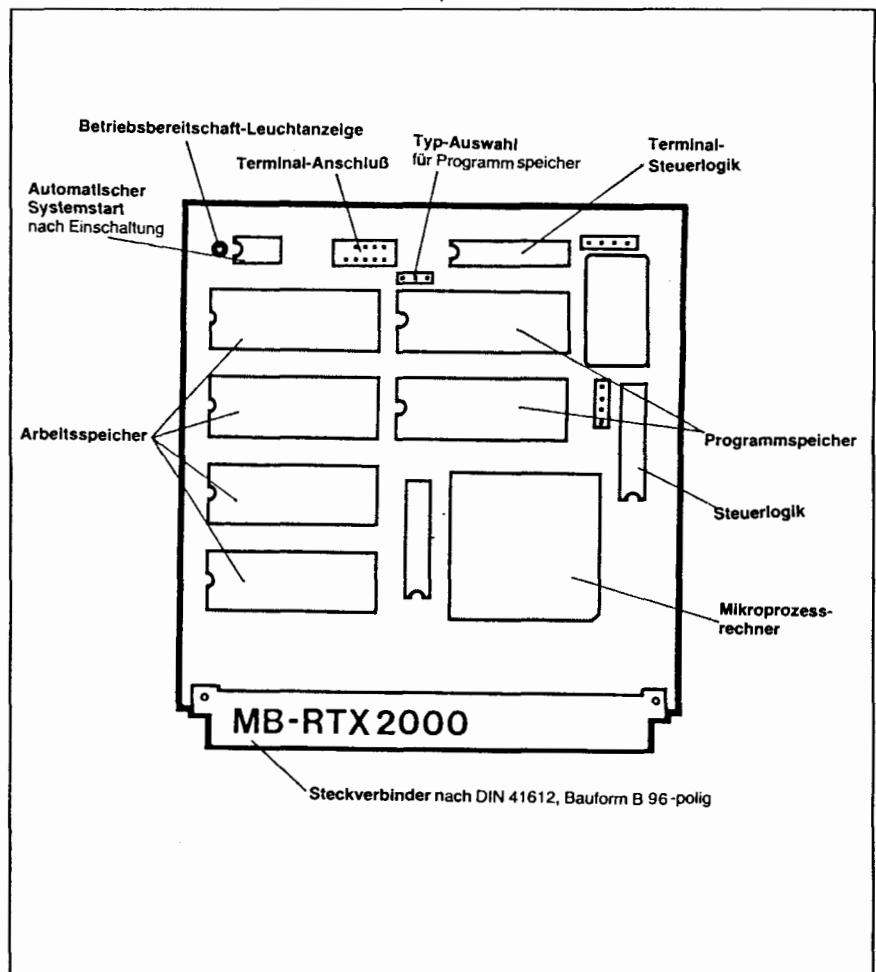
Nürnberg, den 08.11.1988. Die FORTH-RISC-Prozessoren für Echtzeit-Anwendungen rollen an. Den in "VIERTE DIMENSION" Heft Nr.2/3 - September 1988 - auf Seite 47 vorgestellten RISC-Echtzeitprozessor von HARRIS hat die BRÜHL EE GmbH in Nürnberg als erster deutscher Hersteller als Singleboard-Computer in ihr Vertriebsprogramm aufgenommen. Der auf der *electronica '88* vorgestellte Singleboard-Computer RTX 2000-MINI mit den Platinenmaßen 100mm x 100mm hat 128 Kbyte RAM und 64 Kbyte PROM; bei einem Prozessortakt von 6,67 MHz können als Spitzenleistung 10 Mio. Hochsprache-Operationen pro Sekunde erreicht werden. Damit sind erstmalig extrem kompakte Echtzeit-Steuerungen realisierbar, ohne Programme in Maschinensprache/Assembler erstellen zu müssen.

Die Entwicklung des Singleboard-Computers RTX 2000-MINI wurde von der deutschen FORTH-Gesellschaft angeregt, die auch die Lizenzrechte für das als Betriebssystem verwendete FG-FORTH vergibt. Programme für den RTX 2000 können mit jedem PC über die serielle Schnittstelle des MB-RTX 2000-Boards geladen werden; Baudraten bis 19200 Baud werden automatisch erkannt.

Da der Prozessor FORTH-Code direkt ausführt, ist - auch bei zeitkritischen Anwendungen - keine Assembler-/Maschinensprache mehr für die Programmerstellung notwendig. Damit ist der Programmierer endlich von diesem Ballast befreit.

Wer jetzt einsteigen will in den "REAL TIME EXPRESS", kann die komplett bestückte und geprüfte, funktionsfähige Singleboard-Computerkarte mit dem RTX 2000 noch

(RTX 2000 und REAL TIME EXPRESS sind Trademarks der HARRIS CORPORATION, Palm Bay, Florida)



Das RTX 2000-Board

FB-320

Interaktives Software Prototyping System für den DSP

Das interaktive Software-Prototyping-System für den DSP bringt bis zu 10 mal kürzere Entwicklungszeiten, weil man statt der bekannten Cross-Entwicklung direkt auf dem Digitalen Signalprozessor-Chip (DSP) interaktiv in Hochsprache programmiert!

Das FORTH-Entwicklungs-BOARD FB 320, das von der RSO GmbH, UB forthlab, angeboten wird, ist ein Einsteckboard mit dem TMS32020 (oder TMS320C25) Prozessor inklusive polyFORTH für den IBM-PC, XT, AT oder BIOS-Kompatible. polyFORTH ist ein voll multitasking-fähiges Betriebssystem plus Entwicklungsumgebung. Weil beim FB-320 direkt auf dem Chip High-Level-Forth compiliert und ausgeführt wird, können die Programme für den DSP voll interaktiv geschrieben und getestet werden. Im Vergleich zu der für den digitalen Signalprozessor sonst erforderlichen Cross-Entwicklung führt das interaktive Software-Prototyping zu beträchtlichen Zeiteinsparungen bei der Programmerstellung für diesen Prozessor - ganz gleich, ob man Forth vorher schon kannte oder nicht.

Schnelligkeit hat zwei Aspekte

Einmal ist es die Schnelligkeit der Programmierung. Beim FB-320 entwickeln Sie interaktiv: Sie testen Ihre Programmcodezeile sofort. Schreiben und Testen. Ändern und Testen. Kein Warten mehr auf das separate Assemblieren, Linken, Laden, Editieren. Kein Warten mehr,

daß endlich mal was läuft beim mühseligen Debuggen im Nachhinein.

Und Sie programmieren in einer Hochsprache: High-Level-Forth wird direkt auf den Signalprozessor-Chip compiliert und ausgeführt. Der zweite Gesichtspunkt ist die Schnelligkeit der Ausführung. Beim FB-320 erreicht man volle Chip-Geschwindigkeit. Nachdem Sie Ihre DSP-Applikation in Forth abgeschlossen haben, nehmen Sie sich die zeitkritischen Teile nochmal vor und schreiben sie mit Hilfe des

residenten Macro-Assemblers in Assemblersprache. Auch wenn Sie High-Level-Forth und Assembler beliebig miteinander verbinden, erreichen Sie volle Chip-Geschwindigkeit.

Bei Echtzeit kommt es auf das Betriebssystem an

Darüber hinaus gilt: die Schnelligkeit der Ausführung ist abhängig von der Verarbeitung durch das Betriebssystem. Nur wenn das Betriebssystem in Echtzeit arbeitet, kann auch die Ausführung in Echtzeit ablaufen. polyFORTH bewältigt das Interrupt-Handling ohne jeden Betriebssystem-Overhead und jedes I/O kann vom Programmierer selbst bestimmt werden. (der Programmierer hat volle Gewalt über alle I/O's.)

Auch beim FB-320 gibt es Echtzeit-Multitasking mit der Unterstützung einer beliebigen Anzahl asynchroner Prozesse, die konkurrierend ablaufen. Der verwendete Algorithmus maximiert die Bedienung aller Tasks. Unterstützt wird die TMS 320 Harvard-Architektur mit ihren

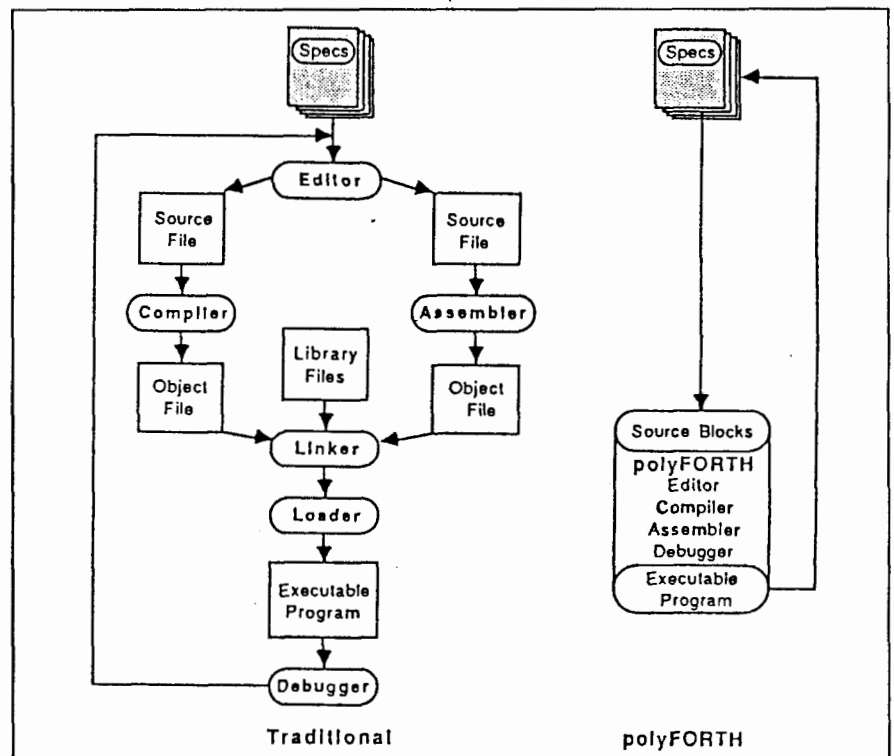


Bild 1: Vergleich der Entwicklungsumg.

Nachrichten

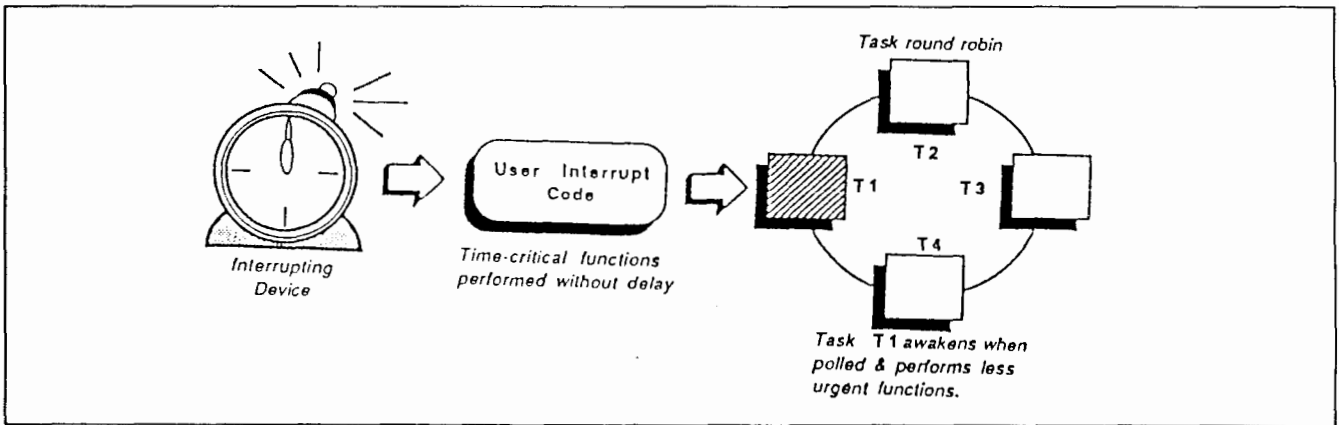


Bild 2: Das DSP-Board FB320

unterschiedlichen Adreßbereichen für Programm und Daten. Bis zu 64K 16-Bit-Worte sind direkt adressierbar.

Es besteht Kompatibilität zu MS-DOS und zu DOS-Files, mit Unterstützung von Disk- und Terminal-I/O durch MS-DOS. Das FB-320 Software-Paket wird als bootbares COM-File und in DOS-Files geliefert. Der Anwender kann seine Applikationen entweder in Source- oder Objektcode in DOS-Files ablegen. Der Transfer von Daten und Programmcode zwischen Host und FB-320 wird schnell und einfach über ein shared Memory Port durchgeführt.

Zum Beispiel: Entwicklerhilfen

Zur Einführung des Forth-Unterkundigen enthält das FB-320 Software-Paket zwei Beispiele unter

Verwendung der schnellen Fourier-Transformation (FFT, Teil der ebenfalls vorhandenen umfangreichen mathematischen Bibliothek) und des High-Speed Graphic-Displays.

Der polyFORTH Macro-Assembler für den TMS3200 bzw. TMS320C25 mit dem erweiterten Befehlssatz sieht dem Assembler von Texas Instruments sehr ähnlich, ist aber im Gegensatz zu ihm vollständig interaktiv.

Das System verwendet den Standard IBM-Graphik-Adapter und Farbmonitor (Option) und bietet damit eine extrem schnelle hochauflösende Graphik zur Unterstützung einer TMS320 Applikation. Auch die Graphik - mit Befehlen für das Zeichnen von Graphen in verschiedenen Formaten mit beschrifteten Achsen - ist interaktiv. D.h. hier: den Befehl für

das Zeichnen eines Kreises schreiben und ihn gleich ausführen und testen.

Weitere Bestandteile sind ein Full-Screen-Editor, eine Programm-Listing-Utility und die PROM-Programmierung. On-Line Dokumentationen und Debugging-Hilfen sind ebenfalls Merkmale für die umfassende Interaktivität der Programmierung.

Insgesamt: Ein integriertes Hard- und Softwarepaket

Das FB320-Entwicklungspaket wird mit dem vollständigen System-Sourcecode ausgeliefert; inklusive einem Target-Compiler, der in Sekundenschnelle das gesamte System für die gleiche oder eine andere Hardware-Konfiguration inklusive ROM-fähiger eingebetteter Applikationen neu compiliert. Zum Standardumfang gehören 1 Jahr Telefonhotline und umfangreiche Dokumentationen.

Die gelieferte Hardware ist ein DSP-Board mit dem TMS32020 oder C25 mit 24 KW SRAM (16 KW Daten, 8 KW Programm), mit der Aufrüstmöglichkeit auf 128K Worte RAM. Es hat alles "on BOARD", auch einen A/D- und D/A-Wandler.

Als Hostrechner muß ein IBM-PC, XT, AT oder vollständig BIOS-kompatibler Rechner mit mindestens 5 1/4" Floppy Drive und Harddisk vorhanden sein.



Das digit. Signalprozessorboard FB 320

FORTH-Mailbox

Allen an FORTH interessierten Computeranwendern steht jetzt eine Mailbox mit einem beträchtlichen Inhalt zur Verfügung. Eingerichtet wurde diese Mailbox (Protokoll 8N1, 300, 1200, 2400 Baud) von der Firma FORTH-Systeme Angelika Flesch. Sie ist unter der Rufnummer 07667/556 erreichbar. Die unten aufgeführte Abbildung zeigt einige Meldungen der Mailbox.

== [LMI Forth Board Bulletin Listings] =====

Additional bulletins for PC/FORTH 3.2, PC/FORTH+ 3.2, and UR/FORTH may be found in conferences 2, 3, and 4 respectively.

Date

Revised # Description

05-15-86 1 About the LMI Forth Board
05-15-86 2 Rules of this Board
05-14-87 3 PC/FORTH & PC/FORTH+ 3.1 (IBM PC) errata & bug notices
05-15-86 4 PC/FORTH (HP) errata and bug notices
08-10-86 5 PC/FORTH (Apple) errata and bug notices
05-15-86 6 8086 FORTH errata and bug notices
05-15-86 7 Z-80 FORTH errata and bug notices
05-15-86 8 8080 FORTH errata and bug notices
05-15-86 9 68000 FORTH errata and bug notices
.
06-08-86 21 Space Shuttle Children's Fund
11-04-86 22 LMI Forth Board Usage
06-22-86 23 PC/FORTH notes for version 3.0 and earlier
06-23-86 24 LMI Vendor Support conference on BIX
06-25-86 25 Programmable Eprom Simulator (PEPS)
07-08-86 26 New Forth/Expert Systems Book

===== [PCBoard Quick Reference] ===== [Bulletin # 19] =====

PROGRAM COMMANDS

Following are the program commands and stacked command combinations supported by the code. Please note that the following commands can be entered in either upper or lower case and can be separated by a semi-colon rather than a space indicated. CAUTION: DO NOT mix the space and semi-colon delineator on the same command line as erroneous program operation will occur.

User Menu Selections

A Abandon the current Conference

B Display Bulletin Listing
B # Display Bulletin number requested.
B # # Display Bulletin numbers requested.
B # # NS Display Bulletin numbers requested in non-stop mode.

C Leave a comment for the sysop (y/n).

D Download a file.
D name Download a specific filename

Erzeugung von Bildschirmmasken

von Johannes Reilhofer

Die Erzeugung von Bildschirmmasken kann je nach Art der verwendeten Sprachen mehr oder weniger aufwendig werden. Aber sicher ist sie am einfachsten, wenn man die Masken wie mit einem Malstift erzeugt. Wenn man nicht den Ort und die Zahl der Zeichen mühsam errechnen muß, sondern dem ästhetischen Gefühl folgend im Bildschirm herummalen kann.

Das Programm wurde so einfach gestaltet, daß jeder erfahrene FORTH-Programmierer es ohne weiteres auf die Hälfte reduzieren könnte. Das Programm soll aber vor allem den Neulingen auf dem FORTH-Sektor demonstrieren, wie einfach man sich im Speicherbereich eines IBM-PC's bewegen kann und wie man DOS-Files erzeugt.

Folgende Hardwarekonfiguration wurde verwandt:

FORTH: Ur/FORTH von LMI
Rechnertyp: PC, AT und Kompatible
Bildschirm: EGA (640 x 350), 16 Farben

Zur Bedienung:

Man lädt das LMI-FORTH, das Screen-File 'MASKE.SCR' und startet mit dem Wort MASKE. Als erstes wird der HELP-SCREEN auftauchen, aber nur bei denen, die ihn mit Hilfe dieses Programmes schon erzeugt haben.

Die anderen werden ein buntes Zufallsmuster sehen. Durch einen Druck auf die ENTER-Taste kommen Sie weiter.

Der nächste SCREEN listet alle Files mit der Extension 'msk'. Bei der ersten Installation finden Sie nur einen Eintrag mit 'leer.msk'. Geben Sie also 'leer' ein, malen Sie Ihr Bild und beenden Sie den Vorgang mit ESC. Die dann erscheinende 'Save-Maske' fordert Sie zur Eingabe eines Namens auf. Das Saven erfolgt im DOS-Format, läßt also Namen bis zur maximalen Länge von 8 Zeichen zu. Die Extension '.MSK' wird automatisch angehängt.

Mit jedem Neuaufruf von 'MASKE' können Sie die bereits geschaffenen Werke bewundern oder verändern.

Ihre anderen FORTH-Programme sind in der Lage, diese Masken auf den Bildschirm zu rufen und zu benutzen.

Beschreibung der Bausteine:

Im SCREEN 12 finden Sie das übergeordnete Wort 'MASKE'.

Die Steuerung soll vorzugsweise durch die Cursor- und die Funktionstasten erfolgen. Was passiert also, wenn man z. B. auf eine Funktions- oder Cursortaste drückt:

Man kann das leicht ausprobieren, indem man folgende Definition schreibt:

```
: TASTEN TEST  
BEGIN KEY KEY 2DUP  
.. DROP 27 = UNTIL ;
```

Die Steuertasten erzeugen bei jedem Druck erst eine 0 und dann ein ASCII-Zeichen, das ohne führende 0 ein Buchstabe geworden wäre. Dieser Unterschied zwischen Steuerbefehlen und Texteingaben wird in der 4. Zeile des SCREENs 12 gemacht.

Help-Screen 1 (Maske)

Help	F1	F2	I/O-Marke	Ende	ESC
Pen down	F3	F4	Pen up	des Maskenaufbaus	
Hilfszeile ON	F5	F6	Hilfszeile OFF		
Farbe	F7	F8	Symbol		
	F9	F10			

weiter mit <ENTER>

Name etc.

