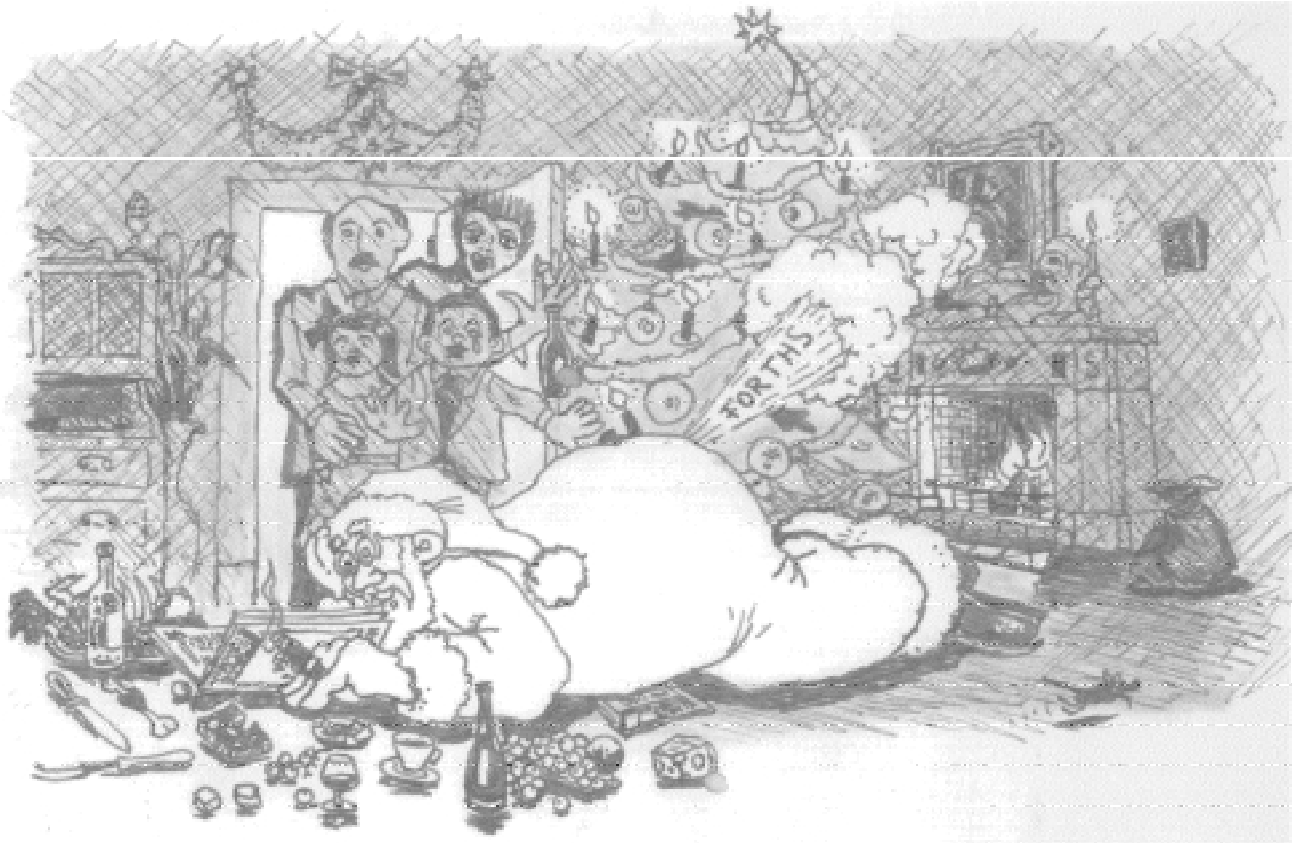


VIERTE DIMENSION

4/1995

11. Jahrgang 1995, 4. Quartal, DM 10.-



- EuroFORTH'95
- iX
- Meßtechnik mit dem PC. Teil III
- VD quo vadis?
- Das große *Weihnachtspäckchen*
- Betreff: Forth Magazin sucht Programmierer
- Das Werkzeug Win32Forth
- Parallel-Forth: Der neue Weg
- Umlaute in den Namen von Forth-Worten

Nur noch bis
15. Dezember..



Gleich bestellen!
Info auf Seite 22

FORTH MAGAZIN

Organ der Forth Gesellschaft e.V.

bigFORTH 95

Es ist da! Nach jahrelanger Ankündigung, vielen Beta-Test-Versionen und noch mehr Beta-Testern: Präsentiert mit „The Division Bell“ von Pink Floyd und allen üblichen Features, so wie es früher war: Voll 32 Bit, grüneres Gras, helleres Licht, süßerer Geschmack! Spaß beiseite, was es wirklich gibt ist:



für alle PCs ab i386 oder kompatible mit DOS

© 1995 by Bernd Paysan

bigFORTH ist ein modernes Forth-System mit Leistung und Umfang, das seinesgleichen sucht. Mit einem DOS-Extender wird der i386 erst voll mit 32 Bit ausgenutzt. *bigFORTH* erzeugt einem Code, so kompakt wie von Forth gewohnt, aber so schnell wie der mancher guter C-Compiler. Es wurde ein professionelles Entwicklungssystem geschaffen, das modernen Compilern „konventioneller“ Sprachen nicht nachsteht:

Mächtiger Compiler *bigFORTH* ist ein 32-Bit-System. Der Compiler erzeugt optimierten Native-Code für den 386er. Selbstverständlich sind auch Turnkey-Applikationen machbar, denen man Forth nicht mehr ansieht. *bigFORTH* entspricht dem vollen ANS Forth Standard.

Vielseitige Tools Multitasker; sourcefähiger Decompiler; Sourcelevel-Debugger (auch bei optimierten Makros) mit Single-Step und Trace, beliebig viele Breakpoints; Post Mortem Dump und Returnstack-Trace bei allen Fehlern; Ctrl-Pause (für den Ausstieg aus der Endlosschleife); 486-Assembler/Disassembler...

Editor Turbo-Vision-ähnlicher Editor für Screen- als auch Stream-Files.

Umfangreiche Libraries Ein umfangreiches und komfortables Fileinterface (mit ALIAS-Definitionen für die gebräuchlichen Unix-Kommandos) erleichtert die Arbeit auch auf einer großen Festplatte. DOS- und BIOS-Aufrufe, soweit vom DOS-Extender unterstützt, sind möglich. Druckertreiber sowohl zum Drucken von Listings als auch zum Protokollieren einer Sitzung; schnelle und genaue Fließkomma-Arithmetik (auf dem 387-Stack); ein "State of the Art" Memory Ma-

nagement, das auch vor komplexen Speicherstrukturen nicht kapituliert, Turbo-Vision-artige grafische Benutzeroberfläche, Object Oriented Forth und mehr...

Transparenz Sämtliche Sourcen einschließlich Kernal und Target-Compiler auf Diskette, der Forth-Wizard kann sich sämtliche Wünsche erfüllen.

Modulares Forth *bigFORTH* ist ab Version 1.10 modular. So ist separate Compilation möglich, kleine Applikationen werden schnell erzeugt. Module können dabei beliebig groß werden.

Andere Plattformen Nach wie vor auch auf allen Atari ST/TT/Falcon 030.

Dokumentation

bigFORTH wird mit einem über 350 Seiten starken Handbuch ausgeliefert. Ein Tutorial und ein Forth-Schnellkurs erlauben einen schnellen Einstieg — auch für Anfänger. Zwei ausführlich erklärte Beispiele weisen in Forth-Programmierung ein. Neben dem umfangreichen Referenzteil, in dem alle über 2000 Befehle beschrieben werden, sorgt ein Indexregister für den schnellen Zugriff.

Wo und für wieviel?

bigFORTH von **Bernd Paysan**
Stockmannstr. 14
D-81477 München
Tel. (089) 79 85 57
Und das für nur **DM 256.-**
Demoversion: **DM 10.-**

Info auch über's Internet:
paysan@informatik.tu-muenchen.de
<http://www.informatik.tu-muenchen.de/~paysan/>

FORTH - Shirt



T - Shirt: hellgrau / grün
in Größe: M-L-XL **25 DM**

Sweat-Shirt: grau / grün
in Größe: M-L-XL **40 DM**

(+ Porto)

ForthWORKS

Ulrike Schnitter
Nelkenstr. 52
85716 Unterschleißheim
Tel.: 089-310 33 85

2PW



Nur noch bis 15. Dezember..

kann man das Forth-Weihnachtspäckchen bestellen.