X:64 Y:22 FOREGR.: 1 BACKGR.: 7 ASCII: 82 TXT: R

Bild 1: Help-Screen

Erzeugung von Bildschirmmasken

Eine führende 0 lenkt zur Cursorsteuerung, zur Farbgebung etc. um. Ein Zeichen ungleich 0 geht in die 2. Hälfte der BEGIN-AGAIN-Schleife und wird durch den EMIT-Befehl auf den Bildschirm gebracht. Nur der Code '27' (ESC) führt zum Ausstieg aus dem Masken-Programm.

Das Programm sollte nun leicht änderbar, erweiterbar und vor allem verbesserbar sein. Dazu wurde jede Steuerroutine so gestaltet, daß sie auf dem Stack ein Zeichen erwartet. In jedem Fall dupliziert sie diesen Wert, war es "Thres" also das erwartete, wird sie ablaufen.

Die ganze Gruppe von Möglichkeiten sehen wir in dem Wort "MASKE" aufgeführt. Um die Übersicht zu behalten, wurde so gering als möglich verschachtelt. Sollen weitere Features in das Programm eingebaut werden, ist nur ein neues Wort in diese Gruppe zu integrieren.

Das Programm ist am eindrucksvollsten, wenn ein farbiger Bildschirm mit EGA-Karte zur Verfügung steht.

Mit der Funktionstaste F9 wird die Farbpalette auf den Schirm gebracht. Man hat nun die Auswahl zwischen 16 Vorder- und 16 Hintergrundfarben. Weiter ruft die Taste F10 die Graphiksymbole auf.

Das Ur/FORTH besitzt eine Voreinstellung, die das Einschalten von entweder 8 alphanumerischen Seiten oder 2 graphischen Seiten mit einer Auflösung von 640 x 350 zuläßt. Diese Eigenschaft ermöglicht das Springen zwischen der Farbpalette, den Graphiksymbolen und der neu entstehenden Maske.

Jede Maske hat die stolze Länge von 4 kByte. Dieser Wert kommt dadurch zustande, daß jeder der 2000 Plätze (80 Zeichen x 25 Zeilen) 2 Bytes benötigt. 8 Bit für den ASCII-Wert, 4 Bit für die Schriftfarbe und weitere 4 Bit für den Hintergrund. Der Bildschirm beim PC ist die Kopie aus einem Teil des Hauptspeichers. Zum Sichern einer Maske ist also nur dieser Speicherbereich auf Platte oder Floppy zu kopieren.

Das LMI-FORTH beginnt mit der 1. Seite bei der "langen Adresse": SEGMENT B800 (47104) OFFSET 0. Die 2. Seite besitzt das gleiche SEGMENT B800 (47104) der OFFSET ist jedoch Hex 1000 (4096).

Noch zu erwähnen ist die Definition 'CURSORINFO': Will man z. B. mit einem Rastermuster, das bereits in der aktuellen Maske verwendet wurde, weiter machen, dann fährt man nur mit dem Cursor über das interessierende Zeichen. Eine

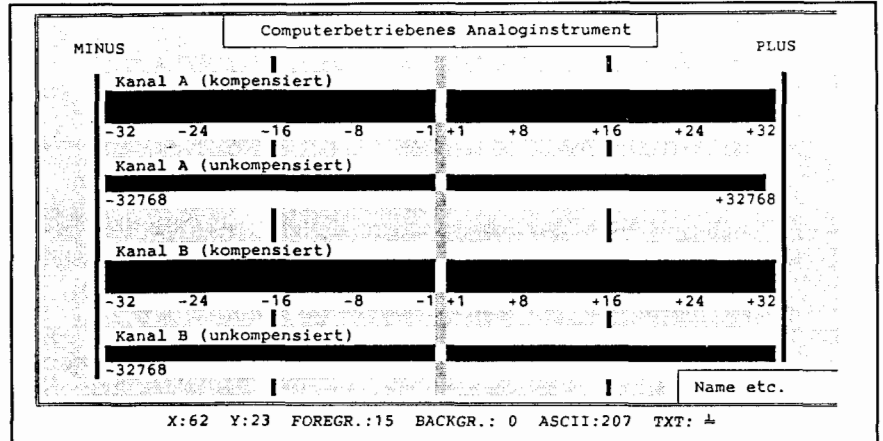


Bild 2: Beispiel-Maske

Die Aufteilung der Schirme ist:

- Seite 0 (0 VPAGE) neue Maske,
- Seite 1 Farben und Symbole,
- Seite 2 HELP-Maske.

Der Aufruf 'RETTEMASKE' kopiert Seite 0 in die 4000 Byte lange Variable 'VMASKE'. 'SAVEMASKE' fordert im Dialog den neuen Masken-Namen an und speichert 4000 Bytes DOS-konform ab. Die letzte Maske, die HELP-Maske, kann nach der jeweiligen Anordnung der Funktions- und Cursorarten gestaltet werden, so daß die HELP-Seite ein Abbild des Keyboards darstellt.

Hilfszeile am unteren Rand des Schirms gibt die XY-Position des Cursors, die Vorder- und Hintergrundfarbe, die Nummer des ASCII-Zeichens und sein Symbol an. Das erspart langes Suchen nach der gewünschten Farb- und Symbolkombination.

Wenn die Maske "sprechen" soll, sind die Orte zu ermitteln, an denen Fragen und Antworten plaziert werden sollen. Dazu dient wiederum die INFO-Zeile. Mit ihren Ortsinformationen ist es nicht mehr schwer, ARRAYS zu organisieren, die diese Masken im Dialog benützen.

Screen # 0

\ Maskenerzeugung

RHF 14:01 29.11.88

Das Programm erzeugt Masken wie sie bei Dialoganwendungen eingesetzt werden. Das "grafische Werkzeug" ist der Cursor.

- Man sieht was man bekommt -

Das Programm eignet sich für IBM PC's AT's und Kompatible mit EGA-Karte (640X350 bei 16 Farben).

Das verwendete FORTH ist das UR/FORTH von LMI

(Laboratory Microsystems, Inc.) Vertr. Fa. Flesch/Freiburg

Es läuft unter MS-DOS oder OS/2

Erzeugung von Bildschirmmasken

Screen # 1

\ Ladescreen 14:05 29.11.88

2 15 THRU

Screen # 4

\ Cursor RHF 12:40 29.11.88

```
: GRENZE? \ verhindert, da der Rahmen gesprengt wird
(xy - xy)
DUP 25 = IF DROP 24 THEN
DUP -1 = IF DROP 0 THEN SWAP
DUP 79 = IF DROP 78 THEN
DUP -1 = IF DROP 0 THEN SWAP ;

: CURSOR \ (n - n)
DUP 80 = IF ?XY 1 + GRENZE? GOTOXY ELSE
DUP 72 = IF ?XY 1 - GRENZE? GOTOXY ELSE
DUP 77 = IF ?XY SWAP 1 + SWAP GRENZE? GOTOXY ELSE
DUP 75 = IF ?XY SWAP 1 - SWAP GRENZE? GOTOXY ELSE
THEN THEN THEN THEN ;
```

Screen # 2

\ Farbe? RHF 15:00 29.11.88

```
: HI 640X350 VMODE CLS ; \ schaltet zum Grafik-Mode
: LO 80X25 VMODE CLS ; \ schaltet um auf alpha-numerisch
: FA 2 FOREGROUND 0 BACKGROUND ; \ grün auf schwarz
: FA1 14 FOREGROUND 1 BACKGROUND ; \ gelb auf blau
: FA2 14 FOREGROUND 4 BACKGROUND ; \ gelb auf rot
```

```
2VARIABLE FORBACK
: FARBE? FA CLS 0 0 GOTOXY
CR 16 0 DO 16 0 DO I FOREGROUND J BACKGROUND I.
LOOP FA." "I. CR LOOP CR CR ." Eingabe (Schrift) : "
#IN CR." (SCHIRM) : "
#IN 2DUP 2DUP
+ 0 = IF 2DROP 2DROP FORBACK 2@ ELSE
FORBACK 2! THEN BACKGROUND FOREGROUND ;
```

Screen # 3

\ Grafiksymbol? RHF 15:02 29.11.88

```
VARIABLE GRAFIKSYMBOL
: GRAFIKSYMBOL? 0
7 BACKGROUND 2 FOREGROUND CLS CR
." " REVERSE
." Auswahl des Symbols und der Farben "-REVERSE
CR CR 225 169 DO I FA 6.R." " FA1
I EMIT 0 0 FOREGROUND BACKGROUND ." "
1 + DUP 8 MOD 0 = IF DROP 0 CR THEN LOOP
FA1 CR 2 SPACES ." Eingabe: "
#IN DUP 0 = IF DROP ELSE GRAFIKSYMBOL ! THEN DROP ;
```

Screen # 5

\ CURSORINFO RHF 16:11 29.11.88

```
47104 CONSTANT LINKS_OBEN VARIABLE FLAGINFOZEILE
2VARIABLE XY

: INFOZEILEON (b - b) DUP 63 = IF 1 FLAGINFOZEILE ! THEN ;
: INFOZEILEOFF (b - b) DUP 64 = IF 0 FLAGINFOZEILE ! THEN ;

: CURSORINFO (b -- b)
LINKS_OBEN ?XY 2DUP XY 2! 160 * SWAP 2 * + @L 256 /MOD
16 /MOD FLAGINFOZEILE @ 1 = IF
12 24 GOTOXY XY 2@ SWAP ." X:" 2.R." Y:" 2.R
SWAP ." FOREGR.:" 2.R ." BACKGR.:" 2.R DUP
." ASCII:" 3.R." TXT: " EMIT XY 2@ GOTOXY
ELSE 2DROP DROP THEN ;
```

Screen # 6

\ InputFarbSymbol RHF 16:09 29.11.88

```
: NEUESGRAFIKSYMBOL? (b - b)
DUP 68 = IF ?XY XY 2!
1 VPAGE GRAFIKSYMBOL? 0 VPAGE
XY 2@ GOTOXY THEN ;

: NEUEFARBE? (b - b)
DUP 67 = IF ?XY XY 2!
1 VPAGE FARBE? 0 VPAGE
XY 2@ GOTOXY THEN ;
```

Erzeugung von Bildschirmmasken

Screen # 7

\ PenupdownOutpt RHF 16:08 29.11.88

```
VARIABLE PENFLAG
: PENUP ( b -- b )
DUP 62 = IF 0 PENFLAG ! THEN ;

: PENDOWN ( b -- b )
DUP 61 = IF 1 PENFLAG ! THEN ;

: SYMBOLAUSGABE ( b -- ) PENFLAG @
1 = IF GRAFIKSYMBOL @ FORBACK 2@
BACKGROUND FOREGROUND EMIT

?XY SWAP 1 - SWAP GOTOXY
THEN ;
```

Screen # 8

\ Helpmaske RHF 15:21 29.11.88

```
VARIABLE HELPSEITE 4000 ALLOT

: BRINGHELP 2 VPAGE
4000 0 DO HELPSEITE 1 + @ LINKS_OBEN 1 8192 +
!L 2 + LOOP 76 22 GOTOXY KEY DRÖP 0 VPAGE ;

: HELP DUP 59 = IF
" HELP.MSK" COUNT STRPCK 0
open DUP HELPSEITE 4000 read DROP close DROP

BRINGHELP THEN ;
```

Screen # 9

\ Screensaver RHF 16:48 29.11.88

```
VARIABLE VMASKE 4000 ALLOT

: RETTEMASKE \ Bildschirm - VMASKE
4000 0 DO LINKS_OBEN 1 @L VMASKE 1 + ! 2 + LOOP ;

: BRINGMASKE \ VMASKE - Bildschirm
4000 0 DO VMASKE 1 + @ LINKS_OBEN 1 !L 2 + LOOP
76 22 GOTOXY ;

: LEER_MASKE
CLS RETTEMASKE " LEER.MSK" COUNT STRPCK
0 creat DUP .S VMASKE 4000 write . close . ;
```

Screen # 10

\ Scr's - Disk RHF 15:21 29.11.88

```
: SAVEMASKE CLS 31 1 GOTOXY 14 FOREGROUND 9
BACKGROUND
." Saven ??? " FA

CR DIR *.MSK CR FA2
." - mit neuem Namen neue Maske kreieren mit altem übers
chreiben - "
CR CR ." - mit (ESC) abbrechen - "

CR CR CR CR FA

." Filename ( Name ohne Ext.) : " PAD 20 EXPECT

PAD SPAN @ ".MSK" COUNT STRCAT STRPCK
0 creat DUP .S VMASKE 4000 write . close . ;
```

Screen # 11

\ Load Maske RHF 15:23 29.11.88

```
VARIABLE AUSSTEIGFLAG
: LADEZUGRIFF PAD SPAN @ ".MSK" COUNT STRCAT STRPCK
0 open DUP VMASKE 4000 read DROP close DROP BRINGMASKE ;

: LOADMASKE CLS CR DIR *.MSK CR FA2 0 AUSSTEIGFLAG !
PAD 10 ERASE 0 SPAN !
." - !!!leere oder bereits bearbeitete Maske auswäh
len - " CR CR
." - mit (ESC) abbrechen " CR CR CR FA
." Filename ( Name ohne Ext.) : "
8 0 DO KEY DUP DUP 27 = IF 1 AUSSTEIGFLAG ! 2DROP LEAVE
THEN
DUP 13 = IF PAD 1 + C! DROP LEAVE THEN
1 SPAN + ! .EMIT PAD 1 + C! LOOP
AUSSTEIGFLAG @ 0 = IF LADEZUGRIFF THEN ;
```

Screen # 12

\ Maske 16:49 29.11.88

```
: MASKE LEER_MASKE
FA 59 HELP DRÖP 0 PENFLAG ! LOADMASKE
1 FLAGINFOZEILE !
BEGIN KEY DUP 0 = IF DROP KEY

CURSOR NEUESGRAFIKSYMBOL? NEUEFARBE?
PENUP PENDOWN SYMBOLAUSGABE HELP
CURSORINFO INFOZEILEON INFOZEILEOFF
DROP ELSE
DUP 27 = IF RETTEMASKE DROP FA SAVEMASKE EXIT

ELSE FORBACK 2@ BACKGROUND FOREGROUND EMIT

THEN THEN AGAIN ;
```


SIG **CALL** **FOR PAPERS** **FORTH '89**



for the first annual

FORTH APPLICATIONS SYMPOSIUM on **REAL TIME SOFTWARE ENGINEERING**

FOUR SEASONS HOTEL • Austin, TX • Feb. 17-19, 1989

The objective of this symposium is to share, discuss and disseminate recent research on and production of real time (software and hardware) computer applications. Attendees will hear presentations from industry experts on many topics, including:

Interrupt Driven Systems	Specialized Architectures
Programming Environments	Microcontroller Applications
Multitasking / Multiuser Systems	Industrial Systems
Parallel Processing	Computer Networks
Fault Tolerant Systems	Biomedical Engineering
Forth Engines and Software	Robotics & Machine Intelligence

Papers for oral and poster presentations are requested from computer professionals and other interested parties. Facilities will be available for scientific and technical demonstrations. Pre-publication proceedings will be made available to the participants at the symposium. Vendors of software and/or hardware may request exhibit space. Authors should submit an abstract of 250 words or less, typed, double spaced by the deadline below. Contributed papers should be previously unpublished work. You are not required to present a paper to attend the symposium.

Please send abstracts and requests for symposium information to:

Dr. Paul Frenger or Mr. Rick Hoselton
Technology Information Center, Inc.
2900 Wilcrest Drive #400
Houston, TX 77042
Telephone: (713) 952-1060

TIMETABLE:

Receipt of Abstract	Jan. 1, 1989
Notification of Acceptance	Jan. 15, 1989
Receipt of Final Manuscript	Feb. 10, 1989

Sponsored by the ACM Special Interest Group on Forth

For ACM SIGForth membership information, refer to the above address.

ADMINISTRATION

Seit Oktober 1988 wird die SFORTH-Gesellschaft e.V. vom FORTH-Büro Unterschleißheim verwaltet. Wir möchten heute einige Fragen, die häufig gestellt werden, beantworten.

Mitgliedsbeiträge und Zahlungsmodus:

Der Mitgliedsbeitrag wird immer zu Beginn eines Jahres (bis Ende Januar) fällig. Als Erinnerung liegt dieser VIERTEN DIMENSION ein Überweisungsformular für 1989 bei. Viele Mitglieder vergessen ganz einfach die Entrichtung des Jahresbeitrages, deshalb haben wir beim Postgiroamt Hamburg die Teilnahme am Lastschriftinzugsverfahren beantragt. Ab 1989 ist es also möglich, den Mitgliedsbeitrag durch eine Einzugsermächtigung, die Sie uns schicken, zu entrichten. Nutzen Sie bitte dieses Angebot. Die Einzugsermächtigung kann jederzeit schriftlich widerrufen werden. Der Mitgliedsbeitrag trägt immer für das laufende Kalenderjahr. In diesem Beitrag ist der kostenlose Bezug der VIERTEN DIMENSION enthalten. Neue Mitglieder erhalten die bereits herausgegebenen Ausgaben der VIERTEN DIMENSION des laufenden Kalenderjahres zugesandt.

Anfragen nach früheren Ausgaben der VIERTEN DIMENSION:

In Unterschleißheim haben wir noch folgende Ausgaben der VIERTEN DIMENSION vorrätig:

- Ausgabe 1985 Volume I Nummer 2 à DM 5,00
- Ausgabe 1986 Volume II Nummer 2, 3 und 4 à DM 5,00
- Ausgabe 1987 Volume III Nummer 1, 2 und 3 à DM 5,00
- Ausgabe 1988 Volume IV Nummer 1 à DM 5,00, die Nummer 2/3 das Doppelheft zu einem Preis von DM 10,00 und diese Nummer 4 zum neuen Preis von DM 7,50

Jedes Exemplar kann zu seinem Preis plus Porto bei uns angefordert werden. Andere VD's gibt es beim Kopierservice: Klaus Schleisiek, Roter Hahn 42, 2000 Hamburg 72.

Spendenbescheinigungen:

Bei Spenden bis DM 100,- genügt zur Geltendmachung der Zahlungsbeleg eines Kreditinstituts oder der Post als Spendennachweis. Wir stellen aber bei Zuwendungen ab 30,- DM jeweils zum Jahresende Spendenbescheinigungen aus.

Lokale Gruppen und Fachgruppen:

Bei der Neuanlage der Mitgliederliste haben wir festgestellt, daß offensichtlich mehr aktive Gruppen existieren als in der VIERTEN DIMENSION bekanntgegeben worden sind. Wir bitten deshalb alle Koordinatoren Ihre Aktivitäten, Treffpunkte und evtl. Telefonnummern in der VIERTEN DIMENSION zu veröffentlichen.

Mailbox:

Ab 1. Januar 1989 wird unsere Mailbox in Hamburg wieder betriebsbereit sein. Sie ist dann erreichbar unter der Rufnummer 040-644 46 30 mit 300 Baud, 8 Bit und no Parity.

Mitgliederliste:

Mit dem ersten Heft 1989 unseres FORTH-Magazins VIERTEN DIMENSION erhalten alle Mitglieder die neue Mitgliederliste zugesandt.

Leserbriefe und Kleinanzeigen:

Zuschriften für unsere Leserbriefecke fördern als offene Diskussionsbeiträge die Kommunikation der Mitglieder in der FORTH-Gesellschaft, deshalb freuen wir uns über jede Zuschrift. In der VIERTEN DIMENSION können ab sofort Mitglieder kostenlos Kleinanzeigen aufgeben. Für Nichtmitglieder berechnen wir eine Mindestgebühr von DM 5,00 für die ersten 5 Zeilen, jede weitere Zeile DM 1,00 (am besten in Form von Briefmarken beifügen).

Für weitere Fragen stehen wir Ihnen gerne zur Verfügung. Telefonisch erreichen Sie uns unter 089-317 37 84.

Allen Mitgliedern und Lesern wünschen wir ein frohes Weihnachtsfest und ein erfolgreiches, gesundes neues Jahr.

Ulrike Schnitter -FORTH-Büro-

Einladung
zur ordentlichen Mitgliederversammlung
der FORTH-Gesellschaft e.V.

am Sonntag, den 9. April 1989 ab 10:00
an der RWTH Aachen, Steinbachstr. 54, 5100 Aachen

Ab 7. April findet das jährliche FORTH-Treffen statt. Auch in diesem Jahr gibt es wieder interessante Vorträge über FORTH-Anwendungen in Wissenschaft und Industrie.