Die CD '*forthige Weihnacht 95*'

enthält u.a. den kompletten Ftp-Server *Taygeta* aus USA mit bald 100 Mb; den kompletten Ftp-Server *Asterix* aus Portugal mit zahlreichen Forth-Systemen und in der Sammlung von Ulrich Hoffmann befinden sich sicher auch einige ANS Quellen, um dem neuen Standard zu Leibe zu rücken.

Sofort bestellen!

IMPRESSUM

Name der Zeitschrift

FORTHMAGAZIN - VIERTE DIMENSION
Organ der Forth-Gesellschaft e. V.

Herausgeberin

FORTH-Gesellschaft e. V.
Postfach 1110
85701 Unterschleißheim
Tel./Fax: 089/3173784
Mail: secretary@admin.FORTH-eV.de

Redaktion & Layout

Claus Vogt
Ebersstr. 10
D-10827 Berlin
Tel.: 030/782 81 79 (Fax & BBS nach Bedarf)
Mail: vd@FORTH-ev.de

Anzeigenverwaltung: Ulrike Schnitter c/o Forth-
Ges.; PF 1110; 85701 Unterschleißheim.

ANS-Forth: Ulrich Hoffmann; uho@informa-
tik.uni-kiel.de; Sehestädter Str. 26;
24340 Eckernförde.

Forth international: Fred Behringer; Planegger
Str. 24; 8124 München.

Zeichnungen: Rolf Kretzschmar;
rolf.kretzsch@forth-ev.de

Titelbild: Hans-Georg Schmid

Redaktionsschluß '96

Erste Januar-, April-, Juli-, und Oktoberwoche.

Erscheinungsweise

Viermal im Jahr.

Preis

Einzelpreis: DM 10,-

Manuskripte und Rechte

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



"Ordentlich fett ..."

... sollte diese VD anlässlich des bevorstehenden Weihnachtsfestes werden. Die Autoren schlugen in die Tasten und gaben sich Mühe, mit möglichst vielen fetten Artikeln den Redaktionsschluß aufzublähen. Die Arbeitskapazität der Redaktion ist nun erschöpft, ein großes Aufatmen folgt: Eine Zeitung - oh Wunder - ward geboren und mag nun hurtig ihrer Wege gehen.

Auch inhaltlich hat uns zumindest ein Autor an unsere Grenzen geführt. Wo bisher eine halbstündige stilistische Korrektur noch jeden Satz verständlich machte (zumindest für die Korrigierenden), hatten wir diesmal eine härtere Nuß zu knacken. Ge-
knackt haben wir sie nicht, wir suchen noch nach geeigneten Nußknackern, die das für uns erledigen. Refugees nennt man das wohl im Fachzeitschriftenjargon.

Weihnachten ist auch das Fest der Liebe. Nein, sie können ruhig weiterlesen. Es geht um Schenken und Beschenktwerden, um dicke Scheine, die in zweifelhaften Gebrauchswert umgemünzt werden. Eigentlich wollte die Redaktion der Flut der Artikel nur ein Tor in den Geschenkberg bahnen. Eine 16seitige Sonderausgabe mit den jüngsten Untaten der Hippeländer hatte es sein sollen. Dann kam ein Wettbewerb dazu, der der Redaktion viel Arbeit sparen soll. Prompt stürzte diese sich in hektische Vorbereitungsarbeiten. Und auf einmal schien es praktischer, statt dreier Disketten lieber eine CD beizulegen. Nicht zuletzt, weil man dann die ganz private Forth-Sammlung (und ein paar hundert Megabyte FTP-Server) mit drauflegen kann und endlich mal alles griffbereit an einem Ort weiß. Sie, liebe Leserinnen und Leser, haben nun zu entscheiden, ob dieses Spontanprojekt zu überstürzt war, denn bis zum 15.12. müssen alle Bestellungen im Sack sein, damit der CD-Brenner weiß, wem er eins mitbrennen soll..

Claus Vogt, Editor des Forthmagazins 'Vierte Dimension'



Einladung zur Forth Jahrestagung '96

vom 19. bis 21. April 1996 in Höfchen (bei Mittweida)

Hotel Höfchen; Moritzfelder Str. 1a D-09648 Höfchen (s.o.)

Organisation:

Thomas Beierlein
Thomas-Mann-Str. 9
D-09648 Mittweida
Tel. (03727) 90941

Am Sonntag findet die Mitgliederversammlung der FORTH-Gesellschaft e. V. statt.

Das Mailbuch

Stand: clv29nov95

Pünktlich zur Weihnachtszeit möchte das Forth-Magazin Vierte Dimension das MAILBUCH der FORTH-GESELLSCHAFT veröffentlichen. Die 11.Jahrestagung der Forth-Gesellschaft hatte sich dies gewünscht, um den Kontakt unter den Mitgliedern zu verbessern.

Die Kürzel der Autorinnen und Autoren der VD haben wir hier auf vielfältigen Wunsch gleich mitveröffentlicht. Bei Rückfragen sollte der betreffende Autor direkt angemailt werden. Funktionen in Forth-Gesellschaft und Redaktion stehen - soweit bekannt - in Klammern dabei.

- 100022.3315@compuserve.com(Ulrich Paul)
- 100542.3405@compuserve.com T (Josef Rieger)
- B.Beuster@bbepoint.forth-ev.de(Bernd Beuster)
- Sebastian.Schmidt@kbbs.org
- anton@mips.complang.tuwien.ac.at (anton ertl)
- bs birgit.steffenhagen@etechnik.uni-rostock.de
.....(Birgit Steffenhagen, Direktorin der FG, Korrektur VD)
- bs114@fim.uni-erlangen.de (zbigniew diaczyszyn)
- clv clv@clvpoint.forth-ev.de W (Claus Vogt, Editor VD)
- dia@almos.vein.hu W (zbigniew diaczyszyn)
- dusan@vukic.forth-ev.de W (Dusan Vukic)
- fep f.prinz@mhb.gun.de T (Friedrich Prinz, forth-gruppe moers)
- ger.xdh0166@applelink.apple.com(ralf neuthe, Direktor FG)
- hd@papalagi.forth-ev.de M (Holger Dyja, Abt. desOrganisation VD)
- hfs@admin.forth-eV.de(Heinz Friedrich Schnitter)
- hg946ha@duc220.uni-duisburg.de (Rainer hansen)
- hp hp@kbbs.org(holger petersen, KBBS'95)
- ius ius@admin.forth-eV.de (Inge Ulrike Schnitter, Anzeigenverwaltung VD)
- jfrr@Forth-eV.deW
- joerg.plewe@jpsforth.forth-ev.de(joerg plewe, Direktor FG)
- joerg.plewe@mpi-dortmund.mpg.de(joerg plewe, Direktor FG)
- jgr joerg.staben@jrgforth.forth-ev.de(joerg staben)
- johannes.teich@kbbs.org(johannes teich)
- kk kk@forthver.forth-ev.de M(Klaus Kohl, Forthvertrieb)
- py paysan@informatik.tu-muenchen.de(bernd paysan)
- rainer.hansen@uni-duisburg.de(rainer hansen)
- rk rolf.kretz@rkaachen.forth-ev.de (Rolf Kretzschmar, Zeichnungen VD)
- ius secretary@forth-ev.de (Inge Ulrike Schnitter)
- tb@htwm.de(Thomas Beierlein)
- tb@tb.forth-ev.de(Thomas Beierlein)
- u.prinz@mhb.hun.de W (Forth-Gruppe Moers)
- u.richter@mhb.gun.deT (Forth-Gruppe Moers)
- uho uho@informatik.uni-kiel.d400.de (ulrich hoffmann, ANS-Kol. d. VD)
- uho uho@pizzicato.forth-ev.de (ulrich hoffmann, ANS-Kol. d. VD)
- us@forth-ev.de T (Udo Schuetz)
- woi@bbs.forth-ev.de(Egmont Woitzel)