Tagesordnung:

1. Rechenschaftsbericht des Direktoriums
2. Kassenbericht und Jahresbilanz 1987 und 1988
3. Aussprache und Entlastung des Direktoriums
4. Berichte aus den lokalen - und Fachgruppen
5. Mitgliederversammlung 1990
6. Verschiedenes

Ergänzungen zu den Tagesordnungspunkten können schriftlich bis März 1989 beim FORTH-Büro beantragt werden.

Einladung

zur

FORTH-Tagung 1989

der FORTH-Gesellschaft e.V.

in

Aachen

vom 7. bis 9. April

- Tagungsort** ist die RWTH-Aachen.
(Prof. Dr.-Ing. K.W. Pleßmann, Leiter des Lehr- und Forschungsgebietes für Verfahren der Prozeßdatenverarbeitung und Prozeßführung hat sich dankenswerter Weise bereiterklärt, passende Räume zur Verfügung zu stellen.)
- Vorträge** sollen den gesamten Bereich der Nutzung und Entwicklung der außergewöhnlichen Programmiersprache FORTH betreffen. Professor Pleßmann wird den Eröffnungsvortrag halten. Wenn Sie mit FORTH Erfahrungen gesammelt haben, werden Sie sicher einen interessanten Beitrag in Form eines Vortrages (30 Min.), oder eines Referates (15 Min.) beisteuern können. Bitte schicken Sie bis zum 31.1.89 eine kurze Mitteilung über den Inhalt Ihres Beitrages an die u.a. Adresse.
- Firmen** sind herzlich eingeladen, Soft- und/oder Hardwareprodukte vorzustellen, die mit der Sprache FORTH in Verbindung stehen. Sie werden sicherlich auf ein sachkundiges Publikum treffen. Firmen, die für Ihre Produktinformationen Platz benötigen, sollten dies zeitig mitteilen, damit für entsprechende Räumlichkeiten gesorgt werden kann.
- Unterkunft** finden Sie in zahlreichen Hotels der Stadt. Für Studenten und Schüler sind wir noch auf der Suche nach preiswerten Unterkünften. Die Verpflegung ist durch mehrere Mensen der TH und durch eine unüberschaubare Anzahl von Restaurants und Studentenkneipen gesichert. Mit der Anmeldung erhalten Sie ausführliche Informationen.
- Anmeldeformulare:** erhalten Mitglieder und Freunde der FORTH-Gesellschaft e.V. Anfang Januar 1989 zugesandt (zusätzliche Formulare gibt es beim FORTH-Büro).
- Anmeldungen** an Rolf Kretzschmar, Rote Gasse 7, 5112 Baesweiler

FORTH-Profil(e):

John D. Hall

Mick Ham

Übesetzung aus "Forth Dimensions" Vol IX, Nr. 5 Jan./Feb. 1988: D.Luda

Mike Ham von den "Forth Dimensions" sprach vor einiger Zeit mit einem der Direktoren der Forth Interest Group. John David Hall sprach über seine berufliche Laufbahn, seine Ansichten über FIG und was er von der Forth Vereinigung hält.

MH:

Wie lange sind Sie schon FIG Gruppen-Koordinator?

JH:

Fünf Jahre lang. Ich begann damit 1983. Ich war bei einer Vorstandssitzung und interessierte mich dafür, wie die einzelnen Gruppen die Sitzungsunterlagen erhalten. Es wurde mit gesagt, daß John Cassidy die Unterlagen an die Gruppen schickt. Aber da es ziemlich viel Arbeit ist, würde er wahrscheinlich nicht rechtzeitig damit fertig. Deshalb erklärte ich mich bereit John zu helfen. Daraufhin sagten die Direktoriumsmitglieder: "Großartig, John James braucht wirklich Hilfe."

Ich sagte: "John James?" und sie sagten: "Ja, er versucht gerade die einzelnen Gruppen zu koordinieren." Sie wollten einen offiziellen Kontakt zwischen FIG und den Gruppen herstellen. John James hatte schon damit begonnen, die schriftlichen Sachen zu erledigen, aber er schaffte es nicht alleine. Und so kam es, daß ich zum FIG Gruppenkoordinator wurde.

MH:

Aus wievielen Gruppen bestand die FIG damals?

JH:

Es waren ca. 16 inoffizielle Gruppen. Sie bilden auch heute noch die Basis der offiziellen Gruppen. Wir hatten damals auch schon Kontakte nach Tokio, die New Yorker Gruppe bestand, außerdem die Potomac Gruppe, die Silicon Valley Gruppe und die Orange County Gruppe.

FORTH wird zwar
immer mehr ein-
gesetzt, ist jedoch
noch nicht sehr
populär

MH:

Wieviele Gruppen gibt es jetzt?

JH:

Laut Liste 80. Letztes Jahr schickte ich allen ein Rückmeldungsfomular. Davon kamen ungefähr 50 zurück. Aber auch unter den verbleibenden 30 Gruppen, gibt es welche, die existieren und sehr aktiv sind, wie z.B. die Potomac Gruppe, die auf keinerlei Anfragen reagiert. Neuerdings rufe ich die Leute an und frage, ob es auch wirklich eine Gruppe gibt. Ich dachte, daß die Chicagoer Gruppe aufgelöst wurde. Aber jetzt habe ich

Kontakt zu einem der Mitglieder und habe erfahren, daß die Gruppe ziemlich aktiv ist. Vorher hatte ich mich einfach an den Falschen gewandt. Derjenige, der es vorher gemacht hatte, war weggezogen und hatte seine Arbeit jemandem übergeben, der von unserer Existenz nichts wußte.

MH:

Die Organisation in Gruppen dient auch der Mitgliederwerbung und Motivation.

JH:

Unsere Absicht war es durch die Gruppenstruktur mehr Mitglieder zu gewinnen. Das gelingt von einer Zentrale aus nicht so gut. Deren Aufgabe und Stärke liegt bei den "Forth Dimensions". Neue Mitglieder zu werben, ist Sache der Gruppen.

MH:

Was haben erfolgreiche Gruppen gemeinsam? Können Sie uns sagen, was notwendig ist, damit eine Gruppe gut läuft?

JH:

Ja: Ein dynamischer Leiter oder dynamische Leiter. Bei der Silicon Valley Gruppe z.B. sind wir wirklich erfolgreich, weil es dort einige dynamische Leute gibt. Verläßt einer die Gruppe, nimmt ein anderer seinen Platz ein. Ich habe versucht andere Gruppen zu ermuntern, ob sie nicht auch ein paar dynamische Leute für ihre Gruppe finden. Finden sie sie nicht, sieht es

Interview mit John D. Hall

meist so aus, daß die Gruppe nicht überlebensfähig ist. Man braucht einfach jemanden, der in der Lage ist, anderen weiterzuhelfen und der von der Sache begeistert ist und sich dafür einsetzt.

Wir haben auch herausgefunden, daß eine Person nicht genügt. Einer ist sehr wichtig. Aber sobald dieser eine keine Lust mehr hat, wem übergibt er das Ganze? Im Idealfall benötigt man also zwei oder drei engagierte Leute. Man muß also eine Führungsmannschaft von zwei, drei Leuten finden - also ein Team - damit, falls einer fehlt, die Gruppe nicht zusammenbricht. Damit, wenn dann einer z.B. in Urlaub oder zu einer FORML-Konferenz geht, der Laden weiter läuft. Das ist das ganze Erfolgsrezept.

MH:

Sie erwähnten bei den Treffen der Silicon Valley Gruppe eine Morgen- und eine Nachmittagsitzung. Was ist darunter zu verstehen?

JH:

Als die Silicon Valley Gruppe anging, hieß die Morgen-Sitzung FORML, genau wie die FORML-Konferenz. Der Grund dafür war, daß morgens technische Themen besprochen wurden. Nachmittags zeigten ein paar Leute Applikationen, also Sachen, die sie selbst gemacht hatten. In letzter Zeit ist dieses Konzept etwas durcheinander geraten, wir besprechen technische Themen auch schon mal abends und schauen uns Applikationen genauso morgens wie abends an. So kam die Aufteilung in morgendliche und abendliche Aktivitäten durcheinander. Ich weiß auch nicht genau, wie es dazu kam. Die Zahl der Beiträge nimmt auch ab. In der Silicon Valley Gruppe werden mehr Applikationen als Code vorgestellt. Ich glaube wir bemühen uns einfach nicht genug Leute zu gewinnen, die sich für technische Probleme interessieren.

MH:

Was machen andere Gruppen?

JH:

Die San Diego County Gruppe hat eine interessante Konzeption. Sie treffen sich zur Mittagszeit. Die

Silicon Valley Gruppe trifft sich jeden vierten Samstag im Monat. Meist dauern die Treffen den ganzen Tag, von 10 Uhr morgens bis 5 Uhr nachmittags. Die San Diego Gruppe trifft sich jeden Dienstag ganz zwanglos zur Mittagszeit. Ich habe heute morgen mit ihnen gesprochen. Guy Kelly ist mehr oder weniger der Leiter der Gruppe. Sie sitzen an einem U-förmigen Tisch. Guy sitzt in der Mitte des U's. Jeder, der etwas sagen möchte, geht zur Wandtafel auf das Podium und beginnt zu sprechen. Sie haben keinen Sekretär oder Kassenswart. Sie kommen schon seit fünf oder sechs Jahren in dieser Form zusammen und es klappt ganz gut. Jeder bringt sein Mittagessen mit. Es ist eine gute Gruppe, ein bißchen unkonventionell. Hat jemand etwas Kontroverses zu sagen, steht er einfach auf und sagt es. So kommt oft eine fruchtbare Diskussion zustande.

MH:

Eine Frage zu Ihnen selbst. Wie kamen Sie zu FORTH und wann war das?

JH:

Ich studierte Physik und Chemie in Alaska und Berkley. Kurz vor Beendigung des College ging ich zur Armee. Dort war ich bis 1963. Anschließend ging ich nach Berkley zurück. Zu der Zeit interessierte ich mich nicht mehr so sehr für Chemie wie vorher, aber ich wollte meinen Abschluß machen und das machte ich dann auch. Währenddessen kamen Computer - große Computer - in Mode und ich lernte FORTRAN. Mein Interesse an Computern war geweckt.

Ich kehrte als Zivilist und verheiratet nach Kanada zurück. 1967 ging ich wieder in die USA. Ich entschloß mich mit Computern zu arbeiten. In einer Chemie-Fabrik fand ich eine Stelle als Labor-Assistent. Sie hatten dort einen IBM 1130. Es war ein brandneues Gerät, aber es stand unbenutzt herum. Deshalb schrieb ich ein paar Programme für die Chemiker.

1970 arbeitete ich bei einem Buchprüfer in Oakland, wo ich jetzt wohne, mit einem IBM 1130. Ich habe ab und zu für Buchprüfer und

die University of California im Agricultural Extension Service der eine Menge Registrationen machen muß, gearbeitet. Alles mit einem IBM 1130.

MH:

Mit welcher Programmiersprache?

JH:

Das waren alles professionelle FORTRAN-Applikationen. Als es COBOL auf dem IBM 1130 gab, arbeitete ich damit. Das war 1974. 1975 las ich einen Artikel über den MITS Computer und ich kaufte mir einen Altair Computer. Den baute ich selbst zusammen. Bis dahin hatte ich mich lediglich mit Software beschäftigt, es war das erste Mal, daß ich Lötdraht in die Hand nahm und mit Hardware arbeitete.

Dadurch kam ich zu den Microcomputern. Zu der Zeit war ich immer noch an der University of California beschäftigt. Ungefähr zwei Jahre später kam der Processor Technology SOL Computer heraus. Ich hatte ja vorher schon für den Buchprüfer-Service gearbeitet und unterbreitete denen den Vorschlag zusammen ein Buchprüfer-Programm zu entwickeln. Alles was an Programmiersprachen damals dafür zur Verfügung stand war BASIC. Deshalb begannen meine Schwester, Beckie Harvey, und ich damit ein Buchprüfer-Programm in BASIC zu schreiben.

Das war so ungefähr im August 1980. Als wir in der Zeitschrift "Byte" die Artikel über FORTH lasen, dachten wir, daß das das Richtige für uns sei. Wir hatten versucht, unsere BASIC-Programme so flexibel wie möglich zu gestalten, indem wir sie aus kleinen Tools zusammensetzten. Ich hatte ein "Sort package" das ich von Applikation zu Applikation bringen konnte. Außerdem hatte ich ein "string-data entry"-Tool, das ich transportieren konnte. Diese Module versuchten wir zusammenzufügen und erzeugten so eine modulare Programmiersprache.

Wir begannen daraufhin, alles was wir geschrieben hatten umzuprogrammieren. Wir hatten ein Buchhaltungsprogramm für

Interview mit John D. Hall

Rehabilitationszentren und andere Programme, die alle in BASIC geschrieben waren. Sie waren natürlich ziemlich langsam. Deshalb schrieben wir die Programme nun in FORTH um. Das klappte auch ganz gut, bis die Firma Processor Technology plötzlich Pleite machte. Wir standen dadurch ganz schön im Regen.

Wir hätten zwar weitermachen und die paar uns noch verbleibenden Prozessoren verkaufen können. Ungefähr 12 davon hatten wir in Rehabilitationszentren installiert, sie waren dort für die Buchhaltung zuständig. Aber die Firma existierte nicht mehr. Was also tun? Wir waren in einer Sackgasse gelandet.

Deshalb trennten Beckie und ich uns 1981. Sie ging nach Los Angeles zu einer Firma, bei der sie in FORTH programmiert. Ich wurde Berater und fand ein paar Jobs bei denen ich FORTH gebrauchen konnte. Einer davon war für eine Firma mit dem Namen Stafa. Dort begannen sie mit der Entwicklung eines Reglers für Röhren. Sie hatten einen Regler in die Röhren eingebaut und man konnte mit dieser Einrichtung mit Hilfe von AC-Verbindungen kommunizieren. Es handelte sich dabei um einen FORTH-Chip.

MH:

Den 8080er ?

JH:

Nein, aber es war ein ähnlicher Chip wie der 8080er. Ein 8080 Kontroller-Chip mit allem was dazu gehört, wie A/D wandler, einfach allem. Ein einziger Chip, der die ganze Hardware auf sich vereinigte.

MH:

Das war ein ganz schön langer Weg vom 1130er, nicht wahr?

JH:

Ja, in der Tat! 1130er wurden ursprünglich als Minicomputer konzipiert, sie waren als wissenschaftliche Minicomputer angelegt. Wir aber hatten sie als Bürocomputer eingesetzt, da sie nicht so teuer waren.

Das Stafa-Projekt nahm ca. 8 Monate in Anspruch. Ungefähr zu dieser Zeit mußte Gary Feierbach ein Testgerät für eine Firma in Ventura bauen. Paul Thomas, Gary Feierbach, Mathew Johnson und ich waren ein Team. Das Projekt war das, was ich als typisches FORTH-Projekt bezeichnen würde: Es sollte damit das Schlimmste verhütet werden. Irgendjemand hatte ein Projekt in Angriff genommen, eine Menge Geld hineingesteckt und gedacht es mit einer herkömmlichen Programmiersprache verwirklichen zu können, aber das klappte nicht. Also kamen sie zu uns und sagten: "Die Sache muß unbedingt erledigt werden und zwar bis zu einem bestimmten Termin. Wir haben gehört, daß es mit FORTH möglich ist, es schnell über die Bühne zu bringen - bitte übernehmt das für uns."

Gary, der damals FORTH unterrichtete und einige Hardwaresachen machte, entschloß sich das Projekt anzunehmen.

Das Projekt war nicht sehr klar umrissen. Und als wir daran arbeiteten, änderten sich die Anforderungen. Der Kunde war der Meinung, daß es doch möglich sein müßte, das Ganze etwas anwenderfreundlicher zu gestalten - "anwenderfreundlich" war zu dieser Zeit das Zauberwort. Nachdem ein Jahr vergangen war, stellte sich heraus, daß wir uns um 180 Grad von den ursprünglichen Spezifikationen entfernt hatten. Anfangs war es geplant, einen Kontroller ohne große Benutzeroberfläche zu entwickeln, der lediglich der Maschinenkontrolle dienen sollte. Am Ende kam aber eine Workstation heraus, die eine recht umfangreiche Benutzeroberfläche besaß.

Bei den meisten Applikationen macht die Schnittstelle zum Anwender gut 50% des Projektes aus, so daß sich der Entwicklungsaufwand verdoppelte.

MH:

Kam das Ganze als Produkt auf den Markt?

JH:

Ja, aber das Projekt wurde von einer anderen Gruppe fertiggestellt.

Nach dieser Sache ging ich zu Rising Star. Die waren seit ca. einem Jahr dabei einen Personalcomputer zu entwickeln. Es handelte sich dabei um einen Computer, der nach dem "turnkey"-Prinzip funktionierte (einschalten und arbeiten). Schaltete man ihn an, erschien ein Editor. Drückte man eine Zifferntaste, erschien die Tabellenkalkulation, drückte man "mail", kam man in das Programm zum Briefe schreiben. Durch Betätigen der "copy"-Taste konnte man eine Diskette kopieren usw. Das System war einfach und deshalb auch für Anfänger geeignet. Damals gab es den IBM PC noch nicht so richtig, deshalb versuchten sie das Ganze auf einem Epson mit einem Z-80 Prozessor zu realisieren. Das Interessante daran war, daß wir zeigen konnten, zu was ein Z-80 fähig ist. Das war wirklich erstaunlich. Wir hatten mehrere Datenspeicher mit 64K. In einem der Speicher legten wir die Tabellenkalkulation ab, in einem anderen den Editor, das Betriebssystem in einem dritten und so fort. Ja und natürlich die Graphik - wir hatten eine Graphik, wie sie zu der Zeit niemand sonst hatte.

MH:

Sie zeichneten doch die Briefe auf dem Bildschirm, nicht wahr?

MH:

Ja, bei dem Editor handelte es sich um einen "What-You-See-Is-What-You-Get" Editor. Es gab alles Fettschrift, Kursivschrift und vieles mehr. Das meiste davon war in FORTH geschrieben, in High-level-FORTH. Das Betriebssystem hatte eine Firma in Assembler geschrieben, die anfangs, als CP/M aufkam. Sie hatten sich ihr eigenes CP/M mit dem Namen TP/M erstellt. Jetzt waren sie dabei, es mit denselben Leuten und mit ihrem eigenen Betriebssystem umzuarbeiten, CP/M kompatibel aber mit Erweiterungen. Sie hatten einen Graphik-Spezialisten. Er kümmerte sich um den Graphik-Prozessor. Man konnten den Graphik-Prozessor aufrufen, um eine Linie zu zeichnen oder den Zeichensatz zu ändern und was es sonst noch so gibt. Die Zeichen wurden also nicht mit FORTH gezeichnet, sondern dazu wurde der Graphik-Prozessor

Interview mit John D. Hall

aufgerufen. Aber um es noch einmal zu sagen, das alles wurde mit einem Z-80 gemacht, mit ein und demselben Z-80, den wir von verschiedenen Stellen aus aufriefen.

MH:

Schaltete man also von einer Applikation um, landete man im Betriebssystem und von dort durch Umschalten in einer anderen Speicherbank - kann man sich das so vorstellen?

JH:

Ja.

MH:

Was für eine Aufgabe hatten Sie beim Rising Star Projekt?

JH:

Das änderte sich dauernd. Ursprünglich sollte ich bei der Erstellung der Menüs helfen. Zwischen den Applikationen gab es Menüs, die es ermöglichten, daß man zwischen den einzelnen Applikationen hin- und herschalten konnte. Mich störte aber bald das verwendete FORTH, das weit unter seinen Möglichkeiten blieb, es war zu schwerfällig. Deshalb entschloß man sich das zu Grunde liegende FORTH neu zu installieren. Das war eine Aufgabe für drei Personen, von denen ich eine war. Wir installierten ein vereinfachtes 83-Standard-Modell, mit separat verwalteten Headern (wie beim Original F83).

Das legten wir dann unter die Applikationen - die immer noch im Aufbau waren. Das was schon am weitesten fertig war, war die Tabellenkalkulation. Aus diesem Grund nahmen wir sie und plazierten unter ihr das neue FORTH. Dabei stellte sich heraus, daß die Tabellenkalkulation nicht verändert werden mußte. Wir hatten also das neue FORTH so gut dem alten System angeglichen, daß keine Schwierigkeiten auftauchten. Das überraschte mich. Ich hatte gedacht, daß es bestimmt ein paar Tools geben würde, die wir nicht integriert hätten, aber das war nicht der Fall. Es paßte alles perfekt zusammen.