(noch) nicht per E-Mail erreichbar sind:

- akg Arndt Klingelberg
- beh Fred Behringer, Kolumne Internationales der VD
- jrd Rafael Deliano

Korrekturen bitte an: MAILBUCH@clvpoint.forth-ev.de

Vielen Dank für Eure Mithilfe,
- Claus -

* verwendete Kurzzeichen:

- T,W, M, J - liest eMail ungefähr täglich, wöchentlich, monatlich, jährlich

Inserenten

- 2 Forthworks
- 2 bigFORTH / Paysan
- 42 cpFORTH
- 42 von & für Forthler
- 44 FORTECH

Rubriken

- 3 Editorial
- 3 Impressum
- 6 Leserforum
- 43 Adressen &Ansprechpartner

Forth-Gesellschaft

- 40 Brief aus der Provinz
- 36 Microcontrollerverleih
Das Forth-Büro
Forth-Magazin sucht..
- 12 Neuheiten in der KBBS
Arbeitsgruppe Marc 4

Termine

- 9 Rochester Conference '96

Forth-Systeme

- 7 Holon-Lite
eForth for Linux
ST6-Mikrocontroller
- 8 PFE 0.0.14 released
F86KANS released oct'95
386-Assembler
Midi auf Atari
P21Forth as shareware
- 16 486-Assembler für F-PC
32-bit protected Forth
- 35 hForth 0.95

ANS

- 8 validation suite

Forth inside

- 10 Quickbasic & Visual C

Prozessorgeflüster

- 9 ShBoom

Bücher

- 41 Zech auch auf Englisch gut
vd sucht Rezensionen
Starting Forth Polyglott

Zeitschriften

- 41 Byte: A brief History
SM&T:ooP von FORTECH?
c't: Open Firmware
- 10 Elrad:Forth-Multitasker
Elrad:Round Robin

Forth Online

- 10 All archives easy access
Taygeta back again
- 12 WWW - Yet another FAQ
- 28 win32forth on taygeta
win32for auch unter OS/2
- 35 Taygeta lebt
- 22 Taygeta und Asterix auf
Konserve

Was noch

- 10 Forth und Musik

Was fehlt:

Die Rubriken Autorengeschenke Anfänger,
Metasprachen und Produktinfos.-

EuroFORTH'95	<i>von Thomas Beierlein ...</i>	11
Der Bericht von der Tagung im internationalen Informatikzentrum Schloß Dagstuhl		
iX	<i>von Rafael Deliano.....</i>	13
DELTA-t) stellt einen neuen Stackprozessor vor - diesmal ein Feldbuscontroller		
Meßtechnik mit dem PC. Teil III	 <i>von Klaus Kohl.....</i>	17
Was macht denn der A/D-Wandler am Druckerport?		
VD quo vadis?	<i>von Claus Vogt</i>	20
Historisches, Utopisches, Nüchternes und anderes aus der Redaktionszentrale		
Das große Weihnachtspäckchen	<i>von Claus Vogt</i>	22
Wer bisher kritisierte, daß die VD nur vier mal im Jahr erscheint, zuwenig Wettbewerbe organisiert und noch nichtmal eine CD herausbringen kann, der muß jetzt zahlen. Und zwar sofort, denn am 15.12.95 ist Bestellschluß!		
Betreff: Forth Magazin sucht Programmierer	 <i>von Claus Vogt</i>	24
Trotz profaner Programmiersprache gewann Gunter Schenck den Programmierwettbewerbs eines namhaften Forth-Magazins, obwohl gar keiner organisiert war.		
Das Werkzeug Win32Forth	<i>von Jörg Staben</i>	27
Auch mit einem wirklich fetten Forthsystem kann man sich auf den Daumen hauen		
Parallel-Forth: Der neue Weg	 <i>von M.B. Montvelishsky</i>	29
Auch für parallele Datenverarbeitung ist Forth der richtige Weg. Vielleicht das erste Mal, daß ein russischer Autor mit ausdrücklicher amerikanischer Genehmigung in einem deutschen Forth-Magazin zu lesen ist.		
Umlaute in den Namen von Forth-Worten	<i>von Fred Behringer</i>	37
Parallel zum russischen Artikel läßt unser Internationalismusbeauftragter uns ein wenig an seinem Forth nippen. Es kann weit mehr als die Tüddelchen über den Umlauten zu parallelisieren.		
Jetzt geht's - ANS Forth!	 <i>von Ulrich Hoffmann</i>	14
'Final Draft' dpANS94 im WWW, mit Kurzlisting schneller ins ANS.		
Forth International	<i>von Fred Behringer</i>	33
Neues aus Holland, Amerika, England.		
'Gehaltvolles' aus 'Het Vijgeblad' 1, Notausgabe.	32

Nur noch bis 15.Dezember..

kann man das Forth-Weihnachtspäckchen bestellen. Die CD 'forthige Weihnacht 95' enthält die Listings dieser VD, der VD des letzten Jahrganges und aller vorhergehenden Jahrgänge, die auffindbar waren. Nebenbei auch noch zwei komplette Ftp-Server und Quelltextsammlungen von Ulrich Hoffmann, Claus Vogt und anderen fleißigen Forthlern.

Gleich bestellen! Info auf Seite 22

1	1.042	30.11.95	22:07	vd95-4\ansi0495.lst
2	6.088	4.11.95	18:22	vd95-4\doerjrg
3	4.184	9.11.88	11:09	vd95-4\doermake.seq
4	881	4.12.95	4:58	vd95-4\mess-lpt.4th
5	2.840	4.12.95	20:56	vd95-4\vd95-4.lst
6	3.335	24.01.94	15:47	vd95-4\fastinfo\fastinfo.seq
7	1.537	13.09.95	23:59	vd95-4\lst2pw\lst2pw-c.c
8	22.326	30.11.95	8:39	vd95-4\lst2pw\lst2pw-f.exe
9	7.820	12.09.95	23:59	vd95-4\lst2pw\lst2pw-p.ans
10	5.346	30.11.95	8:40	vd95-4\lst2pw\lst2pw-p.exe
11	1.899	10.09.95	0:01	vd95-4\lst2pw\lst2pw-p.pas
12	1.571	7.11.95	20:31	vd95-4\parallel\par-skala.seq
13	927	7.11.95	20:31	vd95-4\parallel\par-uhr.seq
14	4.886	7.11.95	20:32	vd95-4\parallel\parallel.seq
15	2.344	14.11.95	20:44	vd95-4\uf38214format.c2



=Listi
ng auf

2PW





Reaktionen VD 4/95

zu 'Forth Optimieren' von Bernd Paysan in VD 3/95, S.21

Hallo Bernd,

ich hatte heute mal Zeit und Muße, mich mit Deinem Optimierungsbeitrag zu beschäftigen.

Klasse, hat mir bis jetzt großen Spaß gemacht!

Ciao, Joerg joerg.staben@jrgforth.forth-ev.de (Joerg Staben)

zum Editorial von VD 3/95

... Ja, ist super geworden. Was mir aber aufstößt, ist die falsche Verwendung des Begriffs "sakrale Programmiersprache". Einzige sakrale Programmiersprache für einen Forth-Priester kann natürlich nur Forth sein. Das, was so gemeinhin der Pöbel programmiert (also C, Pascal, Basic und andere Gossensprachen) nennt sich "profan". Kein Wunder, daß alle anderen Forth für "Hokuspokus" halten. Die Definition von Hokuspokus ist übrigens:

: hokuspokus (:-)
hex hex ;

Bernd Paysan

[Du hast natürlich recht, es hätte 'profan' heißen müssen /clv]

PFE mit GUI

Halli, hallo.

Ich moechte u.U. pfe auf einer HP-UX-Kiste mit einem einfachen GUI versehen. Dazu scheint mir Tcl/Tk geeignet. Hat schon mal jemand sowas gemacht?

Was ist, wenn man ähnliches unter Windows machen moechte; oder noch besser, wenn man ein Forth in eine Applikation einbetten moechte? Kann man ein vorhandenes Windows-Forthsystem eingebettet (irgendwie) betreiben?

Erfahrungen? Ideen? Vorschlaege? QUELLTEXTE?

Joerg Plewe; JPS@Forth-ev.de; +49 (0)208 497068 Haarzopfer Str. 32; D-45472 Mühlheim

zu 'flux' in VD 3/95, S.23

Hallo Claus

Bin gerade in San Francisco um ein paar Faeden wieder aufzugreifen. Es scheint mir das an der Westkueste mehr Platz zwischen den Atomen besteht, alles laeuft im lebenswerten Sinn ab. Na ja New York ist sowieso nicht fuer Menschen gebaut. Habe vom meiner Abreise die M.L.F. (fLUX volume two) abgeschlossen, koennte Dir 'ne kurze Algorithmusbeschreibung zukommen lassen. Hier ist im Moment alles voller Nebel.

ps: Planet Genius (flux volume one, siehe VD 3/95, S.24) hat die #5 position auf der Kanadischen Radio Hitparade eingenommen.

Gruss

Frank Rothkamm fLUXeast: dadua@aol.com fLUXwest: mjmk@sfsu.com

zu 'DOER & MAKE kurz vorgestellt' in VD 3/95, S.17

Wer die beschriebene DOER-Utility von Leo Brodie unter Tom Zimmers F-PC-Forth ausprobieren moechte, findet die Quellen in der KBBS:

forth/portugal/ibm/fpc/ doermake.arc 17.10.1995 12:48:18 2432

Die Telefonnummern der KBBS stehen wie immer hinten auf der inneren Umschlagseite der VD

Claus Vogt

Hi there,

Es war zumindest eine kleine Illustration auf der Titelseite. Innen sah's jedoch dahingehend dueter aus. ICR war der einzige Artikel mit Bildchen (5x5cm). Es werden auch keine von den Autoren kommen, wenn man es nicht explizit einmal in einem



Editorial fordert, daß Autoren sich um Illustrationen für ihre Artikel kümmern sollen.

Die Seiten 4, 12&13 waren vom Layout so gut aufbereitet, daß Illustrationen hier unnötig waren. Aber das geht nicht immer.

Auf den Seiten 30&31 fehlten die Rahmen, um die zusammengehörenden Teile optisch zusammenzuhalten.

Die nichttechnischen Artikel von Rothkamm und Zembrod waren insofern passend, weil sich das Blatt ja "Magazin" und nicht "Fachzeitschrift" nennt. Ein Magazin will unterhaltsam sein, eine Fachzeitschrift legt auf Tiefgang wert. Obwohl sie von der Zielsetzung richtig lagen, waren die Texte allerdings hart zu lesen. In dem Punkt ist man immer den jeweiligen Autoren ausgeliefert.

Daß jetzt auch Display-Schriften in Überschriften verwendet werden wie in "Flux" ist erfreulich. Da sie aber zwangsläufig schwieriger lesbar sind, sollte man dabei eine 20 - 50% größere Schriftgröße als normal verwenden.

Rafael Deliano (jrd), 10.10.95

zu Fred Behringers Frage nach dem kleinsten Forth in VD 3/95, S.6

Zwei Varianten

Memory Map

Loader

Hier als reales Beispiel (Bild 1) die MemoryMap von nano-FORTH auf einem Mitsubishi-6502-Controller. Wenn man echten Sourcecode compilieren will braucht man Primitives und den FORTH-Compiler im EPROM.

Hier also 8kByte. Den Assembler kann man dann bereits ins RAM compilieren. Praktisch will man aber das gesamte Entwicklungssystem im EPROM haben. Sonst verliert man auch diese Teile, wenn man auf den Resetknopf drückt. Etwas was man beim

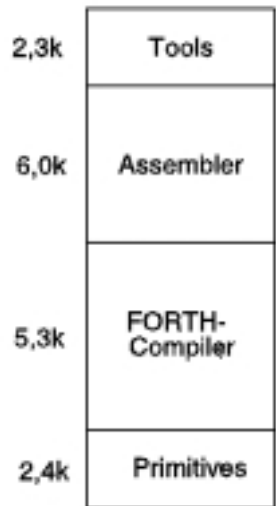


Bild 1 : Memory Map

Entwickeln häufig tut. Nano-

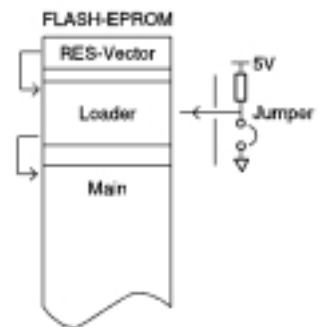


Bild 2 : Loader




 Leserbrief: Am liebsten kurz. Sonst trifft uns die Pflicht zur Kürzung. Die Redaktionsadresse lautet:
Forth-Magazin 'Vierte Dimension'
Claus Vogt, Ebersstr. 10,
D-10827 Berlin,
vd@FORTH-ev.de
Forth. Kurz und Knapp. Das

FORTH wird deshalb komplett mit 16k im Zielsystem installiert.

Zum Ändern von FLASH-EPROMs ist ein simpler Bootloader geeignet. Ein kurzes Assemblerprogramm wird mittels Programmiergerät knapp unterhalb der Reset-Vektoren im FLASH-EPROM installiert. Im Betrieb wird es nach dem Reset angesprungen und fragt einen Jumper auf der Leiterplatte ab. Ist dieser gesetzt wird nicht aufs Hauptprogramm weitergesprungen. Sondern der Loader wartet darauf, daß er über V24 ein Intel-HEX-File erhält. Dieses programmiert er dann ins EPROM. So kann man bequem neue FORTHS oder Applikationen auf einem Einplatinencomputer nachladen. In diesem Fall wird offensichtlich reiner Objectcode übertragen.

Formate die zwischen reiner Source und reinem Objectcode liegen, also Source die mit Magic numbers angereichert ist, sind wenig anwenderfreundlich.

Rafael Deliano, 10.10.95

Korrektur

Zu 'The Freiburg..Story' in VD 3/95, S.34 und 'ultraForth..' VD 3/95, S.11 in VD 3/95

Ich glaube, ich muß dem Freiburg-Zurich-Nashville-Story aus der VD 3/95 noch ein paar Bemerkungen nachschicken. Die Geschichte ist unmittelbar nach unserer Rückkehr aus den Staaten entstanden. Ich war völlig euphorisch und ziemlich anglophil gestimmt und konnte sie irgendwie nicht anders als auf Englisch und

in etwas flapsigem Stil schreiben. Man möge mir das nachsehen. Außerdem fehlen noch die Namen meiner Mitstreiter. Dies sind Matthias Ruhl, dessen Initiative unsere Teilnahme überhaupt zu verdanken ist und der wohl auch der Schnellste von uns dreien ist, und Christian Wetzel. Beide studieren an der Uni Freiburg Informatik; zur Zeit des Wettbewerbs waren sie im 3. (!) Semester.

Und noch zum ultraFORTH rev3.82 (VD 3/95 Seite 11): Das Wort BUSREAD, von dem ich schrieb, ich wisse nicht mehr, was es genau macht, stellt eine Verbesserung des Wortes BUSINPUT dar. Letzteres liest einfach N Bytes vom seriellen Bus, egal was passiert. BUSREAD prüft beim Lesen auf EOF und bricht gegebenenfalls unter Angabe der nächsten zu beschreibenden Adresse ab. Falls es jemanden interessiert, hier die Quelltexte:

```

: businput ( adr n --)
  bounds
  ?DO bus@ l c! LOOP
  pause ;

```

```

: busread ( adr1 u
  -- adr2 true )
  ( adr1 u -- false )
  pause i/o-status?
  $40 and
  IF
    drop true exit
  THEN
  bounds
  ?DO
    bus@ l c!
    i/o-status? $40 and
  IF
    I 1+ UNLOOP
    true exit
  THEN
  LOOP false ;

```

Außerdem hatte das Programm 4format.c zum ASCII-Konvertieren von .D64-Disk-Images einen kleinen Fehler. Ich hatte gedacht, das uF-Format ueberspringt beim 1541-Directory nur die Sektoren 18:0 und 18:1, es sind aber 18:0 bis 18:2. Das korrigierte Programm befindet sich als 4format.c2 auf der Listingdiskette und in der Mailbox.