Rising Star befand sich aber in einer gefährlichen Lage. Sie erhielten Geld von Epson. Nur so war es ihnen möglich ca. 30 Programmierer zu beschäftigen. Diese Programmierer waren über die ganze USA verstreut. Es gab ein Team von ungefähr 6 oder 7, die das Betriebssystem und das Programm zum Briefeschreiben in Assembler programmierten. Sie waren im Osten des Landes. Dann gab es noch ein paar Leute in Oregon. Sie waren für die Dokumentation des neuen FORTH zuständig, das wieder von einer anderen Gruppe entwickelt wurde. Ich lebte in Oakland. Paul Thomas in San Francisco und Ron Braithwaite im Bezirk San Diego.

Es war das erste Mal, daß ich an einem Projekt beteiligt war, bei dem jeder für sich zu Hause programmierte und die einzelnen Module auf einem Bulletin-Board auf dem Computer des Projektleiters hin- und hergeschickt wurden. Wir arbeiteten alle mit dem Epson Computer. Er war weit genug entwickelt, um unseren Zwecken zu dienen. Man konnte die Module mit Hilfe des Bulletin-Boards ein- und ausladen. Jeden Monat kamen wir aus allen Ecken der USA einmal im Hauptquartier in Los Angeles/Torrance zusammen.

Das war schon eine tolle Sache, ideal für Programmierer. ... Wir konnten ziemlich selbstständig arbeiten, das kommt ja besonders FORTH-Programmierern gelegen. Wir möchten alleine an einem Projekt arbeiten und die einzelnen Tools zu einem Ganzen zusammenfügen. So kann jeder sein spezielles Wissen einbringen. Da fühlt man sich freier, als bei der Arbeit im Büro. Im Prinzip würde man ja im Büro dieselbe Arbeit machen, aber da fühlt man sich oft gestört und beobachtet. Arbeitet man zuhause, so schafft man ein größeres Pensum in kürzerer Zeit, da man nicht um fünf Uhr heim geht sondern bis um 10 oder 12 Uhr nachts arbeitet.

Das Projekt Rising Star wurde im Dezember 1984 beendet. Ich hatte ungefähr die Hälfte der Zeit - acht Monate - daran mitgearbeitet. Es war nämlich so, daß IBM immer marktbeherrschender wurde. Epson entschloß sich deshalb kein weiteres Z-80 Produkt einzuführen. Sie

wollten jetzt ein 8088er. Da sie aber dachten, daß wir das Z-80 Projekt noch zu Ende bringen könnten, ließen sie uns trotzdem noch etwas Zeit. Meiner Meinung nach hätten sie uns sonst schon im August oder September gekündigt, aber sie dachten: "Das Projekt steht kurz vor der Beendigung und kann bald herausgebracht werden." Deshalb ließen sie uns eine Galgenfrist bis Dezember, bevor sie uns den Geldhahn zudrehten. Wir wurden aber bis dahin nicht fertig und das Projekt Rising Star brach im Dezember zusammen.

Wenn man irgendwo die Schuld suchen will, so lag es am Editor. Es war ein typischer FORTH-Zeileneditor. Das Prinzip eines Zeileneditors besteht darin, eine einzige Zeile in der Mitte des Bildschirms zu editieren. Der Text kann auf dem Bildschirm nach oben oder unten gescrollt werden und jede beliebige Zeile kann editiert werden. Aber eine Textzeile nimmt einen bestimmten Raum ein. Den muß man markieren - und so ergaben sich eine Menge Komplikationen. Dem Projekt lag aber ein bestimmtes Modell zu Grunde, dessen Rahmen es bald sprengte.

Das Grundkonzept hätte viel früher geändert werden müssen. Rising Star gab viel Geld dafür aus, Leute in Hotels unterzubringen. So wurde z.B. das Editor-Team für drei Wochen in ein Hotel in Klausur gesteckt, damit das Produkt endlich fertig würde. Es war eine sehr schwierige Situation. Es war keine Zeit mehr zum Überlegen. Das Projekt hatte sich festgefahren.

MH:

Packt man eine Sache von der falschen Seite an, kann es sehr lange dauern, bis man es merkt. Aber es gibt immer einen Punkt, wo man sagt "Das ist der falsche Weg, wir müssen noch einmal an den Ausgangspunkt zurück und alles überdenken."

JH:

Ja, natürlich müßte man noch einmal von vorn beginnen, aber sobald Geld im Spiel ist, gibt es automatisch eine zeitliche Begrenzung, ein guter Ruf ist zu verteidigen und so ist es manchmal ein-

Interview mit John D. Hall

fach unmöglich noch einmal neu zu beginnen. Ist es so, ist ein tragisches Ende unausweichlich. Das ist zwar schwer zu begreifen, aber leider ist es oft so.

MH:

Nach Beendigung des Rising Star Projektes wurden Sie wieder Berater.

JH:

Ich arbeitete auch für Rising Star als Berater. Von Stafa über Inner Access zu Rising Star war ich immer Berater. Der Fehlschlag von Rising Star traf mich ziemlich unvorbereitet und so dauerte es fast bis März eine neue Stelle zu finden. Ich übernahm einen Posten bei Lockheed Research und Development in Palo Alto. Die Arbeit dort stellte sich bald als ziemlich interessant heraus. Die Forschungs- und Entwicklungsabteilung von Lockheed befindet sich in der Nähe von Stanford. Dadurch herrscht in dieser Gegend eine akademische Atmosphäre. Ich dachte, daß das eine Möglichkeit sei etwas mit FORTH zu machen, wobei nicht nur Code herauskäme. Nicht daß das bei meinen früheren Jobs so gewesen wäre - aber es gab noch einiges zu erforschen. Das war 1985 und nun arbeite ich schon seit fast drei Jahren dort.

MH:

Sie programmieren also in FORTH?

JH:

Ja. Unsere Hauptabteilung heißt 'Angewandte Physik'. Wir sind eine Untergruppe von ca. 10 Leuten und unsere Bezeichnung ist "Schnelle Prozessoren". Wir arbeiten an der Entwicklung von Sensoren. Der Leiter der Gruppe hat großes Interesse an FORTH. Einige sind für die Software zuständig, andere für die Hardware. Aber bei FORTH gibt es keine klare Trennung zwischen den für Software oder Hardware zuständigen. Ich z.B. habe mehr über Hardware gelernt, als ich mir je vorgestellt habe.

Durch FORTH wurden mir die Augen für Dinge geöffnet, die ich früher für mich als "Black Boxes" bezeichnete. Auf dem IBM 1130

konnte ich zwar die Programmiersprache, aber ich hatte keine Ahnung was das Betriebssystem machte, ich wußte nichts über die Gerätetreiber - alles außer der Programmiersprache und der Applikation war eine "Black Box" für mich. Heute gibt es keine "Black Boxes" mehr. FORTH ist so einfach, ich steige da voll durch und sehe, daß die Sachen, die ich früher "Black Boxes" genannt habe, unnötig kompliziert aufgebaut sind. Es sind komplizierte Strukturen, die eigentlich so gar nicht existieren müßten. Es kann sein, daß die Geräte früher primitiver waren und daß deshalb manches etwas komplexer ausfallen mußte.

MH:

Mit welchem Prozessor arbeiten Sie momentan?

JH:

Als ich bei Lockheed anfang, arbeiteten wir mit Intel Entwicklungsgeräten. Genaugenommen mit 8086ern und 286ern die parallel liefen. Sie kommunizierten über einen einfachen Speicher auf einem Bus. Es gab 4 unabhängige CPU-Boards, die über einfachen Speicher kommunizierten. Jedes einzelne Board sammelte, speicherte, manipulierte und protokollierte die Daten.

Jetzt arbeitet unsere Gruppe mit den Novix 4016 Chips. Wir haben herausgefunden, daß man mit dem 4016 Hardware durch Software ersetzen kann. Möchte jemand einen Sensor, so nehmen wir den 4016, verbinden ihn mit was immer abgetastet werden soll und lassen das Ganze so schnell ablaufen, daß dazwischen keine Hardware benötigt wird. Wir brauchen überhaupt wenig Hardware - höchstens einen A/D-Wandler. Es ist uns möglich, viele sehr schnelle Abtastungen durchzuführen.

Wir benutzen Computer Cowboy Boards. Im Augenblick arbeiten wir an einem Projekt, bei dem versucht wird, ein paralleles Novix-System aufzubauen, bei dem viele Novix-Chips und Boards zusammenarbeiten.

MH:

Wie viele?

JH:

Wir haben mit 10 unabhängigen CPU's begonnen, die parallel geschaltet sind. Gerade sind wir dabei, dafür die Software zu entwickeln. Die Arbeit wird von einem Leiter an die einzelnen Mitarbeiter vergeben. Der Leiter schreibt die Anweisungen auf eine Tafel. Sobald die Arbeit ausgeführt ist, wird das Ergebnis auf der Tafel festgehalten und die nächste Aufgabe kann übernommen werden.

MH:

Nun zu einem ganz anderen Thema - FIG. Wohin bewegt sich FIG Ihrer Meinung nach?

JH:

FIG nicht FORTH ist auf einem bestimmten Level angekommen. Mein Eindruck ist, daß FORTH zwar mehr und mehr gebraucht wird - aber immer noch weit unter seinen Möglichkeiten eingesetzt wird. Es geht jedoch kontinuierlich aufwärts: Solche Sachen wie Rapid File, VP Planner, die Cannon Cat - alles Beispiele dafür, für was FORTH gebraucht wird - werden immer häufiger. FIG dagegen stagniert im Moment.

MH:

Das hört sich so an, als ob die FIG der Programmiersprache FORTH beim Flügel werden half und dann als sie etabliert war, suchte die FIG nach einem neuen Betätigungsfeld. Ich glaube aber, daß die FIG den Leuten beim Einstieg in FORTH hilft. Die Aufgabe der FIG hinsichtlich fortgeschrittenem FORTH ist jedoch nicht klar zu erkennen.

JH:

Da haben Sie recht. FORTH wurde von der Allgemeinheit der Programmierer nicht benutzt, bevor es die FIG gab. Die einzigen beiden Firmen, die mit FORTH arbeiteten, waren FORTH, Inc. und Miller Microcomputer Services. Die Absicht der FIG-Gruppe bestand darin, eine Version von FORTH zu kreieren, die für alle zugänglich und verständlich ist. Es kam ein gutes

Interview mit John D. Hall

FORTH dabei heraus, das Interesse weckte. Und heutzutage kann jeder es erlernen und verstehen.

Da FORTH wie vorher schon gesagt, die Black Boxes beseitigen hilft, können Leute, die vorher nicht wußten, wie man in einen Compiler kommt, das nun ohne Schwierigkeiten. Bevor es FORTH gab, wurden Programmiersprachen einfach benutzt. Es wurde nicht hinterfragt wie sie aufgebaut sind, warum es sie gibt und wie sie funktionieren. FORTH ist beides Betriebssystem und Programmiersprache. Deshalb erschließt sich dem FORTH-Benutzer das Betriebssystem. Vielen Leuten wurden die Augen geöffnet, so daß sie jetzt wissen, wie leistungsfähig FORTH ist.

Martin Tracy nennt diese Leute Hobbyanwender. Ich dagegen würde sagen, jemand, der sich für so etwas interessiert, ist ein Profi. Er kennt sich sowohl mit Hardware, als auch mit Software aus. Man kann durchaus von anderen Fachgebieten kommen und hat durch FORTH dann die Möglichkeit Microcomputer von Grund auf kennen und verstehen zu lernen.

Heute kauft man sich nicht mehr einfach einen Ismai, Altair oder Sol, baut ihn selbst zusammen und schreibt seine eigene Programmiersprache dazu. Man geht vielmehr in einen Laden, kauft sich einen Macintosh und kriegt fertige Applikationen vorgesetzt. Vielleicht will man dann auch einmal eine eigene Applikation schreiben. Dazu muß man aber von dem Level, auf dem man sich befindet, zurück gehen. Das Anwenderinterface und viele andere Dinge mehr müssen erstellt werden. Deshalb gehen die Leute heute von anderen Voraussetzungen aus als früher. Früher erwartete man gar nichts - alles mußte selbst aufgebaut werden. FIG-FORTH war da ein gutes Werkzeug, um alles zu bewältigen. Solche Leute - sie kann man Hobbyanwender nennen - gibt es heute kaum mehr.

50% der FIG-Anwender sind Hobbyanwender und 50% Profis. Wir haben versucht, beide Gruppen unterschiedlich anzusprechen und zu unterstützen. Wir sind zu dem Schluß gekommen, daß FIG-Hobbyanwender hinreichend zufrieden-

stellt und die Basis für Profianwender darstellt, diese jedoch nicht ganz befriedigen kann. Deshalb sind wir auf der Suche nach Möglichkeiten diese Lücke zu füllen.

MH:

Sollen die Profianwender Ihnen mitteilen, was sie wollen und brauchen?

JH:

Meinungen sind uns immer willkommen. Aber was sie brauchen und wollen ist eigentlich schon klar. Sie brauchen Geld und moralische Unterstützung. Sie wollen nicht, daß ihre Chefs sagen "Was ist FORTH und für was ist es zu gebrauchen? Wieso arbeiten Sie nicht mit C?". Sie möchten, daß ihre Chefs sagen "Es ist eine wunderbare Idee, daß Sie FORTH einsetzen wollen."

Ich glaube, daß wir einiges tun können, um dieser Einsicht Vorschub zu leisten. Darüber haben wir bei einer Versammlung geredet. Wir müssen Möglichkeiten finden diesen Leuten zu helfen. Wenn nicht direkt, so indirekt, indem wir die Augen der Vorgesetzten öffnen.

MH:

Was muß FIG Ihrer Meinung nach alles tun.

JH:

Öffentlichkeitsarbeit mit allen Mitteln. In der Vergangenheit haben wir das allein den "FORTH Dimensions" und den FORTH-Gruppen überlassen. Neuerdings ist noch 'GENIE' hinzugekommen. Dabei handelt es sich um eine neuartige Form der Kommunikation. Ich bin gerade dabei zu begreifen, daß es eine breite Palette von Kommunikationsmöglichkeiten gibt. Wir brauchen eine monatliche Zeitschrift, wir brauchen Gruppen, die direkt mit anderen Gruppen in Verbindung treten (GENIE kann dabei sehr nützlich sein), wir müssen Autoren ermutigen, einführende Artikel über FORTH zu schreiben, die in Computerzeitschriften plaziert werden können.

Könnte ich noch einmal von vorn anfangen, würde ich FORTH nicht mehr direkt fördern, sondern würde interessante Applikationen vor-

stellen und beiläufig erwähnen, daß sie in FORTH geschrieben wurden, statt zehn Mannjahre nur zwei Mannjahre lang dauern und soundsoviele Codezeilen X Dollars kosten.

Auf diese Art und Weise würde ich es versuchen. Ich kenne eine Menge Leute, die mir gesagt haben, daß wir es so machen sollen. Aber ich wollte es nicht hören und die meisten anderen leider auch nicht.

MH:

Ein großes Problem von FORTH ist seine Evidenz. Es ist immer schwierig, über das zu sprechen, was offensichtlich ist. Arbeitet man mit FORTH, kann man erzählen, wie produktiv man ist, wieviel man mit sehr wenig Aufwand leistet. Man kann erzählen, wie die verschiedenen Tools und das Wissen sich zu einer immer größeren Einsicht zusammenfügen, die man bei FORTRAN und Assembler niemals erreicht. Es ist einfach zu offensichtlich zum Erklären. Deshalb werden FORTH-Anwender gerne als Fanatiker hingestellt. Fragt man einen FORTH-Anwender, warum er mit FORTH arbeitet, kann er es einfach nicht erklären. Er sagt dann: "Benutze es einfach." Die meisten denken dann: "Ja, ja wieder so ein FORTH-Freak."

JH:

Etwas was ich schon immer komisch fand ist, warum verwenden Assembler-Programmierer FORTH nicht? Es ist doch leicht zu verstehen, daß sie mit FORTH eine Applikation in Assembler schreiben könnten und dadurch eine Interaktivität bekämen, wie sie mit Assembler niemals zu erreichen ist. Bei Assembler gibt es einfach nichts vergleichbares. Das wundert mich sehr, daß diese Leute FORTH nicht benutzen. Das wäre wirklich ein großes Potential, aber der Zug ist glaube ich abgefahren, sie arbeiten nun alle mit C.

FORTH-WINDOWS - Fenstertechnik für die Schneider CPCs

Claus Kühnel

Die Schneider CPCs sind bezogen auf ihre Rechnerklasse mit einer Vielzahl hervorragender Eigenschaften ausgestattet, die durch den BASIC-Befehlssatz gut unterstützt sind. Damit ist die Nutzung dieser Eigenschaften auf einfache Weise möglich.

Mit der FORTH-Erweiterung WINDOWS sollte die einfache und wenig aufwendige Möglichkeit der Nutzung der komfortablen Eigenschaften der Schneider CPCs unter FORTH verdeutlicht werden.

Mit dem cpcFORTH (Fa. FORTH-SYSTEME Angelika Flesch, Breisach) steht auch eine sehr komfortable FORTH-Implementierung für die Schneider CPCs zur Verfügung. Auch unter FORTH kann von den im Betriebssystem enthaltenen Routinen vorteilhaft Gebrauch gemacht werden. Anhand der FORTH-Erweiterung WINDOWS sollen Vorgehen und Anwendung betrachtet werden.

Im Dictionary des cpcFORTH sind die Fenstertechnik betreffend die Worte STREAM zur Umschal-

tung des aktiven Ein-/Ausgabegerätes und das Wort WINDOW zur Definition eines Textfensters im laufenden E/A-Gerät enthalten. Das Grafikfenster bezieht sich auf den ganzen Bildschirm. Der Ursprung läßt sich aber mit ORG beliebig setzen. Es liegt in der Natur von FORTH, daß man sich mit den gegebenen Eigenschaften nicht abfinden muß.

Eine Schlüsselstellung in der vorliegenden FORTH-Erweiterung nimmt das Definitionswort KOORD ein. Bei der Definition eines Textfensters werden die Fensternummer (Stream) und die Fensterkoordinaten einem Fensternamen zugeordnet. Beim Aufruf des so definierten Fensters übergibt das Wort die Daten auf den Stack. Das in Screen #7 enthaltene Demo verdeutlicht die Anwendung. Durch die Angabe 1 0 0 39 20 KOORD WINDOW1 wird ein Textfenster mit dem Namen WINDOW1 definiert, welches dem Stream #1 zugeordnet wird und sich über die Zeilen 0 bis 20 und die Spalten 0 bis 39 erstreckt. Die beim Aufruf auf dem Stack bereitgestellten Daten werden von den Worten FRAME und SELECT benötigt. Durch das Wort FRAME erfolgt ein Umranden des Textfensters, was dann zwangsläufig um diesen Rand verkleinert wird. Die um diesen Rand reduzierten Fensterkoordinaten hinterläßt das Wort wiederum auf dem Stack.

Erst SELECT definiert dann das eigentliche Textfenster. Es ergeben sich zwei Möglichkeiten der Definition. Soll das Fenster mit einem Rahmen versehen werden, dann muß der Aufruf beispielsweise durch WINDOW1 FRAME SELECT erfolgen. Nach dieser Wortfolge beziehen sich alle weiteren Textausgaben auf das ausgewählte Textfenster. Die Fensterumschaltung kann durch das Wort STREAM in Verbindung mit dem gewünschten Fensternamen (8 WINDOW1 STREAM) vorgenommen werden. Die überflüssigen Fensterkoordinaten und der von STREAM hinterlassene alte Wert werden vom Stack entfernt. Das ebenfalls in cpcFORTH enthaltene Wort CLS löscht das aktuelle Fenster und positioniert den Cursor links oben (Home Up). Soll auf das Löschen verzichtet werden, kann der Cursor durch HOMEUP positioniert werden.

Für die Definition eines Grafikfensters wurden vier neue PRIMITIVE WORDS definiert, die Bezug auf Routinen des Betriebssystems nehmen. Zum cpcFORTH mitgeliefert wird ein 8080/Z 80-Assembler, der die Definition von neuen PRIMITIVE WORDS ganz einfach über das Wort CODE zuläßt. Das Betriebssystem enthält u.a. Routinen zum Setzen der Breite und der Höhe des Grafikfensters. Fenstergrenzen werden automatisch auf Bildschirm-Bytegrenzen verlegt. Benutzt man später die Grenzen des Grafikfensters (z.B. bei der Normierung einer Wertefolge), sollten die aktuellen Grenzen, die durchaus von den vorgegebenen geringfügig abweichen können, abgefragt werden. Zur Formulierung der Grafikroutinen ist die Kenntnis der Adressen der Sprungtabelle und der Ein- bzw. Ausprägungsbedingungen notwendig. Zum Setzen der Breite des Grafikfensters sind die linke und rechte Fensterbegrenzung in den Doppelregistern DE und HL zu übergeben. Bevor die Routine durch CALL BCBF aufgerufen wird, ist unbedingt das Doppelregister BC (Instruction Pointer) zu retten. Die Routine zum Setzen der Höhe des Grafikfensters wird in der Sprungtabelle an der Adresse BBD2 vorgefunden. Der restliche Ablauf ist gleich. Das Verfahren bei der Abfrage der Fenstergrenzen ist praktisch umgekehrt. Die Routinen

FORTH-WINDOWS

übergeben die Koordinaten in den Doppelregistern DE und HL, die dann nur noch auf den Stack "gePUSHt" werden müssen. Das Grafikfenster kann durch Aufruf des Wortes GRFRAME umrandet werden.

Den Abschluß bildet das in Screen #7 enthaltene Demo. Es werden zwei Textfenster und ein Grafikfenster definiert. Zur Ausgabe im

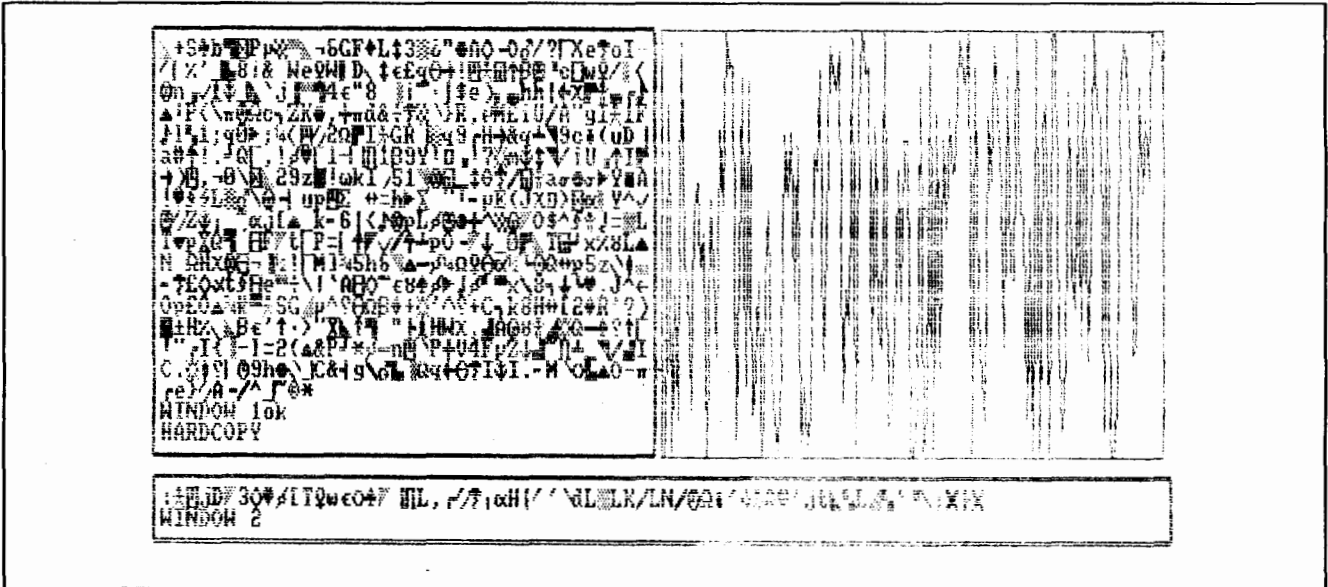
Textfenster dient das Wort NON-SENS, welches 1000 über das Wort RANDOM zufällig bestimmte Zeichen an das ausgewählte Fenster sendet. Im Grafikfenster wird Rauschen dargestellt, welches ebenfalls mit dem Wort RANDOM erzeugt wird (siehe Bild unten).

Die vorgestellte Erweiterung belegt 563 Byte (ohne Demo) und ist damit sehr kompakt. Durch die

Verbindung von Text- und Grafikfenstern ergeben sich günstige Möglichkeiten für die Gestaltung eines Dialoges mit dem Bediener.

Dr. Claus Kühnel
Zschertnitzer Str. 52
DDR-8020 Dresden

Registriernummer des Büros für
Urheberrechte der DDR: VZA-
2052-88



Screen # 0
(WINDOWS:

CK 20/ 7/88)

Die FORTH-Erweiterung WINDOWS erlaubt die von BASIC her gewohnte Fenstertechnik mit den Schneider CPCs.

Verwendet wurde das cpcFORTH der
Fa. FORTH-SYSTEME Angelika Flesch PF 1103 7814 Breisach.

Zur Verdeutlichung der Anwendung ist in Screen #7 ein Demo
enthalten.

FORTH-WINDOWS

Screen # 1
(WINDOWS: Load Screen CK 20/ 7/88)

DECIMAL

CR SP@ ^STR - U. .(bytes in dictionary)

CR CR .(Wait ... loading WINDOWS) CR CR

2 7 THRU

.(WINDOWS loaded.)

CR CR SP@ ^STR - U. .(bytes left in dictionary)

CR CR

Screen # 2
(WINDOWS: KOORD CK 20/ 7/88)

```
: KOORD CREATE C, C, C, C, C,  
  DOES> DUP C@ SWAP 1+  
         DUP C@ SWAP 1+  
         DUP C@ SWAP 1+  
         DUP C@ SWAP 1+ C@ ;
```

(Definition von Textfenstern:

Fenster# xlinks yoben xrechts yunten KOORD Fenstername)

Z-80 cpcForth 1.1 (Forth-83) by Forth-Systeme A. Flesch 1985

FORTH-WINDOWS

Screen # 3

(WINDOWS: FRAME

CK 20/ 7/88)

```
: FRAME      ( yu xr yo xl Fenster# -- yu' xr' yo' xl' Fenster# )
  0 STREAM DROP
  OVER 3 PICK GOTOXY 150 EMIT
  3 PICK 2 PICK 1+ DO 154 EMIT LOOP 156 EMIT
  4 PICK 3 PICK 1+ DO
    OVER 1 DUP ROT SWAP GOTOXY 149 EMIT
    4 PICK SWAP      GOTOXY 149 EMIT LOOP
  OVER 5 PICK GOTOXY 147 EMIT
  3 PICK 2 PICK 1+ DO 154 EMIT LOOP 153 EMIT
  4 ROLL 1- 4 ROLL 1- 4 ROLL 1+ 4 ROLL 1+ 4 ROLL ;
```

Screen # 4

(WINDOWS: >STREAM SELECT HOMEUP

CK 20/ 7/88)

```
: >STREAM      ( yu xr yo xl Fenster# -- )
  STREAM DROP 2DROP 2DROP ;

: SELECT      ( yu xr yo xl Fenster# -- )
  STREAM DROP
  ROT SWAP 2SWAP WINDOW ;

: HOMEUP      ( -- )
  0. GOTOXY ;
```

Screen # 5

(WINDOWS: GRAFIK1

CK 20/ 7/88)

HEX

```
BBCF CONSTANT ADDRSETWIDTH
BBD2 CONSTANT ADDRSETHEIGHT      ( Firmwareadressen )
BBD5 CONSTANT ADDRGETWIDTH      ( Schneider CPC 6128 )
BBD8 CONSTANT ADDRGETHEIGHT
```

DECIMAL

CODE SETWIDTH

D POP H POP B PUSH ADDRSETWIDTH CALL B POP NEXT

CODE SETHEIGHT

D POP H POP B PUSH ADDRSETHEIGHT CALL B POP NEXT

FORTH-WINDOWS

Screen # 6

(WINDOWS: GRAFIK2

OK 20/ 7/88)

CODE GETWIDTH

B PUSH ADDRGETWIDTH CALL B POP D PUSH H PUSH NEXT

CODE GETHEIGHT

B PUSH ADDRGETHEIGHT CALL B POP H PUSH D PUSH NEXT

: GRFRAME

(--)

```
  GETWIDTH  GETHEIGHT  3 PICK  2 PICK  MOVE
  2 PICK  2 PICK  DRAW  2 PICK  OVER  DRAW
  3 PICK  OVER  DRAW  3 PICK  2 PICK  DRAW
  2DROP  2DROP ;
```

Screen # 7

(WINDOWS: DEMO

OK 20/ 7/88)

1 0 0 39 20 KOORD WINDOW1 (Definition von zwei)

2 0 21 79 24 KOORD WINDOW2 (Textfenster)

320 630 SETWIDTH (Setzen der Koordinaten)

70 390 GETHEIGHT (des Grafikfensters)

: NOISE 641 320 DO I 350 RANDOM 50 + DRAW 2 +LOOP ;

: NONSENS 1000 0 DO 256 RANDOM EMIT LOOP ;

: DEMO

WINDOW1 FRAME SELECT CLS NONSENS CR ." WINDOW 1"

WINDOW2 FRAME SELECT CLS NONSENS CR ." WINDOW 2"

GRCLS GRFRAME NOISE 1 STREAM DROP ;

: RUNDEMO 10 0 DO DEMO LOOP ;

Screen # 8

Der Kopierservice der FORTH-Gesellschaft e.V.

Das Archiv der FORTH-Gesellschaft befindet sich in Hamburg. Hier werden alle Publikationen erfasst und gelagert, die uns von den Mitgliedern zur Verfügung gestellt werden. Auf dieses Archiv baut der

Kopierservice als Dienstleistung für die Mitglieder

auf. Wichtige Artikel werden von uns in einer Bestellliste geführt und können von den Mitgliedern bestellt werden. Außerdem versenden wir Inhaltsverzeichnisse von FORTH-Publikationen. Einzelne Artikel aus diesen Publikationen können bei uns bestellt werden.

Bestellungen werden innerhalb einer Woche bearbeitet. Die Seite kostet DM 0,30. Zur Verwaltungsver-einfachung bearbeiten wir nur Bestellungen, die per Vorauskassa bezahlt wurden. Je Bestellung wird eine Bearbeitungs-/Versandkostenpauschale von DM 3,- berechnet. Die Bezahlung erfolgt entweder durch Zusendung von Briefmarken oder durch Überweisung auf das Postgirokonto des Kopierservice:

FORTH-Gesellschaft e.V.
Sonderkonto K
Postgiroamt Hamburg
Nr. 5226 46 - 203
BLZ 200 100 20

Im folgenden der erste Bestellzettel des Kopierservice. Bei Bestellungen bitte neben der gewünschten Position die Anzahl der Seiten eintragen, am Ende die Gesamtanzahl der Seiten zusammenzählen, mit DM 0,30 multiplizieren, DM 3,00 Versandkosten hinzuzählen und auf das Kopierkonto überweisen. Bei Überweisung von einem Girokonto kann der Bestellzettel unmittelbar an den Überweisungsauftrag angeheftet werden. Sonst bitte Überweisungsbeleg - oder einfach den Betrag in Briefmarken beilegen.

Sollten Sie wichtige Artikel, Zeitschriftenausschnitte etc. haben, die auch für andere Mitglieder interessant sein könnten, so schicken Sie uns diese bitte zu. Wir werden sie ins Archiv und den Kopierservice aufnehmen.

BES#...TTITEL.....QUELLE..AUTOR....SEIT.

LS001...Forth auf F65F11	MC 1 '85 . R.Zech.....	5
LS002...Multitasking in Forth	Micro 9 '84 . Butterfield.....	6
LS003...Structure Trees.....	Micro 9 '84 . M.Dougherty.....	3
LS004...Faster Forth.....	Byte 6 '84 . R.L.Greene.....	7
LS005...Forth-83 Evolution	Byte 8 '84 . C.K.McCabe..	10
LS006...polyFORTH/pcFORTH		
.....Review	Byte 11 '84 . E.Tello	9
LS007...Roboter programmiert seine Bahnkurven		
..... selbst.....	Elektronik 22 '84 . H.Weidner.....	4
LS008...Forth Computer	Wireless-World '83 . B.Woodroffe .	26
LS009...Forth Language.....	Wireless-World '83 . B.Woodroffe ..	8
LS010...Silizium-Software		
.....F65F11	Elek-Indust. '85 . C.Streicher.....	2
LS011...FORTH	Elektronik-Industrie '85 . G.Meyer.....	5

BES#...TTITEL.....SYSTEM..AUTOR....SEIT.

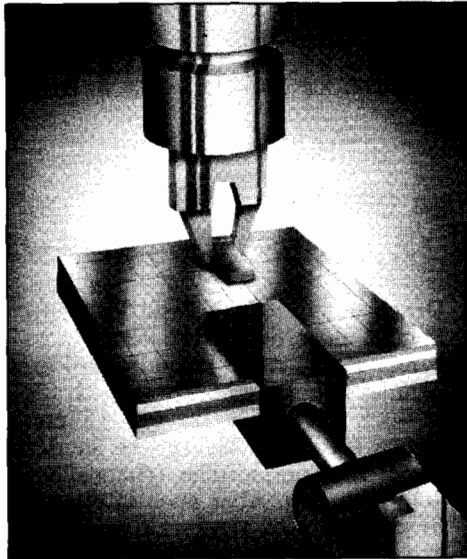
FK001...Fast Array Indexing.....	6502.. C.Springer.....	4
FK002...Forward Referencing.....	MVP.. R.Koluek.....	5
FK003...Spooler/Multitasker.....	FIG 8080.. FTS.....	1
FK004...DIGIT Contest.....	FIG.. Or. County.....	1
FK005...Sorting WORDS.....	F83.. W.Baden.....	10
FK006...VIC-20 Terminal Emulation	FIG.. D.E.Legan.....	2
FK007...CP/M in Forth.....	Forth-83.. K.Schleisiek.....	5
FK008...GUARD control structures	FIG.. C.Springer.....	12
FK009...FMODEM	FIG.. Z.Thomas.....	9
FK010...Modem Program	F83.. D.Dillon.....	3
FK011...MODEM7.....	FIG.. E.Ramm.....	5
FK012...XMODEM Protocol	PC/FORTH.. R.Taylor.....	9
FK013...CASE.....	polyFORTH.. W.Baden.....	3
FK014...nützliche Utilities.....	FIG / Forth-83.. W.Baden.....	4
FK015...Number I/O	FIG.. T.Almy.....	1
FK016...UPPER case/Parameters	FIG.. D.Doudna.....	1
FK017...Ascii Printer Graphic.....	FIG.. W.Baden.....	2
FK018...Turtle Graphic.....	FIG.. J.W.Brown.....	8
FK019...Converting FIG - Forth-83	FIG.. R.Duncan.....	7
FK020...Disassembler 6502	volks4th.. G.Rehfeld.....	2
FK021...Cordic	F83.. A.T.Furman.....	2
FK022...64 Bit Arithmetic	Forth-83.. D.A.Beers.....	5
FK023...Floating Point	Forth-79.. M.Jesch.....	4
FK024...C.Moore's BASIC Compiler	FIG.. M.Perry.....	4
FK025...8080 Assembler	FIG.. J.J.Cassady.....	2
FK026...High Level Interrupts.....	R.L.Keck.....	2
FK027...Stringstack.....	volks4th.. K.Schleisiek.....	5
FK028...Forth Inc. Line Editor.....	FIG.. S.H.Daniel.....	9
FK029...Forth Database Design.....	G.B.Haydon	8
FK030...ISAM	F83.. M.C.Stolowitz..	4
FK031...Indexer for Data Base Model	R.N.Watkins....	7
FK032...Userstack.....	8080.. P.H.Helmerts....	3
FK033...Stack Diagram Utility.....	FIG.. B.A.Cole.....	10
FK034...New Syntax for Def. Words.....	FIG.. B.Ragsdale.....	8
FK035...Forth Gedichtgenerator.....	FIG.. B.Ragsdale.....	1
FK036...Quick Text		
.....Formatter QTF	Forth-79.. L.Brodie.....	12
FK037...QTF Extensions	Forth-79.. R.Koluek.....	5
FK038...Database Field Definitions	Forth-79.. E.W.Fittery.....	5
FK039...Quicksort.....	FIG.. W.Baden.....	2
FK040...SELECT ORDER PERFORM.....	Forth-79.. W.Baden.....	4
FK041...68000 Assembler	F83.. M.A.Perry.....	10
FK042... + LOAD + THRU		
Shadow -Screens	Forth-83.. K.Schleisiek.....	1
FK043...ZEN Floating Point.....	F83.. M.Tracy.....	1
FK044...Forth Slide Rule (Float.).....	Forth-79.. N.Grossman.....	7
FK045...Floating Point Standard Words.....	M.Tracy.....	13
FK046...High Level Adress Interpreter	F83.. L.Craymer.....	6
FK047...Names for record fields	Forth-83.. W.Carpenter....	7
FK048...Report of Forth Non-Stand. Team	C.Curley.....	10
FK049...ENCLOSE MATCH for 6301	FIG.....	2
FK050...Nonce Defining Words	F83.. W.Baden.....	6
FK051...8087 Floating Point.....	F83.. S.Pollack.....	17
FK052...Simple Metacompile	FIG.. G.M.Kelly.....	9
FK053...Naming Conventions	K.Harris.....	4
FK054...Ackermann Funktion	FIG.. U.Hoffmann....	1
FK055...Simple Expert System.....	FIG.. J.J.Cassady.....	9

Inhaltsverzeichnisse von FORTH-Publikationen

BES#...PUBLIKATION.....SEITEN

I01.....Journal of Forth Application and Research.....	15
I02.....Rochester Conference Proceedings	20
I03.....Forth Dimensions, alle Ausgaben.....	50

EINIGE ROBOTER KÖNNEN NICHT GLEICHZEITIG GEHEN UND ZUGREIFEN...



...WAHRSCHEINLICH HABEN SIE NOCH NIE ETWAS VON FORTH GEHÖRT.

polyFORTH II® ist die vierte Generation professioneller FORTH-Systeme und wurde speziell für Echtzeitanwendungen entwickelt. Es ist im Multiuser-Betrieb außergewöhnlich leistungsfähig, und eine beliebige Anzahl asynchroner Prozesse können konkurrierend ablaufen.

polyFORTH II hält für den Software-Entwickler alle Werkzeuge bereit, um selbst die präzisesten und kniffligsten Roboterleistungen zu programmieren. Weil polyFORTH II maximale Flexibilität und Erweiterbarkeit in einem kompakten Paket vereinigt, hat es neue Maßstäbe für die an Robotersoftware zu stellenden Anforderungen gesetzt.

Die Wahl fällt auf polyFORTH II, wenn es um die Programmierung von Steuerungssystemen für Roboter und Frontend-Rechner geht; auch für alle anderen Echtzeitanwendungen wurde polyFORTH II geschaffen: Meß- und Prüfsysteme, Datenerfassung und -analyse, Bildverarbeitung, Fertigungssteuerung und Automatisierung, u. a. m.

Das benutzerfreundliche polyFORTH II® gibt es als Hochleistungs-Native-System für

IBM-PC, -XT und -AT, Intel 8086/8088 SBC's, Motorola 68000, RCA 1802 und 1805 und für DEC PDP-11 und LSI-11.

Auch ist es als betriebssystem-residentes und -kompatibles System für MS-DOS, RSX/VMS, CP/M-80 und CP/M-86 erhältlich.

polyFORTH II® ist eine vollständige, voll integrierte Programmierumgebung und alle Funktionen, die gemeinhin durch separate Programme repräsentiert werden, sind gleichzeitig systemresident: Betriebssystem, Editor, Assembler, Compiler, Debugger, Dienstprogramme und Interpreter. Ein Target Compiler ist für verschiedenste Zielprozessoren verfügbar.

Sprechen Sie noch heute mit uns . . . dann tanzt Ihr Roboter vielleicht schon morgen wie Baryshnikov.

Alleinvertretung der FORTH, Inc., USA, in Deutschland, Österreich und Schweiz:



RSO Gesellschaft für technische Kybernetik
Am Moosfeld 85 · 8000 München 82
Tel.: 089-429188, Telex 5212678 rso d

FORTH, polyFORTH und Target Compiler sind eingetragene Warenzeichen der FORTH, Inc. U.S.A.