Philip Zembrod, zembrod@informatik.uni-freiburg.de

Der Redaktion herzliche Gratulation zur letzten VD. Das Blatt

ist kontinuierlich reizvoller geworden, und die

letzte Ausgabe war eine reine Freude. Einschließlich des Berichtes von Frank Rothkamm. Ich hätte nie gedacht,

je Forth und Laurie Anderson in einem Satz vereint zu finden! Hierzu volle Unterstützung für die Redaktion

von Eurem dankbaren Leser.

100012.201@compuserve.com
(Wolf Wejgaard)

Zu 'DOER & MAKE' von Thomas Höhenleitner in VD 3/95, S.17 ff

Für F-PC-Begeisterte, die die Listings des Artikels testen möchten, haben sich inzwischen zwei verschiedene Listings zum DOER von Leo Brodie eingefunden. Die eine wurde von Jörg Staben gefunden, die andere ist in der KBBS zu finden:

25.04.1995 08:56:48 2432
forth/portugal/ibm/fpc/
doermake.arc

Ob sie identisch sind, weiß ich nicht und habe sie einfach mal beide auf die Listingdiskette VD95-4 aufgenommen.

Claus Vogt

Holon-Lite

Die neue Forth-Entwicklungsmethode Holon ist jetzt frei verfügbar in einer 'Lite' Version HOLON-LT (in allen guten Forth Kiosken und auch im VD-Weihnachtspäckchen).



HOLON-LT ist eine vereinfachte aber voll funktionsfähige Ausführung von Holon. Sie koennen damit das neue Entwicklungskonzept von Holon studieren und Turnkey-Applikationen fuer 80x86 Systeme unter DOS entwickeln. Das System beschreibt sich selbst in einem Tutorial und dem online Handbuch.

Wolf Wejgaard

Forth-Systeme: eForth for Linux

I've succeeded in porting eForth to Linux for Intel CPU's. In itself this is no big feat, at least two other people have done this in the past. The added value in my case is:

This eForth is written in assembler. A small C program loads the image file and provides I/O functions.

This eForth has DOES> ! Other enhancements are [CHAR] and S" .Can execute any shell command from eForth using the following syntax: S" ls -a" SYSTEM

The return value of the SYSTEM command can be fetched with: RETURNCODE @

Real interactive KEY , but eForth's input can still be redirected to a file or pipe. (I'm indebted to Robert Hoeller (rhh@lht.rwth-aachen.de) who provided code to do this in a simple and efficient matter).Can call any C function from Forth using SYSCALL (a jump table approach).

I'm not convinced that this eForth is useful enough to be put on Taygeta, for me it was just an exercise before moving on to port iForth to Linux. FWIW, the code is available on my home site (ftp iaehv.iaehv.nl, directory /pub/users/mhx, file eforth.zip). If you want to rebuild the image file you'll have to do it with Borlands TASM under MS-DOS.

Warning: Real C programmers may get an heart attack viewing my C hacks.

mhx@iaehv.IAEhv.nl (Marcel Hendrix) in /comp/lang/forth, nov '95l

Forth-Systeme: ST6-Microcontroller

Auch der Controller ST6 von SGS Thomson - Ulli Hoffmann nannte ihn einmal die 'ultimative Non-Forth-Maschine' - hat seit 27.10.95 'sein' Forth. Olaf Kaluza (olaf@criseis.ruhr.de)

freut sich auf Feedback, weist aber darauf hin, daß es sich um eine Alfa-Testversion handelt. Sein 'ST6FORTH Forthcrosscompiler for ST6 Microcontroller' ist in der Version 0.1a erschienen, hat aber bereits ein wenig deutschsprachige Dokumentation. Die Datei findet sich auf localhost.ruhr.de im Pfad /pub/olaf/st6forth heißt st6forth.0.1a.tar.gz und hat trotz des winzigen Prozessors die stolze Größe von 100k. Sie erfordert einen SGS-Development-Kit-Assembler, wahlweise Linux oder DOS und wird unter der Gnu Public Licence vertrieben.

clv nach de/comp/lang/forth

Forth-Systeme: PFE 0.0.14 released

Hello,

I just uploaded a new release (patchlevel whatever) of the portable forth environment, i.e. the most complete and available free ANS Forth, to

roxi.rz.fht-mannheim.de/pub/languages/forth/pfe/pfe-0.9.14.tar.gz

Diffs vs. 0.9.13, the same files in ZIP-format and DOS-OS/2-EMX-binaries are in the same directory. The changes this time are: 0.9.13 -- 0.9.14

- Fixed 'READ-LINE' reporting EOF too late.

- Fixed a bug in 'CATCH' occurring with some (Sun acc) compilers.

- ☺ Fixed things like '-.+ ' being a floating point number.
- ☺ Fixed ~/.pferc not loaded.
- ☺ Fixed 'PLACE' (from the L&P-F83 words).
- ☺ Made PFEINCLUDE include path work.
- ☺ Made codes THROWN by signals compatible with gforth.
- ☺ Finds files in another user's home like this '~user/...' now.

☺ 'SEE' now displays floating point literals in exponential form.

☺ Added constants 'S-K1' through 'S-K10' for codes returned by 'EKEY' when shifted function keys are pressed.

This time I tested it with Linux, FreeBSD, alpha/OSF1, AIX3 and HPUX. I hope it works. Else drop me a line. Thank you!

*duz@roxi.rz.fht-mannheim.de
<Dirk Zoller>*

Forth-Systeme: F68KANS released oct'95

Seit Oktober 95 ist ein neues Release von F68KANS verfügbar. F68KANS ist ein ultraportierbares, 32bit nativecode ANS Forthsystem für die Prozessoren der Motorola 68k-Familie.

Neu ist ein Signifikanter Speedup von Kontrollstrukturen (x3), Variablen und Konstanten (x3), >R, R> und R@ (x3). Der Code ist kürzer, mehr in Forth statt Assembler geschrieben. Die Entwicklungsumgebung für Ataris wurde ästhetisch ansprechender gestaltet. Sie erlaubt jetzt beliebig viele Editoren gleichzeitig, das Kompilieren von im Editor markiertem Text auf Knopfdruck und das Kompilieren aus dem Clipboard

Das System habe ich bereits auf ftp://taygeta.oc.nps.navy.mil/pub/Forth/ANS/f68kans.zip abgelegt.

Das System ist weiterhin fuer den privaten Einsatz kostenlos. Fuer den kommerziellen Einsatz sind Lizenzbedingungen mit mir abzusprechen. joerg.plewe@jpsforth.forth-ev.de (Joerg Plewe)

*Haarzopfer Str. 32, D-45472
Muelheim, Germany +49 (0)208
497068 /*

Forth-Systeme: chipForth on F68HC11

Our chipFORTH cross-development system for 68HC11s

includes an assembler, multitasking, lots of libraries, extensive docs and more. It can run on the NMI boards.

For more info on chipFORTH, check our web page: <http://www.earthlink.net/~forth>.

erather@forth.com (Elizabeth D. Rather) in /comp/lang/forth, Sep'95

Forth-Systeme: 386 Assembler

With all the talk about assemblers lately I thought I'd toss in my two cents' worth. I have posted my 386 adaptation of Mike Perry's F83 8086 assembler to: <ftp://taygeta.oc.nps.navy.mil/pub/Forth/Tools/asm386.fth> It will run on Win32Forth but should be portable to another 32-bit architecture. It will not assembly every opcode and operand, but was designed to support a metacompiler. Drop me a line if you find it useful.

*mckewan@netcom.com
(Andrew McKewan) in
/comp/lang/forth, Sep'95*

Forth-Systeme: Midi auf Atari ST

Okay, I found Formula, at <ftp://bigbird.csd.scarolina.edu/pub/formula> as advertised. Except I can't get it to build! Very frustrating, because I printed and read the manual and I think it is just what I was looking for...