Wie das Titelbild entstand:

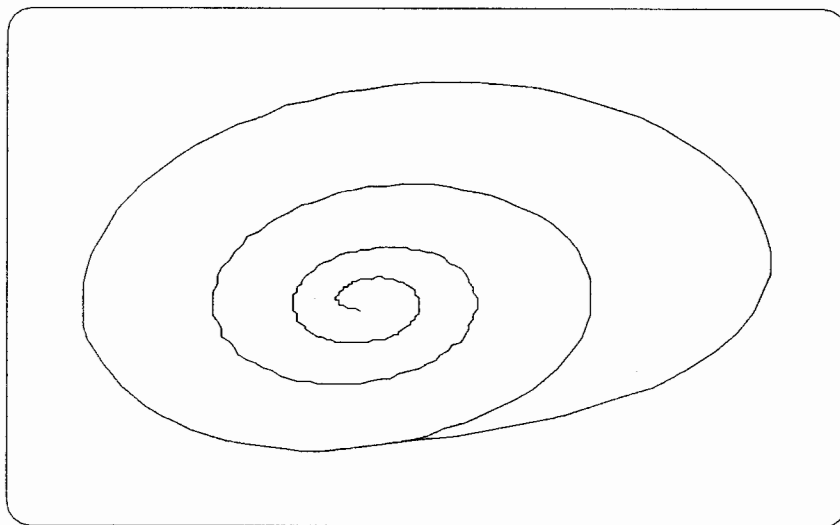
von Christoph Krinninger

Das nachstehende Programm berechnet den sogenannten Rössler-Attraktor. An diesem Beispiel läßt sich auch die ideale Verschmelzung von Floating-Point Worten und Infix-Notation zeigen.

An diesem Attraktor kann man sehr schön die Eigenschaften 'chaotischer' Attraktoren untersuchen. Ein Punkt im dreidimensionalen Raum wird iteriert und bewegt sich dabei auf einer 'Trajektorie'. Die Projektion dieser Trajektorie auf die x-y-Ebene wird auf dem Bildschirm dargestellt. Nach einiger Zeit schwingt sich das System ein und ergibt z.B. für $M=2,6$ eine einfache periodische Schwingung (Die Trajektorie der Einschwingphase kann man mit

Druck auf die CLR-Taste löschen). Erhöht man den Wert von M , so kann man eine Periodenverdopplung beobachten: Periode 2 für $M=3,5$, Periode 4 für $M=4,1$ etc. Werte für $M > 4,23$ ergeben in der Regel nichtperiodische (chaotische) Schwingungen. Aber auch in diesem Bereich gibt es vereinzelt periodische Schwingungen, wie z.B. $M=5,4$ mit einer Periode 3. Das Programm kann mit Druck auf die UNDO-Taste beendet werden.

In diesem Programm wurde das ZEN-Floating-Point-Paket von Martin Tracy verwendet, das zwar sehr einfach ist, aber problemlos in 2 Screens implementierbar ist. Ferner habe ich für die relativ komplexe Berechnungsformel den Infix-Parser



Beispiel 1

Stichworte:

- » Rössler-Attraktor,
- » Floating-Point,
- » Infix-Notation

von Dick Pountain verwendet, der die Lesbarkeit von mathematischen Formeln stark verbessert, ohne einen Laufzeit-Overhead zu erzeugen.

Das Wort INFIX(wandelt während der Compile-Zeit den Ausdruck zwischen den Klammern in die FORTH-übliche Postfix-Notation um. Man kann diesen Infix-Parser beliebig erweitern, beispielsweise ist der Ausdruck "Y fnegate" nicht sehr schön, gemeint ist eigentlich "-Y". Wie im Sourcecode erklärt, ist Y eine Spezialvariable ähnlich dem "TO-Konzept", die bei ihrem Aufruf ihren Wert auf dem Stack hinterläßt, wie dies z.B. auch eine Konstante macht. Den Wert der Variablen kann man mit dem Wort "-" ändern, z.B. "3 - fred". Die Laufzeit dieses Speicherwortes ist geringfügig größer als z.B. bei "!" oder "2!", dieser wird aber durch den Wegfall von Worten wie "@" oder "2@" und der besseren Lesbarkeit von Formeln kompensiert. Eine Formel wie

$$X_{neu} = X + (-Y-Z) * Dt$$

lautet dann in FORTH

```
infix( ( Y fnegate f- Z ) f* Dt
f+ X ) - Xneu.
```

Fnegate, f* und f+ sind die Floating-Point Versionen der bekannten FORTH-Worte. Die Infix-Version entspricht weitestgehend der Original-Formel. Wie man am obigen Beispiel sieht, kann der Infix-Parser noch nicht die Regel "Punkt-vor-Strich" verarbeiten, so daß die Formel geringfügig umgestellt werden mußte. Der Infix-Parser liefert folgenden FORTH-Code:

```
Y fnegate Z f- Dt f*
X f+ (-Xneu)
```

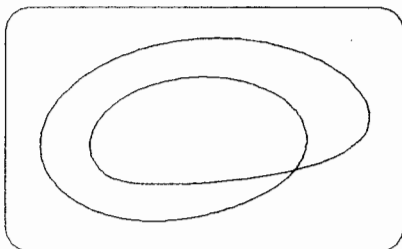
Man sieht also, daß man die bessere Lesbarkeit von Infix-Notation bei Formeln nicht durch schlechtere Laufzeiten erkaufen muß.

Rössler-Attraktor

Das Programm ist in volksFORTH 3.80 für den Atari ST geschrieben. Das Wort PLINE zeichnet eine (Poly-)Linie. Dieses Wort aus der VDI-Bibliothek wurde den schnelleren Line-A Grafikbefehlen vorgezogen, da zum Ausdruck ein GEM kompatibles Metafile benötigt wurde, das mit dieser Methode problemlos herstellbar ist. Das Wort PLINE nimmt sich n-mal x/y-Paare vom Stack. Bei diesem Programm wird aber keine Polylinie erzeugt, sondern nur eine einfache Gerade. PLINE kann also bei anderen FORTH-Versionen auf einfache Weise ersetzt werden. (PLINE (x y ... xn yn n --))

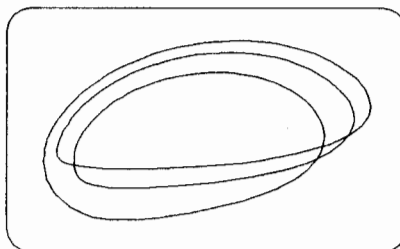
Literaturangaben

1. Changing Reverse Polish to Infix, Dick Pountain
BYTE 01/88,
S.285 ff



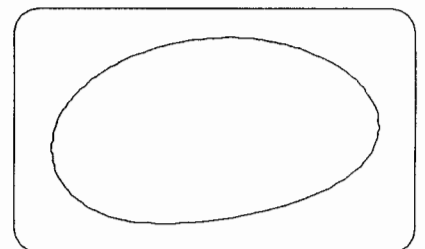
Beispiel 3

2. Zen Floating Point, Martin Tracy,
Dr. Dobb's Toolbook
of FORTH, Vol. 2,
S. 261 ff



Beispiel 4

3. Rössler-Attraktor, Marc Schwehm,
68er, 08/88
(Zeitschrift der
AGS e.V.)



Beispiel 5

Screen # 0

Screen # 13

\ 02dez88 ck

Das nachstehende Programm berechnet den sogenannten Rössler-

Attraktor. An diesem Attraktor kann man sehr schön die Eigenschaften 'chaotischer' Attraktoren untersuchen.

Ein Punkt im dreidimensionalen Raum wird iteriert und bewegt sich auf einer 'Trajektorie'. Die Projektion dieser Trajektorie wird auf dem Bildschirm dargestellt.

Nach einiger Zeit schwingt sich das System ein und ergibt für verschiedene Werte von M einfache, periodische oder nicht-periodische (chaotische) Schwingungen.

Rössler-Attraktor

Screen # 1

```
\ Loadscreen                03dez88 ck
gem \needs pline    include vdi.scr
Onlyforth    gem also
decimal
\needs float 2 8 thru
\needs it      : it ;
forget it     : it ;
9 12 thru
```

Screen # 2

```
\ ZEN Floating Point        02dez88 ck
| : d2* ( d1 - d2 ) 2dup d + ;
| : d10* ( d1 - d2 ) d2* 2dup d2* d2* d + ;
| : tuck ( n1 n2 - n2 n1 n2 ) swap over ;
| : d+- ( udn n - dn ) 0 IF dnegate THEN ;
| : trim ( dn n - f ) r tuck dabs
BEGIN
over 0 over or
WHILE 0 10 um/mod r 10 um/mod nip r r 1 + r
REPEAT
rot d +- drop r ;
```

Screen # 3

```
\ F+, F-, FNEGATE,        02dez88 ck
' extend Alias sd    ( n - d )
: f+ ( f1 f2 - f3 )
rot 2dup - dup 0
IF negate rot r nip r swap r ELSE swap r nip THEN
r sd r dup 0
?DO r d10* r 1- over abs 6553 IF leave THEN LOOP
r over + r IF rot drop ELSE rot sd d+ THEN r trim ;
: fnegate ( f1 - -f1 ) r negate r ;
: f- ( f1 f2 - f3 )    fnegate f + ;
```

Screen # 14

```
VDI-Packet laden
Minimales Floating-Point Packet und Infix-Parser laden
Floating-Point von: M.Tracy, Dr. Dobbs Toolbook of FORTH Vol.2
Infix-Parser von: D. Pountain, BYTE 01/88
Kleiner Trick, um das Programm wiederholt laden zu können
FORGET IT löscht alle Worte des eigentlichen Programmes
Eigentliches Programm laden
```

Screen # 15

```
D2*      32-bit Integer verdoppeln
D10*     32-bit Integer mal 10
TUCK     Top-of-stack unter Next-of-stack
D+-      32-bit Integer erhält Vorzeichen von n
TRIM     32-bit Integer in vernünftige FLOAT-Zahl wandeln
```

Screen # 16

```
SD       16-bit Integer in 32-bit Integer wandeln
F+       Floating-Point Addition
FNEGATE  Vorzeichen umdrehen
F-       Floating-Point Subtraktion
```

Rössler-Attraktor

Screen # 4

```
\ F*, F/, FLOAT          02dez88 ck

: f*  rot + r
2dup xor r abs swap abs um* r d +- r trim ;

: f/  over 0 = Abort" D/"
rot swap - r 2dup xor -rot abs dup 6553 min rot abs 0
BEGIN 2dup d10* nip 3 pick WHILE d10* r 1- r REPEAT
2swap drop um/mod nip 0 rot d + r trim ;

: float ( n - f )
dpl @ negate trim ;
```

Screen # 5

```
\ F., FN                  02dez88 ck

: f.  ( f - )      r dup abs 0
# r@ 0 max 0 ?DO Ascii 0 hold LOOP
r@ 0
IF r@ negate 0 max 0 ?DO # LOOP Ascii . hold THEN
rdrop #s rot sign # type space ;

| : do-10*      ( n1 mal -- n2 )  0 ?DO 10 * LOOP ;
| : do-10/      ( n1 mal -- n2 )  abs 0 ?DO 10 / LOOP ;

: fn  ( f - n )
0.5 float f +
dup 0 IF do-10/ ELSE do-10* THEN ;
```

Screen # 6

```
\                          03dez88 ck

: fvar: ( - ) Create 0. , ,
( - f )      Does 2@ ;

: (- ( f - )      r dup 2+ r @ 2! ;

: - ( - ) ' body compile ( - ; immediate restrict

Variable forward

: infix(      ( - ) forward @ execute ;

: next-symbol ( -- adr char ) bl word dup 1+ c@ ;
```

Screen # 17

F* Floating-Point Multiplikation

F/ Floating-Point Division

FLOAT Umwandlung einer 32-bit Integer in eine Floating-Point Zahl

Screen # 18

F. Floating-Point Zahl ausdrucken

DO-10* 16-bit Integer nach links verschieben

DO-10/ 16-bit Integer nach rechts verschieben

FN Floating-Point Zahl in Integer umwandeln

Screen # 19

FVAR Eine modifizierte VARIABLE für Floating Point
Hinterläßt bei ihrem Aufruf ihren Inhalt auf dem Stack, ähnlich wie eine Konstante.
(Die Runtime-Routine für -

- Speichert einen Wert in die FVAR Variable
z.B. 3 - test
FORWARD Enthält die Adresse für Vorwärts-Referenzen

INFIX(Dummy-Definition, führt nur den Inhalt von FORWARD aus
NEXT-SYMBOL Holt das nächste Wort aus dem Source-Code und hinterläßt zusätzlich den ersten Buchstaben

Rössler-Attraktor

Screen # 7

```
\
                                03dez88 ck

Defer compile-term

: next-term  ( - flag )
next-symbol dup
Ascii ) = IF 2drop      true exit THEN
Ascii ( = IF drop infix( false exit THEN
count pad place pad capitalize bl pad count + c! find
IF
compile-term
ELSE
number drop [compile] Literal
THEN
false ;
```

Screen # 8

```
\
                                03dez88 ck

: (compile-term ( cfa - )
dup
[ ] f+ case? IF next-term drop , exit THEN
[ ] f- case? IF next-term drop , exit THEN
[ ] f* case? IF next-term drop , exit THEN
[ ] f/ case? IF next-term drop , exit THEN
drop , ;

' (compile-term is compile-term

: infix(      ( - )
BEGIN next-term UNTIL ; immediate restrict

' infix( forward !
```

Screen # 9

```
\ Einige Konstanten und Variablen          02dez88 ck

3.5 float 2Constant M
0.01 float 2Constant DT
0.2 float 2Constant N

fVar: X      fVar: Xneu
fVar: Y      fVar: Yneu
fVar: Z      fVar: Zneu

Variable last-x
Variable last-y

$4700 Constant cls          $6100 Constant undo
```

Screen # 20

COMPILE-TERM Vorwärts-Referenz, um NEXT-TERM rekursiv machen zu können.
NEXT-TERM Verarbeitet eigentlichen INFIX-Term s.a. BYTE 01/88, S. 285
"Changing Reverse Polish to Infix"
von Dick Pountain

Screen # 21

(COMPILE-TERM Liste der Worte, die jetzt auch in INFIX-Notation verwendet werden können.

INFIX(Hauptschleife des INFIX-Parsers

Screen # 22

M DT N Mit diesen Konstanten kann man die Eigenschaften des Rössler-Attraktor verändern

X Y Z Räumliche Koordinaten
Xneu Yneu Zneu Hilfsvariablen

LAST-X LAST-Y Bildschirm-Koordinaten des letzten Punktes, wird für die Darstellung von Koordinaten gebraucht.

CLS UNDO Tastatur Scan-Codes

Rössler-Attraktor

Screen # 10

\ Eigentliche Rössler-Berechnung 03dez88 ck

```
: calculate
infix( ( Y fnegate f- Z ) f* Dt f+ X ) - Xneu
infix( ( N f* Y f+ X ) f* Dt f+ Y ) - Yneu
infix( ( ( X f- M ) f* Z f+ N ) f* Dt f+ Z ) - Zneu
Xneu - X Yneu - Y Zneu - Z;
```

```
\\
: calculate (-)
Y 2@ fnegate Z 2@ f- DT f* X 2@ f+ r r
X 2@ Y 2@ N f* f+ DT f* Y 2@ f+ r r
X 2@ M f- Z 2@ f* N f+ DT f* Z 2@ f+
Z 2! r r Y 2! r r X 2!;
```

Screen # 23

CALCULATE Eigentliche Rechen-Routine
Xneu = X + (-Y-Z)*Dt
Yneu = Y + (X+0.2*y)*Dt
Zneu = Z + (0.2+Z*(X-M))*Dt
X=Xneu Y=Yneu Z=Zneu

Dieselbe Routine in klassischer Postfix-Notation

Screen # 11

\ Tastatur Dekodierung 03dez88 ck

```
: init-rössler (-)
page
0. - X 0. - Y 1. float - Z
255 last-x ! 235 last-y !
overwrite solid ( Einstellung des Linien-Stiftes );

: rdecode ( flag c - flag )
cls case? IF page exit THEN
undo case? IF drop true exit THEN
2drop true ;
```

INIT-RÖSSLER Initialisierung

RDECODE Tastatur-Bearbeitung
Noch viel Platz für Erweiterungen

Screen # 12

\ Hauptroutine 02dez88 ck

```
: rössler (-)
init-rössler
BEGIN
false ( Flag für UNTIL )
calculate
last-x @ last-y @
X [ 19. float swap ] Literal Literal
f* 250. f+ fn
Y [ 19. float swap ] Literal Literal
f* 230. f+ fn
2dup last-y ! last-x ! 2 pline
key? IF key rdecode THEN
UNTIL ;
```

RÖSSLER Die eigentliche Hauptroutine

Umwandlung in geeignete Bildschirm Koordinaten

PLINE (Polyline) zeichnet eine (viele) Geraden
PLINE (x1 y1 ... xn yn n -)

SIG FORTH



the Association for Computing Machinery's Special Interest Group on Forth

Be Part of the Future of Forth...

SIGForth is the ACM special interest group whose members are the programmers, managers, scientists, engineers, and educators that are interested in applying Forth to solve hypothetical and real-world problems.

Become a **SIGForth** member. Get the **SIGForth** quarterly newsletter and receive a 25% discount on publications and conference registration with your membership. Stay up to date on the latest developments in Forth hardware and software. Your dues also help sponsor several projects including: promoting Forth education in our universities and colleges, an annual Forth industry survey, an ANS Forth X3J14 representative and "Forthics," the creation of Forth programming "ethics" and metrics to foster more successful Forth projects. You'll read about the results first only in the **SIGForth** newsletter.

Join SIGForth Now!

Send your name, mailing address and ACM number (if applicable) with payment by check (payable to ACM), money order or credit card to the address below. **SIGForth** membership fees are: Non-ACM members \$42. ACM Student members \$11. ACM members \$20. Library subscriptions \$33. Foreign air shipment add: \$6 (partial), \$8 (full).

For additional information contact:

ACM, 11 West 42nd St, New York, NY 10036 • (212) 869-7440

ESY - Die Wissensbasis für ein Expertensystem.

**Autor: Ekkehard Flögel, Aurikelstr.24,
8176 Waakirchen, Tel: 08021/8414**

ESY stellt eine Wissensbasis für ein Expertensystem dar, das nach Art einer Sammlung von Karteikarten aufgebaut ist. Ein Experte programmiert diese Wissensbasis durch Eingaben von Fragen und gibt mögliche Antworten vor. Ein Nicht-Experte kann durch Beantworten dieser Fragen versuchen, ein Problem zu lösen.

Im Gegensatz zu LISP und PROLOG läßt sich mit FORTH ein Programm erstellen, das keinerlei Programmierkenntnisse verlangt.

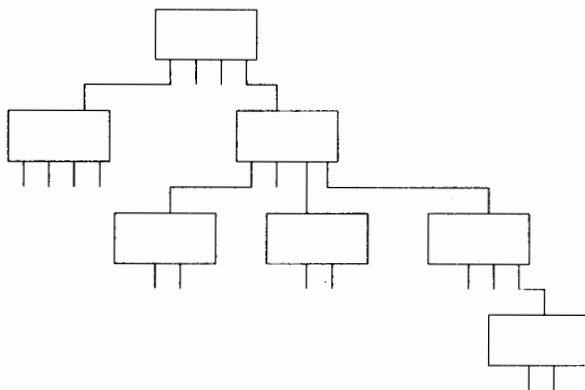


Abbildung 1: Hierarchische Baumstruktur

Blatt Nr. 7

Ist Bruch muschelrig?

Ja	Nein
----	------

Menü :ESC
Zurück :F2
Notiz :F5

Abb. 2: Kärtchen aus einer Wissensbasis

Bei einem Expertensystem soll das Wissen eines Experten durch einen Rechner erfaßt und dann anderen zugänglich gemacht werden. So ein Expertensystem besteht im allgemeinen aus einer Wissensbasis und einem Inferenzmechanismus.

Die Wissensbasis ist eine Datenbank, in welche das Wissen des Experten eingetragen wird. Diese Datenbank ist als hierarchischer Baum aufgebaut. Einen solchen Baum zeigt die Abbildung 1.

Der Inferenzmechanismus ist ein Mechanismus zum Auffinden von Schlußfolgerungen. Hier versucht das Expertensystem aus den vorhandenen Daten Schlußfolgerungen zu ziehen. Es kann eine bestimmte Strategie entwickeln und es zeigt Lernverhalten.

Stichworte:

- » Expertensystem,
- » Wissensbasis,
- » Wissensdarstellung,
- » hierarchische Bäume.

ESY - Expertensystem

Im Programm ESY wird ein Knoten dieses Entscheidungsbaumes als ein leeres Kästchen (Abb. 2), einem leeren Notizzettel, dargestellt.

Dieses Kästchen ist in zwei Felder, das "Fragefeld" und das "Antwortfeld" eingeteilt. In das Fragefeld wird die Frage, welche die nächste Entscheidung darstellt, eingetragen. Die möglichen Antworten werden in das Antwortfeld eingetragen.