I have an Atari 520ST, memory expanded to 1 meg, TOS version 1.4 (rainbow). I have Forthmacs 1.1. I checked the 'flags' source file, then did an 'fload load', and it starts building as expected, but in 'machine.st', around the definition for '(set-default-mask)' or 'set-default-mask', it sort of loses its mind (appears to me like Forthmacs gets confused about its context, because it starts complaining about each word, including '\ ' and comments it shouldn't be parsing at all).

I've looked at many of the source files, fiddled with the flags and commenting out some of the code, made sure I had no

accessories and stuff eating up memory ('memfree .' always gives me over 100k), put debugging stuff in to try to narrow the problem down, but I'm basically stumped. I need a clue, or else just a pre-compiled FORMULA.TOS (which the manual implies was part of the original release disks).

andy@access2.digex.net (Andy Moore) in /comp/lang/forth und /comp/music Sep'95

Forth-Systeme: P21Forth as shareware

I finally got around to posting P21Forth 1.02 as shareware. In addition to the User's Manual in html and .DOC and the OK source code I have placed the ROM images and source at my web site. I will also upload a copy of P21-4TH.ZIP to Skip Carter's ftp site. I have updated the "what's new" on my home page <http://www.dnai.com/~jfox> with an F21_Status document and a description of the components in the P21Forth 1.02 shareware package. The first F21 prototypes are expected back in about a month for testing.

*Jeff Fox in comp/lang/forth,
Sep'95
Ultra Technology
2510 10th St.
Berkeley CA 94710
(510) 848-2149
jfox@netcom.com
<http://www.dnai.com/~jfox>*

"Do we have a validation suite for declaring a Forth dialect as being ANS compliant or not? ... If we don't, we SHOULD."

John Hayes wrote a test program that is very popular. I have used it to validate some of my compilers and found it to be very useful. There are two files, you can get them from:

<ftp://taygeta.com/pub/Forth/ANS/coretest.fth>
<ftp://taygeta.com/pub/Forth/ANS/tester.fth>
*carter@adak.oc.nps.navy.mil
(Skip Carter) in /comp/
lang/forth, dec'95*

Prozessorgeflüster: ShBoom

"I read today in Electronics Weekly (Nov 8) that National Semiconductor is to start shipping a new Microprocessor called ShBoom 'by the year end'. This has been developed by Patriot Scientific of California. Its a 32bit 100MHz un-pipelined dual-stack dual processor chip. Forth is not mentioned in the article, but a variant would obviously fit well.

I recall reading somewhere that Chuck Moore had some connection with something called ShBoom, and I suppose that this is the same thing?"

*Keith@wootten.demon.co.ukin
/comp/lang/forth, nov '95*

Similar but not the same. The original ShBoom was a 32bit 50MHz done at Oki. This ShBoom besides being faster has an expanded architecture and instruction set. This ShBoom has been in development longer than F21. The original ShBoom had an instruction to overflow the bottom of the onchip stack registers to a stack in memory via a stack pointer, but it was not handled automatically when the onchip stack register pointer wrapped. So code had to manage the onchip stack overflow. The original was really a Forth Engine but did have a floating point step instruction (I think).

The new one had many enhancements with new hardware and a lot of new instructions.

You can order a manual for the original ShBoom from Offete Enterprises. But there were only a few chips, and Chuck only had two.

I saw a presentation a few years ago. The chip has lots of stuff planned at that time. I do not know about the current details. I have heard Chuck talk about the chip a lot of times.

jfox@netcom.com (Jeff Fox)

A friend of mine recently sent me a copy of their preliminary data data sheet. It looks very interesting. They have a web site at <http://www.ptsc.com/>, but it doesn't work very well with my preferred browser (lynx).

In addition to the two stacks, it has 16 global registers. Data can be moved between the stacks and the global registers. You can address memory indirectly through the top of the return stack, including predecrement and postincrement addressing modes. There is an additional register called X through which you can indirectly access memory, again including predecrement and post increment addressing modes.

The processor has a 32-bit bus and uses eight bit instructions, so it fetches four instructions at once from memory. It has a microloop instruction that allows you to conditionally re-execute the instructions last fetched, so if you have a three-instruction loop it should do that really fast :-). Writes are posted; i.e., the processor gives the write to the memory interface and keeps on executing.

In addition to the CPU, there is also a DMA controller and an I/O processor on the chip. The I/O processor has enough intelligence to test inputs, generate outputs, delay, and loop. It can also interrupt the CPU. The I/O processor has access to the global registers, so the CPU and I/O processor can communicate using the global registers. The I/O processor has absolute priority for performing bus accesses, so for the CPU to do anything the I/O processor must be executing a delay.

Unlike MuP21, ShBoom does not have fixed regions allocated to DRAM and SRAM. Although it does have a memory controller that can run DRAM, you can program the regions allocated to DRAM and SRAM.

ivie@cc.usu.edu (Roger Ivie)

Prozessorgeflüster: shBoom

mikc@gnu.ai.mit.edu (Mike Coughlin) writes: "Is the National Semiconductor ShBoom chip a Forth processor not?"

The original posting said that the chip was produced for Patriot Scientific, by National Semi (as the foundry, I suppose.) I found that Patriot has information on the web at:

<http://www.ptsc.com/shboom.html>

Information is somewhat spotty, but does include an opcode list. No mention of price, availability, or any relationship to any Forth engine of past or present.

milan@xyplex.com (Milan J. Merhar) in comp/lang/forth, dec '95

I have received a 'Preview' of the ShBoom chip, it comprises two pages and dangles a lot of carrots without much meat. I hope to receive further information soon.

In addition to what others have mentioned in this thread, ShBoom also has Fast multiply (how fast?) Fast bit-shifter (Barrel Shifter?) and 'Hardware single- and double-precision IEEE floating point support'.

All this at 100MHz with a flat address space makes one powerful Forth chip. Interestingly, Forth is not mentioned anywhere in the literature I have so far :-)

The company responsible for the design is:

Patriot Scientific Corporation
(619) 679-4428

*keith@wootten.demon.co.uk
(Keith Wootten) in comp/
lang/forth dec '95*

Keith Wootten writes: "In addition to what others have mentioned in this thread, ShBoom also has Fast multiply (how fast?) Fast bit-shifter (Barrel Shifter?) and 'Hardware single- and double-precision IEEE floating point support'."

ShBoom does not have floating point instructions. It does have

instructions that let you manipulate parts of an IEEE number; for example, you can add the exponent fields together.

*ivie@cc.usu.edu (Roger Ivie) in
comp/lang/forth, nov '95*



Termine: 1996 Rochester Conference in Toronto

16th Annual 1996 Rochester
Forth Conference on Open
Systems

June 19-22, 1996

Ryerson Polytechnic University
Toronto, Ontario, Canada

Preliminary Call for Papers

The Rochester Conference will again provide a forum for researchers, developers, and vendors to present the latest practical results dealing with open systems. The conference seeks original papers relevant to the design, development, implementation, and use of open systems. Conference topics include:

- ☉ The Open Boot Standard - Plug and Play Systems
- ☉ Scripting Languages - SGML and HTML

© Distributed Computing - Educational Issues

Other areas of interest are Forth programming standards, embedded systems, real-time systems, and the use of Forth for scientific and engineering applications.

Additional information will be posted on the world-wide web as it becomes available: <http://maccs.dcss.mcmaster.ca/~ns/96roch.html>

Kontakt:
Nicholas Solntseff
Dept. of Computer Science & Systems
McMaster University
Hamilton, Ontario
Canada
L8S 4K1
ns@maccs.dcss.mcmaster.ca

Was noch: Forth und Musik

aus /comp/lang/forth, sept'95

Hey, has anybody got any experience or clues for MIDI applications in Forth, preferably on an Atari ST? I've just set out to explore

using Forth to (a) map my playing (MIDI in) to notes/sounds (MIDI out), possibly with various kinds of transformations; (b) generate/compose music/sounds and play them (MIDI out); (c) analyze or "listen" to my playing (MIDI in); etc... I decided to look at Forth because I've hacked it a little in the past, it looks to me like it would be relatively easy and even fun (I'm sick of structured/planned coding at work all day), as well as appropriate to exploratory/experimental stuff. Anyway, I'd be interested in discussing anything you've already done in this area, or any thoughts you have on how it can/should be approached... Thanks...

Andy Moore

In 1991 I made a cassette album of algorithmic music on the Atari ST. I used Mitch Bradley's Forthmacs and an extension package called FORMULA (Forth Music Language). I found the package

to be extremely versatile, and recommend it highly to anyone interested in algorithmic composition.

One of the most interesting aspects was the ability to interact with the composition as it is playing. For example, I created several "musician" processes, each with its own style and instrument assignment. These entities were taking their cues from a global variable that contained the current chord. I could change that chord from the Forth terminal while the music was playing, thus influencing the musical decisions that the processes were making.

I have yet to find another musical computer language that approaches the flexibility of the Forthmacs/FORMULA package.

Bob Lee



Thanks so much for the responses I've received (nearly a dozen)!! Over the labor day weekend I "auditioned" a half-dozen Forth's for the Atari ST (wow, that's impressive by itself), and have tentatively settled on Forthmacs as the best for my purposes. I'll look for Formula next. Again, thanks for the help and tips...

Andy Moore

Formula could be found at
<ftp://bigbird.csd.scarolina.edu/pub/formula>
source and doc.

Sten Dahlgren

Forth inside: Quickbasic & Visual C

Clif Penn writes: "Better than a game, QuickBasic was originally written in FORTH". This is not correct. QuickBasic was not written in Forth. It did, however, compile BASIC code to direct threaded code and ran this code on a virtual stack machine with a very familiar "threaded code interpreter." At one of Microsoft's yearly "system strategy seminars" I ran into the lead programmer for QuickBasic, who was being quite smug about his clever implementation, and had the satisfaction of telling him that he was reinventing Charles Moore's technology 20 years later.

...
You are confusing the language with the implementation (a very common problem of Forth zealots, and an underlying theme in many of the discussions/debates/flame wars in this newsgroup.) Just about anything can be compiled to direct-threaded code, tokenized code, pseudo-code, etc. UCSD Pascal compiled to a pseudocode that ran on a virtual machine that looked very Forth-like in a debugger. QuickBasic actually compiled to what we recognize as direct threaded code. Visual C++ 1.0 had the capability to compile to token-threaded code to save memory (I don't know if the new 32-bit Visual C++ 2.0

does this though). I've even heard of an early Fortran that compiled to some kind of threaded code. But none of this means that Forth was anywhere involved, either in the creation of the language compiler or in the programming language/interface presented to the user.

Looking at this from another perspective, Tom Almy's FORT-HCMP compiler translates Forth source code to optimized in-line machine code. There's no threaded code involved at all, and in fact if you look at a compiled application in a machine-level debugger you'd be pretty hard-pressed to say WHAT the original programming language was. Does this mean his compiler is not a Forth programming system? Only a person who confuses the lan-

guage with the implementation would make such a statement.

Making bogus claims that "popular product xxx was written in Forth" will not serve to further the credibility or popularity of Forth...

much the contrary, since such claims are easily refuted (and will be).

duncan@nic.cerf.net (Ray Duncan) in /comp/lang/forth, Sep'95

Forth online: All archive easy access

All the Forth related files freeware and shareware archives we've been able to find (plus another 23,853 programs) are now linked through JUMBO! Over 600,000 links for fast and easy and access to more than 200 sites! Never any waiting and full descriptions of every program BEFORE downloading
Access us at: <http://www.jumbo.com/>
or: <http://www.jumbo.com/prog/dos/forth/>
xBQnMeCx@jumbo.com in /comp/lang/forth, nov'95

Forth online: Taygeta back again

After much haggling with name servers, taygeta is back.

my Forth page: <http://www.taygeta.com/forth.html>

Forth Interest Group page: <http://www.taygeta.com/fig.html>

my home page: <http://www.taygeta.com/skip.html>

Anonymous FTP: <ftp://www.taygeta.com/pub>

FYI the new IP address is 192.160.158.184 (at the moment this is the IP address of BOTH taygeta.com and www.taygeta.com).

Everett (Skip) Carter, skip@taygeta.com, comp/lang/forth, oct'95

EuroFORTH '95

27. bis 29. Oktober 1995

von Thomas Beierlein

tb@tb.forth-ev.de; Thomas-Mann-Str. 9; D-09648 Mittweida

Die diesjährige europäische Forth-Konferenz fand im internationalen Begegnungs- und Forschungszentrum für Informatik - Schloß Dagstuhl mit 32 Teilnehmern aus ganz Europa und den USA statt.

Stichworte: Tagung euroForth95 Dagstuhl

Das Umfeld und auch die Organisation der diesjährigen Konferenz war beeindruckend. Der Tagungsort lag nicht nur in einer reizvollen Landschaft, auch seine Ausstattung war hervorragend. Er verfügte neben einer ganzen Reihe von Tagungsräumen, die bestens mit Tafeln, Projektoren und Computertechnik ausgerüstet waren, auch über eine sehr reichhaltige Informatikbibliothek. Wie schon in vorherigen Jahren wurde die Konferenz von *Marina Kern* von *deltaT* zur besten Zufriedenheit der Teilnehmer organisiert. Ich denke, ich spreche im Namen aller, wenn ich mich mit diesem Beitrag bei Ihr bedanken möchte.

Das Repertoire der 19 Vorträge zeigte die Vielfalt der gegenwärtigen Arbeiten an und mit Forth und war ein schöner Beweis, daß Forth noch längst nicht am Ende ist.

Unter anderem zeigte es sich, daß Forth in der Welt der 'bunten Fenster' eine feste Position erobert hat. *Nick Nelson* schilderte in unnachahmlich amüsanter Art und Weise seine Probleme bei der Gestaltung mehrsprachiger Windowsprogramme. Er präsentierte eine Lösung für die klassischen europäischen Sprachen, mußte jedoch bei den etwas ausgefalleneren Sprachen wie Arabisch und Japanisch (vorläufig noch!) passen. *Gary Ellis* stellte eine Anwendung von MPE's 'ProForth for Windows' vor, die Bedieneraktionen zur automatischen Presentation von Programmen emuliert.

Während *Egmont Woitzel* die Arbeitsweise von fieldFORTH, einem leistungsfähigen, interaktiven Crosscompiler unter 'ComFORTH für Windows', vorstellte, präsentierte *Thomas Höhenleitner* die Anwendung dieses Systems für die Programmierung digita-

ler Signalprozessoren für Bilderkennungsaufgaben.

Einen wichtigen Punkt nahmen auf der diesjährigen Konferenz die Bemühungen zur Quelltextverwaltung und zur Dokumentation von Forth-Programmen ein. *Ulrich Hoffmann* erläuterte sein Konzept zum Aufbau 'intelligenter Quelltext-Bibliotheken', bei deren Import nur die angeforderten Komponenten in die Applikation übernommen werden. Seine Erfahrungen mit einer Verwaltung des Quelltextes in einer Datenbank stellte *Wolf Wejgaard* vor - ein Konzept, das er seit einigen Jahren in seinem interaktiven Metacompiler 'Holon' realisiert hat. Hauptvorteil der geänderten Quelltextverwaltung ist, daß Werkzeuge wie *View* oder *HyperText* angewendet werden können, ehe die Programme compiliert sind.

Die Anwendung von Donald E. Knuth's 'Literate Programming' auf Forth diskutierte *Peter Knaggs* in seinem Paper. Das dahinter stehende Konzept erlaubt es, den Quelltext und dessen Dokumentation als eine Einheit zu schreiben und später entweder in ein übersetzbare Programm oder in eine gedruckte Dokumentation zu transformieren. Hauptproblem bei der Anwendung auf Forth scheint jedoch der typische Arbeitsstil eines Forth-Programmierers, bei dem das Endprogramm nicht allein durch vorherige Überlegung, sondern inkrementell und iterativ in Vorausplanung, Versuch und Irrtum entsteht.