In der vorliegenden Form stellt ESY nur die Wissensbasis eines Expertensystems dar. Es ist eine Datenbank, in die ein Experte sein Wissen in Form von Daten und Fragen eingibt und mögliche Antworten aufzeigt. Jede Karte stellt einen Knoten in einem hierarchischen Baum dar. Die Antworten enthalten Zeiger, die dann zu den anderen Knoten verzweigen.

So eine Wissensbasis wird hauptsächlich bei der Erkennung von Fehlern in komplexen Systemen verwendet. In den USA wurden in dieser Form zwei Expertensysteme aufgestellt. Mit dem einen werden Fehler in Diesellokomotiven, mit dem anderen Ursachen der Wasserverschmutzung untersucht.

Bei einem Störfall befragt der Operator die Wissensbasis und versucht durch Beantworten von Fragen die Ursache der Störung einzukreisen.

Ist in der Wissensbasis keine Lösung vorhanden, so muß sie, nach der Beseitigung der Störung durch den Experten, erweitert werden.

Dies ist in ESY leicht möglich, da die Einträge in die Karten und die zugehörigen Antworten jederzeit geändert werden können.

Ein anderes Anwendungsgebiet ist das programmierte Lernen. Hier werden dem Lernenden gezielte Fragen gestellt, von deren Beantwortung das weitere Lernverhalten abhängt.

Das Programm ESY ist in der Programmiersprache FORTH geschrieben. Im Gegensatz zu den Sprachen LISP und PROLOG läßt sich damit ein Programm erstellen, das in der Bedienung sowohl vom Experten, als auch vom Benutzer keinerlei Programmierkenntnisse verlangt.

In der augenblicklichen Version können in eine Karte ein Text von 192 Zeichen und vier Antworten mit jeweils 7 Zeichen gespeichert werden. Die Form der Karte und die Zahl der Antworten kann bei der Programmerstellung vorgegeben werden.

Jeder Karte, d.h. jedem Knoten des Baumes ist eine zweite Karte zugeordnet, in die sowohl der Experte, als auch der Benutzer Zusatzanmerkungen eingeben kann.

Ist eine Lösung gefunden worden, so kann der Weg, der zu dieser Lösung geführt hat, ausgedruckt werden.

Die Programmierung eines Inferenzmechanismus ist weitgehend von der Art des Expertensystems abhängig. Die Verwendung der

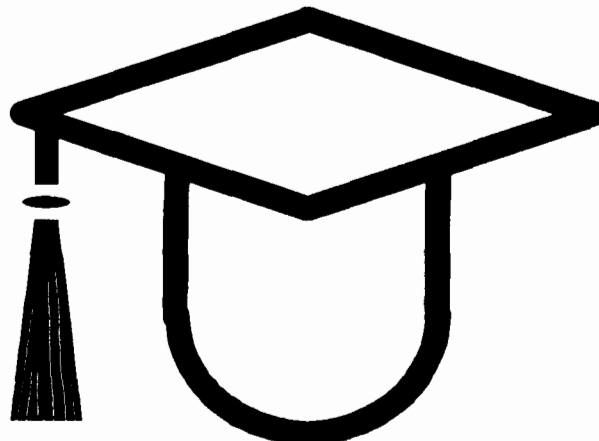
Sprache FORTH macht es leicht, nachträglich Such- und Lernstrategien zu implementieren und an die spezielle Aufgabenstellung anzupassen. Auch die beiden oben erwähnten Expertensysteme wurden zuerst in LISP getestet und dann in FORTH programmiert.

Das Programm ESY ist LAN-fähig, d.h. man kann von einem PC über Kabel auf einen Zentral-PC zugreifen. Dieser Zugriff kann auch von mehreren Stellen gleichzeitig erfolgen.

Zur Zeit wird an einer Erweiterung von ESY zur automatischen Fehlererkennung in Prozessen gearbeitet. Dazu wird die Möglichkeit der Sprache FORTH genutzt, mit Worten zu programmieren. Eine bestimmte Messung wird durch den Eintrag in die Karte ausgelöst. Das Meßergebnis wird mit den als Antwort vorgegebenen Meßwerten verglichen. Der Entscheidungsbaum wird solange durchlaufen, bis eine Lösung gefunden wird.

Quellenangaben:

JON83 Expertensystem for dieselectric locomotiv repair, The Journal of FORTH-Application and Research Vol I, Issue I, Page 7



MARC4 - ein applikationsorientierter, FORTH-programmierbarer Mikrocontroller

G. Göttle, R.W. Kobenter, EUROSIL electronic GmbH

Warum MARC4 ?

Unter Berücksichtigung der Anforderungen des Marktes für batteriebetriebene tragbare Applikationen hat die Firma EUROSIL electronic, als Marktführer in 1,5 Volt CMOS-Technologie das MARC4 Konzept entwickelt.

Der MARC4 Mikroprozessor-Kern besitzt eine 4-Bit stack-orientierte Harvard Architektur. Die Vorteile einer Stack- bzw. Null-Adreß-Maschine liegen u.a. in einem kompakten Befehlssatz, durch die implizite Adressierung der obersten Stackelemente, was eine direkte Programmierung des Prozessors in FORTH unterstützt. Durch zusätzliche Implementierung von FORTH-spezifischen Befehlen wie ROT, SWAP, OVER, DO und I ergibt sich selbst in kompilierter Form nur ein geringer Overhead im Vergleich mit Assembler-programmierten Mikrocontroller-Applikationen.

Ein weiterer Vorteil ist die geringe Chipfläche für den Prozessorkern, im Gegensatz zur Register-Architektur (z.B. NEC 75xx, Z80) und die sehr einfache Schnittstelle zur On-Chip Peripherie (über TOS). All dies ergibt ein sehr gutes Verhältnis zwischen On-Chip Peripherie-Modulen und dem eigentlichen

CPU-Kern. Die Leistungsfähigkeit der MARC4 CPU bei 1,5 Volt ist vergleichbar mit anderen 5 Volt 8-Bit Mikrocontrollern wie Rockwell 65F11, 80C48, 68HC04 und NEC 78C0x.

Application Specific Standard Configurations

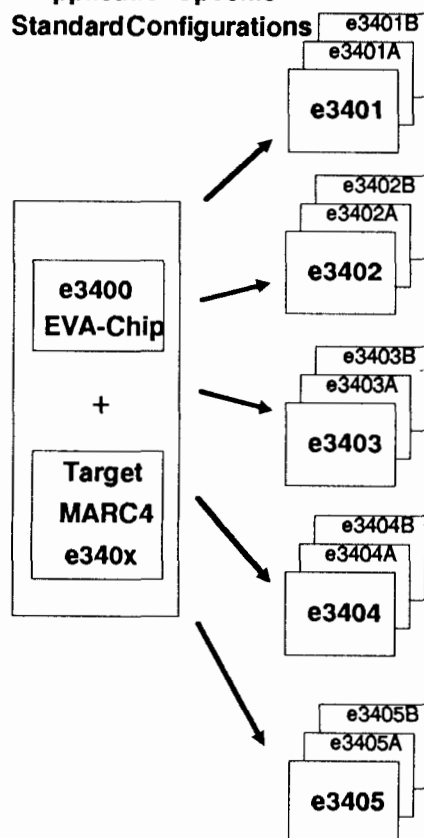


Bild 1: Die MARC4 Mikrocontroller Fam.

Das Applikations- umfeld

Wie bereits erwähnt, liegt der Zielmarkt in den Bereichen wo extreme Anforderungen an geringen Stromverbrauch über eine lange Zeit gestellt werden. Solche Anwendungen finden wir zum Beispiel bei

- **Haushaltsgeräten**
(Digitale Batterie-Ladesteuerung, LCD-Küchenwaage, Kaffeemaschine)
- **Zeitsteuerungen**
(Schaltuhr, Rolladensteuerung, Klimaanlage)
- **Datenerfassung**
(Intelligenter Sensor, Betriebsstundenzähler, Wetterstation, Altimeter, Variometer, Meßuhren, Heizkostenverteiler, solarzellenbetriebene Umwelt-Meßdatenerfassung)
- **Consumer-Bereich**
(Weltzeituhr + Taschenrechner, Fahrrad-Computer, Funkuhr, I2C-Bus Controller, IR-Remote Control)
- **Telecom**
(Pager, Feature-Telefon, Tragbares Telefon)
- **Identification/Entry Systeme**
(Core mit On-Chip EEPROM)

Große Teile dieser Anwendungsgebiete werden durch die EUROSIL Standard Mikrocontroller Familie bereits abgedeckt.

MARC4

Tabelle 1: MARC4 Stand. Konfigurationen

part number	features	Memory		I/O Lines			Interrupts	Prescaler/ Watch-Timer	8 Bit Timer/ Counter	LCD
		ROM	RAM	I/P	O/P	Bi				
e3400EVA		(ROM less)	252 x 4 Bit	8 lines			5external	2 kHz ÷ 1 Hz INT 4	---	---
e3401	II/89	4 k Byte	252 x 4 Bit	24 lines			2external 1Timer/Counter 1Prescaler	256 Hz ÷ 1 Hz INT 3	---	---
e3402		4 k Byte	252 x 4 Bit	4	12 lines		2external 1Prescaler	2 kHz ÷ 1 Hz INT 5	---	up to 64 segments static, 2:1, 3:1, 4:1
e3403	II/89	4 k Byte	252 x 4 Bit + 96 x 4 Bit (*)		4 (***)	12	2external 2Timer/Counter 1Prescaler	256 Hz ÷ 1 Hz INT 3	2 T/C 2 PWM (**)	---

Notes: *) I/O mapped **) for D/A Conversion ***) high current outputs

Weitere Standard-Konfigurationen werden nach der gründlichen Analyse eines Anwendungsgebietes definiert, womit viele Kunden ein bestimmtes Design kostengünstig einsetzen können. Für Anwendungen, die mehrere Peripherie-Bauelemente erfordern, bietet EUROSIL electronic eine Anzahl von MARC4-kompatiblen, externen Bausteinen an.

Der Schritt zur Realisierung mit dem MARC4

qFORTH ist eine Implementation der FORTH-Syntax in kompilierter Form für die MARC4 Prozessor-Familie. Damit ist der MARC4 der einzige Mikrocontroller in dieser Preis-/Leistungsklasse, der effektiv in einer höheren Sprache programmiert wird.

MARC4 Peripherals/Support Devices
EUROSIL electronic bietet dem MARC4-Anwender an Hand einer leicht verständlichen Einführung (MARC4 Evaluation Package, inkl.

qFORTH Tutorial) eine Anleitung in die Arbeitsweise des Prozessors und dessen Programmierung.

Für die Umsetzung einer Projektspezifikation in ein MARC4 Produkt wird die Programmentwicklung wesentlich vereinfacht durch

- MARC4/qFORTH Programmer's Guide, und
- qFORTH Language Reference Dictionary.

Diese Manuals beginnen dort, wo das qFORTH Tutorial aufhört und führen den Anwender an Hand von frei verfügbaren Programm-Modulen (Toolbox-Konzept) in kürzester Zeit zur Realisierung.

e1350 (III / 89)	LCD Driver with 1 ² C Bus Interface	128 Segments Static, 2:1, 3:1, 4:1
e2850 (III / 89)	A/D Converter	8 Bit Successive Approx. 4 Bit Parallel Interface
e6111	Static RAM	256x4 Bit 4 Bit Parallel Interface
e6210	EEPROM	64 Bit Serial Interface (Di, Dout, Clk)
e7313	Static RAM with 1 ² C Bus Interface	256x8 Bit Alarm + Clock/Calendar

Tabelle 2: MARC4 Peripherie-Bausteine

Die MARC4/qFORTH Entwicklungs-Umgebung

Das MARC4-Entwicklungssystem ist ein voll-integriertes Paket von Werkzeugen, das einen

- Editor,
- File-Manager,
- qFORTH Target-Compiler,
- MARC4 Software Simulator, und den
- MARC4 Echtzeit-Emulator

beinhaltet und durch einen User's Guide unterstützt wird.

MARC4

MARC4 Emulator		UserBreak	after 636.162 µsec	EUROSIL electronic	
EXP Stack		ROM Data Disassembly			
TOS	F	01F	7C	NOP	
		020	7C	NOP	
		021	7C	NOP	
		022	1C	DECR	
		--> 023	9D	SERA	_#DOS72
		__LOOPS72	2F	DROPR	
		025	1C	DECR	
		026	9B	SERA	_#DOS71
		__LOOPS71	2F	DROPR	
		028	19	SET_BCF	
		029	9B	SERA	_BEGIN\$70
		__LOOPS70	1D	RTI	
		02B	C1	SCALL	\$RESET

Registers	RETURN Stack	Symbolic RAM Data	
PC 023	SHOWACTUAL	S0	0 F 5 1 3 3 E E E 0
CCR -Z-I	_#DOS71	37h	5 9 0 0 0 9 0
SP 2D		LCDISPLAY	0 5 9 0 F
RP 04		P_KEYCOUNT	0 0
X A1		SWITCHTIME	0 0 0 0 0 0 0 0 0 0
Y 13		4Fh	0 0 0 0 0 0 0 0 0 0

F1-Help F2-Step F3-Run F4-Reset F5-BrkPts F6-Load F7-Symbols F8-Trace F9-Edit

Bild 2: MARC4 Emulator User Interface

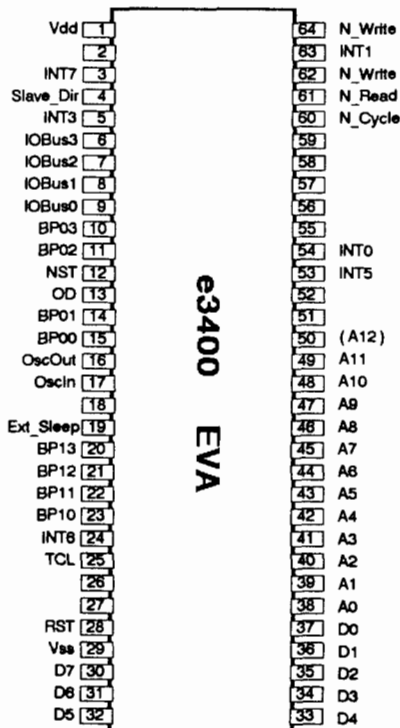


Bild 3: Pinbelegung des ROM-less MARC4

Um den Einstieg in die MARC4 Programmierung zu erleichtern, bietet die Firma EUROSIL electronic ein *Einstiegspaket* zum Preis von ca. 250,- DM an. Dieses Paket enthält auf zwei IBM-PC formatierten Disketten das komplette Software-Entwicklungssystem V1.1 mit dem dazugehörigen User's Guide.

Als ein weiteres Entwicklungswerkzeug wird Anfang 1989 die Version 2.0 des Entwicklungssystems angeboten, die zusätzlich den bereits erwähnten MARC4 Hardware Emulator (als IBM-PC Einsteckkarte) sowie den qFORTH Programmer's Guide enthält.

Mit Hilfe des im 2. Quartal 1989 verfügbaren ROM-less MARC4 EVA-Chips lassen sich sehr schnell Prototypen der Zielapplikation aufbauen, ohne das Risiko und die Kosten einer Programm-Maske.

Weitere Informationen können Sie jederzeit bei der Firma EUROSIL electronic (089/31906-275) erhalten.

Berichtigung:

In der "Vierten Dimension" Vol. IV, Nr. 2/3, September '88, S. 57/58 sind (durch die Besonderheiten des Ventura Publisher Zeichensatzes) die technischen Daten des MARC4 Mikrocontrollers der Firma EUROSIL electronic GmbH, Eching, verfälscht wiedergegeben worden.

Die CPU des MARC4 arbeitet mit einer Befehlszykluszeit von 2 Mikrosekunden bei 1.5 Volt, so daß ca. 80% der Befehle wie DUP, DROP, SWAP, OVER und alle ALU-Operationen mit dieser Taktrate ausgeführt werden. Bei einer Versorgungsspannung von 5 Volt vervierfacht sich die maximale Taktrate. Die Stromaufnahme des Prozessors liegt dabei zwischen einem (im Standby/Sleep-Mode) und 300 Mikro-Ampere bei 1.5 Volt.

Bild 4: Berichtigung

Auswertung der Umfrageaktion

von Christoph Krinninger

Die Umfrageaktion der FORTH-Gesellschaft e.V., die im September '88 gemacht wurde, hat sehr große Resonanz und Zustimmung gefunden, wie sich an der enormen Anzahl der zurückgeschickten Fragebögen zeigte. Das Direktorium möchte sich auf diesem Wege bei allen Einsendern herzlich bedanken und wird versuchen, alle Anregungen und Erkenntnisse in die Praxis umzusetzen.

Im folgenden möchte ich einige Anregungen erläutern:

1. *Es sollen Lösungen und Tools gesammelt werden und Toolboxes daraus gemacht werden.*

- Das Direktorium wird in den folgenden Ausgaben der 'Vierten Dimension' verstärkt darauf achten, daß möglichst viele Applikationen mit FORTH veröffentlicht werden. Ferner existieren in Hamburg und in München ziemlich große FORTH-Bibliotheken, die in Zukunft verstärkt den Mitgliedern zugänglich gemacht werden. Diese Unterlagen können dann entweder über den Kopierservice in Hamburg, den Diskettenservice oder aus den Mailboxen bezogen werden. Mehr dazu in dieser und der nächsten 'Vierten Dimension'.

2. *Der FORTH-Tree soll wieder zugänglich gemacht werden.*

- Die FORTH-Mailbox in Hamburg wird voraussichtlich am 1. Januar 1989 wieder am Netz

sein. Die FORTH-Mailbox in München in einem vergleichbaren Zeitraum. Der Ausbau des Mailbox-Netzes vollzieht sich sehr viel langsamer, als vom Direktorium gewünscht. Dies liegt in aller erster Linie an den begrenzten finanziellen Möglichkeiten der FORTH-Gesellschaft. Ein vollständiger Mailbox-Knoten kostet etwa DM 2.500. Der größte Teil des Jahresbudgets wird für die Herstellung der 'Vierten Dimension' benötigt. Es erfolgt hiermit also der dringende Aufruf nach Spenden an die FORTH-Gesellschaft nach dem Motto "Auch Kleinvieh macht Mist".

3. *Die amerikanische FORTH-Dimension soll auf Wunsch zugeschickt werden*

- Eine Aufgabe für den Kopierservice Hamburg

4. *Jede lokale Gruppe soll mit einem gewissen Stamm von FORTH-Bibliothek ausgestattet werden*

- Die Förderung der lokalen Gruppen liegt sehr im Interesse der FORTH-Gesellschaft. Leider ist laut Satzung keine finanzielle Förderung der lokalen Gruppen vorgesehen. Der Aufbau der Münchner FORTH-Bibliothek ist ausschließlich durch freiwillige Beiträge ermöglicht worden. Also auch hier der Aufruf nach Spenden. Das Direktorium

wird sich auch in Zukunft darum kümmern, im Rahmen der finanziellen Möglichkeiten, die lokalen Gruppen so weit wie möglich zu fördern.

5. *Es sollen mehr Informationen über Teilgebiete veröffentlicht werden, wie z.B. über FORTH-Maschinen, Künstliche Intelligenz, Objekt-orientiertes Programmieren.*

- Siehe Literaturliste in dieser und den folgenden 'Vierten Dimensionen'.

6. *Es sollen Kurse für Anfänger organisiert werden.*

- Die Förderung von Anfängern ist, wie sich etwa in der Münchener Lokalgruppe zeigt, ein ernsthaftes Problem. Ein FORTH-Kurs in der Vierten Dimension scheitert an den großen Pausen zwischen den Ausgaben. Ein Kurs auf regionaler Ebene oder über die Mailboxen ist also die einzige Möglichkeit. Es erfolgt also der Aufruf nach Vorschlägen, Initiativen, Bedürfnissen etc.

7. *Es sollen mehr Applikationen erscheinen*

- Dies wird in Zukunft der Fall sein. Wer noch Artikel in der Schublade besitzt, soll diese doch an die Redaktion schicken!

8. *Die Verwendung des GEM im volksFORTH ist sehr mangelhaft beschrieben*

- Dies ist auch meine Meinung. Da im nächsten Jahr das ANSIFORTH veröffentlicht wird, steht dem volksFORTH eine große Veränderung bevor. Zu diesem Zweck bildet sich zur Zeit eine Fachgruppe FG-FORTH, die auch speziell die Dokumentation neu erstellen wird. Wer sich für eine Mitarbeit in dieser Fachgruppe interessiert, bitte mit dem Büro in Verbindung setzen.

Umfrageaktion

9. Es besteht Bedarf an Lösungen zu folgenden Problemen

- Ein/Ausgabe von Zahlen mit mehr als 32-Bit
- Text lesen mit einem Scanner
- mathematische Funktionen für volksFORTH in Assembler
- Steuerung von (Musik-)Geräten über MIDI
- Anfängerartikel über fast alle Teilgebiete
- FORTH und Btx
- Gegenüberstellung von fig-FORTH/FORTH-83

- Da es unter den Mitgliedern also mit Sicherheit für diese Probleme Experten gibt, bitte melden.

Ferner haben sich fast alle Einsender dafür interessiert, den Sourcecode aus der 'Vierten Dimension' auf Diskette zu beziehen. Dieser wird für folgende Systeme angeboten werden: MS-DOS 5 1/4" 360KB, Atari ST 3 1/2" einseitig 360 KB, C64 etc. VC1541-Format.

Es hat sich gezeigt, daß Fach-/Lokalgruppen für folgende Bereiche fehlen:

- GEM-Programmierung,
- Kommerzielle Anwendungen
- Postleitzahlbereich 8500/8700
- Wesel/Niederrhein
- Celle/Braunschweig
- Köln
- Stuttgart
- Hannover/Braunschweig

Wer sich bereit erklärt, diese Gruppen zu leiten, möge sich mit dem Büro der FORTH-Gesellschaft in Verbindung setzen.

Die T-Shirts wurden an folgende Einsender verlost:

Jürgen Brust, Ebersbach
Claus Vogt, Berlin
Christian Schwarz, München
Wigand Gawenda, Hamburg
Uwe Oestermeier, Tübingen
Max Strobel, Hamburg
Rolf Kretzschmar, Baesweiler
Dirk Brühl, Nürnberg
Holger Blum, Hamburg
Peter Bierbach, Lüneburg

Herzlichen Glückwunsch !!

Da die Umfrageaktion so großen Erfolg zeigte, werden wir diese in regelmäßigen Abständen wiederholen. Nochmals vielen Dank für die rege Beteiligung. Für Anregungen, Kritik und Engagement wird das Direktorium auch in Zukunft ein offenes Ohr zeigen.

Real Time

Forth-Prozessoren

Aber wo gibt's Unterstützung?
Ganz einfach: Vierte Dimension!
Wort steht z.B. zu lesen:

Johannes Teich
Echtzeit-Applikationen
Hauptmann-Bauer-Weg 16
D-9118 Murnau
Telefon 08941-1489

Anzeige

FORTH-Bibliothek, Teil 1

Dies ist eine auszugsweise Übersicht der FORTH-Bibliothek der Münchner Gruppe. Bei Interesse an einem Artikel kann man sich an Christoph Krininger wenden. Diese Reihe wird in den folgenden 'Vierten Dimensionen' fortgesetzt werden. (verw. Abkürzungen: DDTof = Dr. Dobb's Toolbook of FORTH, DD Journal = Dr. Dobb's Journal)

° A FORTH-Oriented Real-Time Expert System for Sleep Staging - a FORTES Polysomnographer	Artikel über die Kombination von Realzeitmethoden mit denen von Knowledge-Based Systemen	Dana Redington	DDToF Vol. 2
° Acht Damen in FORTH	Das berühmte Problem	G. Bruziks	PASCAL, 03/88
° An Approach to Natural Language Parsing	A description of the design of an expectation-based parser is given.	Jack Park	DDToF, Vol. 2
° Directed Nets of Rule Sets and a Hybride Search Strategy	Details über ein Expertensystem in MACH2	Thomas Jost	euroFORML '87
° Implementation of an Expert System	Artikel über ein Expertensystem in MACH2	Dr. Manfred Walter	euroFORML '87
° Knowledge Representation in FORTH	Kurzer Artikel über Knowledge Representation in FORTH	Dana Redington	DDToF, Vol. 2
° The FORTH Wave in AI	Historisches und Trends	Robert B. Trelease	AI Expert, 10/87
° Trainable Neural Nets in FORTH	Implementation eines neuronalen Netzes	John D. Carpenter	euroFORML '87
° Adding Record Structures to FORTH	Records with local field names	Jörg Langowski	The Complete MacTutor, Vol. 2
° FORTH Shifts Gears	The next FORTH generation has syntax that allows object-like multiple code fields	George W. Shaw	Computer Language, 05/88 & 06/88
° LIST: A Generator for Object Oriented, Cyclic Linked Lists	Implementation eines objektorientierten Kernes	Karl-Dietrich Neubert	euroFORML '87
° METHODS, Object oriented extensions redux	Implementation eines objektorientierten Kernes	Terry Rayburn	euroFORML '87
° Multiple Inheritance Object Systems	Beschreibung von ForthTalk, einer objekt-orientierten Erweiterung	Stephen D. Lindner	DDToF, Vol. 2
° Object-oriented FORTH	Einleitender Artikel über objektor. Worte	Dick Pountain	BYTE, 8/86
° A Custom Floating Point Package	IEEE 32-bit real arithmetic	Jörg Langowski	The Complete MacTutor, Vol. 2
° A FORTH Slide Rule	Ein einf. Floating Point Packet in FORTH	Nathaniel Grossman	DDToF, Vol. 2
° A Software Floating Point Package with Sizeable Accurazy	Artikel über die Implementation eines Floating-point Paketes mit dem CORDIC-Algorithmus	Christophe Lavarenne	euroFORML '88
° Arithmetik in FORTH	Einf. Artikel über Rechnen mit FORTH	Rüdiger Birkemeyer	PASCAL, 01/88
° Changing Reverse Polish to Infix	Bessere Lesbarkeit von mathematischen Formeln mit Infix-Notation in FORTH	Dick Pountain	BYTE 01/88
° Dimensional Data Types in FORTH	UNIT's, Inches, Meters etc	Julian V. Noble	DD Journal. 01/88
° FORTH and the Fast Fourier Transform	FFT am Beispiel von Aktienanalysen	Joe Barnhart	DDToF, Vol. 2
° FORTH and the NC4016	Fibonacci Laufest	Douglas Ross	BYTE, 12/87
° FORTH Floating-point package		Alfred J. Monroe	DDToF
° FORTH Goes SANE	Floating Point Primitives für Macintosh	Jörg Langowski	Best of MacTutor, Vol. 1
° FORTH mit Fließkommaarithmetik	Artikel über Floating-point in fig-FORTH	Peter Rix	65xx Micro Mag
° FORTH-Befehle für 32-bit-Zahlen	Die wichtigsten D-Befehle		65xx Micro Mag
° Fourier transform faster than fast Fourier transform (FFT)	Continuous Fourier Transform in FORTH	Chen-Hanson Ting	SPIE Vol. 241 1980
° Multiplikation und Division im fig-FORTH	U* und U/ für 6502		65xx Micro Mag
° Series Expansion in FORTH		Wendall C. Gates	DDToF
° Signed integer division	/MOD im Detail	Robert L. Smith	DDToF
° SIN, COS und 3D	Trigonom. Funktionen in FORTH für C64	Frank Schmidt	c't, 10/85
° Solving Systems of Linear Equations	Curve Fitting in FORTH	Jörg Langowski	Best of MacTutor, Vol. 1
° Think like a User, Write like a Fox	Implementation des Wortes 'DIGITS', um eine bestimmte Anzahl von Zahlen einzugeben	Michael Ham	DDToF, Vol. 2
° Converting fig-FORTH to FORTH-83	Unterschiede zwischen den beiden FORTH-Versionen	Ray Duncan	DDToF

Anzeigen

Da auch wir nicht allein von Luft und Liebe existieren können, ist es möglich, Anzeigen in der 'Vierten Dimension' zu plazieren. Ist der Leserkreis auch nicht sehr umfangreich, so werden doch im Gegensatz zu anderen Zeitschriften nur wirklich Interessierte und Fachkundige angesprochen. Deshalb lohnt es sich auf alle Fälle eine Anzeige in der 'Vierten Dimension' aufzugeben. Über Preise und alle weiteren Modalitäten können Sie sich unter der Telefonnummer 089/6708355 bei D. LUDA Software informieren.

**Die nächste
'Vierte
Dimension'
erscheint im
März '89.**

Gesucht wird ...

... ein FORTH-EPROM von David Husband für den ZX 81.
Adr.: R.Kretschmar, Rote Gasse
7, 5112 Baesweiler, 02401/4390

Typenraddrucker zu verkaufen

Die FORTH-Gesellschaft verkauft einen Typenraddrucker Typ Microscan MS-15, 12 Zeichen/Sekunde, DIN A3 breit, Endlostraktor und Einzelblatt,

Preisliste vom 21.11.88: volksFORTH und ultraFORTH der FORTH-Gesellschaft e.V.

1. ultraFORTH 3.8 für Commodore C64, C16 und PLUS 4

Disketten (4*170K, 5 1/2")	DM 25.--
Handbuch	DM 65.--

2. volksFORTH 3.80 für ATARI ST

Disketten (3*360K, 3 1/2", einseitig)	DM 25.--
Handbuch	DM 65.--

3. volksFORTH 3.8 für CP/M2.2-Systeme

Diskette (Osborn-Format 5 1/4")	DM 15.--
(andere Diskettenformate auf Anfrage)	
Handbuch	DM 70.--

4. volksFORTH 3.81.2 für IBM-PC und Kompatible

Diskette (1*360K, 5 1/4")	DM 15.--
Handbuch	DM 70.--

Aufpreis für Verpackung und Versand:

Diskette(n)	DM 3.--
Handbuch (mit Disketten)	DM 4.50
Nachnahme	DM 1.70

Preisnachlaß für Mitglieder der FORTH-Gesellschaft e.V.:

Diskette	- DM 3.--
Handbuch	- DM 17.--

Bezugsquelle: M&K Kohl, Pestalozzistr. 69, 8905 Mering
Bankverbindung: Postgirokonto München, 2835 06-800

Mit der Ausgabe dieser Liste verlieren alle bisher angegebenen Preise für volksFORTH und ultraFORTH ihre Gültigkeit. Mitglieder der FORTH-Gesellschaft e.V. erhalten Ermäßigungen sowohl für Disketten als auch für das Handbuch. Der Rechnungsbetrag kann entweder direkt auf das Konto überwiesen, per Nachnahme eingezogen oder als Verrechnungsscheck beigelegt werden. Bitte geben Sie unbedingt die gewünschte FORTH-Version an. Kritik, Lob, Hinweise auf Fehler, Fragen, Anregungen und Programme zum volksFORTH sind ebenfalls an die oben genannte Adresse zu senden. Sie werden dann in den 'Vierten Dimensionen' oder auf den Programmdisketten veröffentlicht.

serielle und parallele Schnittstelle, Standardfarband und Handbuch.
Preis DM 450,00

Möchten Sie ...,

... daß an dieser Stelle eine Kleinanzeige von Ihnen steht? Dann senden Sie uns einfach den Text zu. Für Mitglieder der FORTH-Gesellschaft e.V. ist dies kostenlos, andere zahlen DM 5,- für 5 Zeilen und für jede weitere Zeile zusätzlich eine Mark.

FORTH auf dem Atari ST

Ein Angebot an alle, die sich für 32FORTH (Atari ST-FORTH-System) interessieren. Sie können bei uns die beiden M&T-Bücher 'Programmieren in FORTH', Atari ST und '32FORTH-Compiler', Atari ST zu einem um 20% reduzierten Preis beziehen.

**D.LUDA Software,
Staudingerstr.65,
8000 München 83,
Tel. 089/6708355**

Gruppen

Lokale FORTH-Gruppen, die sich regelmäßig treffen:

- 2000 Hamburg Karsten Roederer, Tel. 040/4104446. Treffen im TBZ (Technologie-Beratungs-Zentrum Hamburg) .
4130 Moers 1 Friederich Prinz, näheres Tel: 02841/583 98
Rhein-Ruhr Jörg Plewe, Tel: 0208/291 66, Treffen jeden 4. Freitag im Monat um 20.00 Uhr im Bahnhof Ottenbruch, Funckstraße, Wuppertal-Elberfeld
6100 Darmstadt Andreas Soeder, Tel. 06257/2744. Treffen an der VHS an einem Mittwoch in der Mitte des Monats.
8000 München Heinz Schnitter, Tel. 089/3103385 und Christoph Krinninger 089/7259382. Treffen jeden 4. Mittwoch im Monat 19 Uhr 30 im Vereinsraum 1 im Bürgerhaus Unterschleißheim am Rathausplatz (S-Bahnhaltepunkt S1 Unterschleißheim).

FORTH-Fachgruppen:

- 8000 München RTX 2000 Gruppe, Koordinator Max Diez, Treff- und Zeitpunkt wie oben bei der lokalen Münchner Gruppe.
6800 Mannheim FIS (FORTH Integriertes System) - Datenbank, Textverarbeitung, Kalkulation, Postadresse: Dr. med. Elemer Teshmar, Danziger Baumgang 97, 6800 Mannheim 31

Es möchten in ihrer Region eine Gruppe gründen:

- 7000 Stuttgart 31 Wolf-Helge Neumann, Huttenstr. 27, Tel. 0711/882638.
8500 Nürnberg 20 Thomas G. Bauer, Fichtestr. 31, Tel. 0911/538321.
5000 Köln 60 Michael Heycke, Boltensternstr.
1000 Berlin 30 Claus Vogt, Bülowstr.67, Tel.: 030/2168938
4830 Gütersloh 1 Ludwig Röver, Holzheide 145A

Eine Fachgruppe will gründen:

- 7000 Stuttgart 80 Grafik/Arithmetik, Jörg Tomes, Anweilerweg 56, Tel. 0711/7802293.
8000 München 70 Btx u. FORTH, Christian Schwarz, Lindenschmitstr.30, 8000 München 70

Hier kann man um Rat fragen:

- 02103/556 09 Jörg Staben, Dienstag und Freitag, 20.00 - 22.00 Uhr
02845/28951 Karl Schroer

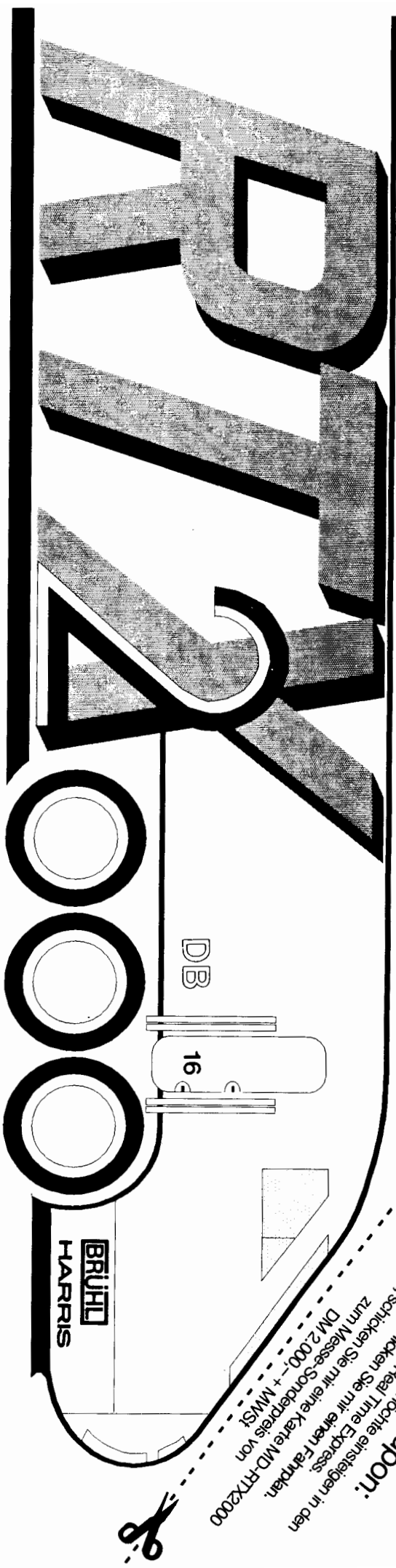


Ansprechpartner zu bestimmten Interessengebieten:

- xolksFORTH/ultraFORTH: Bernd Pennemann, Tel. 0228/640979 und Klaus Schleisiek, Tel. 040/6449412.
32-Bit Systeme: Robert Jones, Tel. 02434/4579
Forthchips,- maschinen und RISC: Roland Steck, Tel. 06151/661192
Künstliche Intelligenz: Ulrich Hoffmann, Tel. 0431/678850
NC4000 Novix Chip: Klaus Schleisiek, Tel. 040/6449412
Realtime relationale Netze: Wigand Gawenda, Tel. 040/446941
Gleitkomma-Arithmetik: Andreas Döring, Tel. 02631/52786
32FORTH Rainer Aumiller, Tel. 089/6708355

**FORTH-Gesellschaft e.V. - Postfach 1110 - D-8044 Unterschleißheim
Postgiroamt Hamburg, Kontonr.: 563211-208 BLZ 20010020**

Ergänzungen, Änderungen bitte dem Büro der FORTH-Gesellschaft e.V. mitteilen.



Die Zukunft hat schon begonnen

Warten Sie nicht auf Transkarbid oder Grande Tristesse!
 Steigen Sie jetzt ein in den Real Time Express.
 Nur mit dem Real Time Express gewinnen Sie Echt Zeit.
 Sie brauchen Ihr Fahrzeug nicht mehr mit Assembler zu tunen.
 Genießen Sie das Hochgefühl mit Hochsprache und Echt Zeit Ihr Ziel zu erreichen!

REAL TIME EXPRES und RTX ist Trademark der HARRIS CORPORATION, Palm Bay, Florida

MINIBEE und QuadCard ist Trademark der BRÜHL EE GmbH, Nürnberg



SONDERPREIS für Mitglieder
 der FORTH-Gesellschaft:

DM 1.777,--

(incl. Versand, incl. MWSt)
 Hochschulrabatt auf Anfrage!

Coupon:
 Ja, ich möchte einsteigen in den Real Time Express.
 Ja, schicken Sie mir einen Fahrplan zum Messe-Sonderpreis von DM 2.000,- + MWSt



Karten zum Einsteigen erhalten Sie ab sofort bei:
 BRÜHL ELEKTRONIK ENTWICKLUNGS-GESELLSCHAFT mbH, Hegelstraße 10, 8500 Nürnberg 10, Tel. 0911/359088
 QuadCard 100x100 mit RTX2000, 128 KB RAM, 64 KB PROM,
 40 MHz Quarztakt: DM 2.000,- + MWSt (electronica Sonderpreis incl. FG-FORTH-Compiler)

UR/FORTH

- Forth-83 Standard
- Für MS-DOS, OS/2, 80386, 68000 UNIX und XENIX
- 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

HARRIS RTX 2000

Informieren Sie sich über diesen Prozessor, der auch von uns unterstützt wird.

DSP APPLIKATIONEN

DSP Anwendungen mit dem AT&T DSP-32. Informieren Sie sich über unser Angebot.

FORTH MAIL BOX

Für alle FORTH-Interessierten hat unsere Firma eine Mailbox eröffnet. Sie ist unter der Nummer 076 67 556 zu erreichen und akzeptiert 300, 1200 und 2400 Baud, 8N1. Außer einer offenen Hauptkonferenz und einigen Fileareas enthält sie auch Supportkonferenzen für unsere komplette Produktlinie.

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	● 8096/97
● Z80	● HD64180
● 8080/8085	● 8031/32/535
● 68000	● 6303
● Z8	● 6502
● 1802	● 6802
● 6809	● 68HC11
● 65816/65802	● RTX 2000

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

Forth Programme, die mit einem LMI interaktiven Forth System z. B. PC/FORTH oder Z80 Forth geschrieben und getestet wurden, werden im Normalfall mit nur geringen Änderungen übersetzt.

Serieller ROM/RAM Simulator

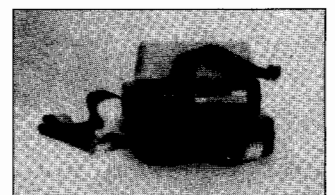
Entwickeln Sie romfähige Programme ?

Müssen Sie neu entwickelte Einplatinencomputer testen ?

Setzen Sie 2764, 27128, 27256, 27512 oder 4364, 43256 oder kompatible ROM/RAM-Bausteine ein ?

Wollen Sie diese Bausteine mit bis zu 38 400 Baud über die serielle Schnittstelle laden ?

Können Sie eine zusätzliche serielle Schnittstelle über den Speichersockel zum interaktiven Programmieren gebrauchen ?



Dann ist unser SRS63 die optimale Ergänzung Ihres Arbeitsplatzes.

Sie werden vom Preis-Leistungsverhältnis überrascht sein.

Unsere ROM-Compiler liefern direkt verwendbare Dateien, wir akzeptieren auch Intel-Hex oder Motorola-S-Formate.

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