Daß man mit Forth auch "richtige Probleme" lösen und Geld verdienen kann, zeigte u.a. *Vassil Lolov*. Er stellte eine Steuerung für Steinkohle-Abbaumaschinen vor, die eine automatische Verfolgung der Kohleflöze erlaubt und

den Abbau tauben Gesteins minimiert. *Paul Frenger* berichtete über seine Arbeiten zur Steuerung künstlicher Gliedmaßen. Forth kommt hier sowohl in den, die Gelenke steuernden, Mikrocontrollern, als auch für die übergeordnete Bewegungskoordination zum Einsatz.

Stephen Pelc zog ein Resumee aus einem ESPRIT-Projekt, welches Verbesserungen in Integration und Debugging von Software in verteilten Steuerungssystemen zum Ziel hatte. Ein interessanter Punkt seines Vortrags war die Verwendung eines token-threaded Codes, wie er analog auch in Mitch Bradleys Open Firmware Verbreitung findet. Auch *Malcolm Bugler* plauderte aus dem 'kommerziellen Nähkästchen' und kam, ausgehend von einer Erfolgsanalyse für Softwareprojekte, zum Schluß, daß nicht die verwendete Programmiersprache, sondern die Einstellung und Arbeitsweise des Programmierers ausschlaggebend für den Erfolg eines Projektes sind.

Klaus Schleisiek stellte drei verschiedene Befehlssatz-Strukturen für den Feldbuscontroller iX1 vor und diskutierte den Einfluß der vorgestellten Varianten auf Speicherbedarf und Zugriffsraten der Programme, sowie die Komplexität von Befehlsdekoder und Interruptmechanismus.

Eine ganze Reihe von Vorträgen befaßte sich mit Arbeiten zu Forth-Compiler-Techniken. *Markus Dahm* stellte die im letzten Jahr erreichten Fortschritte an 'natOOF' in den Mittelpunkt seines Vortrags. Es gelang ihm sein objektorientiertes Forth so zu erweitern, daß man Algorithmen und Anweisungen in einer sehr natürlichen Sprache beschreiben kann. Aus eigener Erfahrung kann ich sein System als einen sehr eleganten und leistungsfähigen Ansatz für ein objektorientiertes Interpretersystem einschätzen.

Sergei Baranoff präsentierte einen in Forth programmierten offenen C-Compiler für embedded Controller, der in Zusammenarbeit mit *Christophe Lavarrenne* entstand. Genau den umgekehrten Weg beschreitet die Gruppe um *Anton Ertl*. Er berichtete über Arbeiten zur Schaffung eines Compilers, der Forth-Code nach C übersetzt und präsentierte erste, vielversprechende Ergebnisse.

Man erhofft sich auf diesem Wege von den Optimierungsmöglichkeiten moderner C-Compiler zu profitieren.

Sergei Baranoff las weiterhin den Vortrag von *Michael Gassanenka*, der leider nicht zur Tagung anreisen konnte. Dessen Paper befaßte sich mit einem Formalismus, mit dem man Returnstack-Manipulationen beschreiben und ihre Korrektheit beweisen kann.

Eine interessante Anwendung der binären XOR-Operation stellte *Dietrich Neubert* vor. Er definiert damit 'aktive Flags', die bei jedem Aufruf abwechselnd eine von zwei komplementären Funktionen ausführen und gleichzeitig die alternative Funktion für den nächsten Aufruf vorbereiten. Einen recht ausgereiften Compiler zur Definition von State Machine's und seine Anwendung diskutierte *John Pate* in seinem Vortrag und ich selbst berichtete schließlich über meine Versuche, 'Holon' für das Erlernen von Forth einzusetzen.

Neben den Vorträgen wurde in vier Workshops die Themenkreise *industrielle Netzwerke*, *Forth als Zwischensprache*, *Literate Programminig* und *objektorientierte Programmierung* diskutiert.

Für die nötige Auflockerung des Geistes sorgten eine abendliche Diskussionrunde mit 'Inpromptu Talks', ein Jazzkonzert am Samstagabend und ein kleiner Wettbewerb mit dem Thema: "Wie bringt man ein Forth-System auf besonders *elegante* Weise zum Absturz?" Dieses Problem wurde von *Egmont Woitzel* mit der folgenden, netten Sequenz am besten gelöst:

1 dup Base ! .

(Wer's nicht glaubt möge es probieren ...)

Die nächste EuroForth Tagung wird im Herbst 1996 in Sant Petersburg stattfinden. Langfristig zeichnet sich aber Schloß Dagstuhl als fester Austragungsort der Konferenz ab. Angesichts der diesjährigen Erlebnisse eine Entwicklung, die ich nicht bedauern kann. Also denn:

Auf Wiedersehen (in) Dagstuhl!



Neu in der KBBS-Box

Neuheiten in der KBBS / Forth-eV - Box seit dem 17.10.1995 17:08:44

Bereich public

- 11.11.1995 16:42:37 1708 public/linux/isdn/faxfilter.gz
- 16.11.1995 19:15:05 3533 public/linux/isdn/imontty-0.3.tar.gz
- 16.11.1995 19:15:48 289389 public/linux/isdn/isdn4linux0.7.0beta.tar.gz
- 11.11.1995 16:42:50 18288 public/linux/isdn/isdnlog-1.22.tar.gz
- 16.11.1995 19:15:51 21853 public/linux/isdn/isdnlog-1.24.tar.gz
- 16.11.1995 19:16:07 107663 public/linux/isdn/modules-1.2.8.tar.gz
- 11.11.1995 16:42:42 32668 public/linux/isdn/teles4isdn4linux-0.4g.tar.gz

Bereich CT (Heise-Verlag)

- 16.11.1995 08:30:03 534217 ct/ct9511.zip

Bereich faq (von mir ausgewaehlte News-Mitschnitte)

- 04.11.1995 12:25:12 4047 faq/Elektronik/805xx/serial-IRQ
- 18.11.1995 15:55:49 30784 faq/Elektronik/DCF77
- 12.11.1995 12:44:40 8075 faq/Linux/Shared-IRQ

Bereich FORTH

- 11.11.1995 11:09:47 56301 forth/vd/vd95-3.zip
- 21.11.1995 22:28:49 1543 forth/winforth/win32for.txt
- 21.11.1995 22:28:49 695043 forth/winforth/win32for.zip
- 21.11.1995 22:28:49 60 forth/winforth/win32in.bat

Arbeitsgruppe AG Marc 4

Das Direktorium hat die Gründung der Arbeitsgruppe zum Prozessor Marc 4 mittlerweile gutgeheißen [wir berichteten in *VD 3/95, S.34*]. Es liegen bereits erste Anfragen von Interessenten vor.

Wer Kontakt zur Arbeitsgruppe aufnehmen möchte, wende sich an:

AG MARC4 *Rafael Deliano*, Tel Fax 089/8418317

Netz:

Forth Online Resources List on Web

The Forth Online Resources List now has a Quick-Reference version on World Wide Web at

<http://www.complang.tuwien.ac.at/forth/forl.html>

thanks to the assistance of Anton Ertl. It may also be linked with from the FIG page at taygeta. The complete version is available in Forth Dimensions magazine.

Additions, changes and deletions may be sent to forl@artopro.mlnet.com

forl@artopro.mlnet.com (Forl) in /comp/lang/forth, oct '95

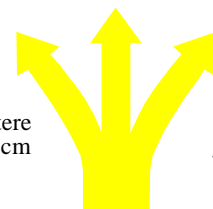
Hier wäre Platz für eine weitere Aktivität von genau 4,46 x 2,4 cm Größe gewesen. Nächstesmal?

WWW:

Yet another FAQ

I've given the old FAQs on implementations and books that I used to maintain a bit of an overhaul so that they are reasonably up to date I'm making them publically available via WWW (<http://panther.cs.man.ac.uk/~bevan/forth>) and as a one-off I'm posting them to comp.lang.forth. I am *not* volunteering to post them as monthly FAQs, though if someone else wants to do so I have no objection. However, I will be trying to keep the documents up to date so if you have any suggestions/corrections, please email them to me.

bevan@cs.man.ac.uk (Stephen J Bevan) in /comp/lang/forth Juli '95



Geschichte wird gemacht - es geht voran!



von Rafael Deliano

Steinbergstr. 37; D-82110 Germering

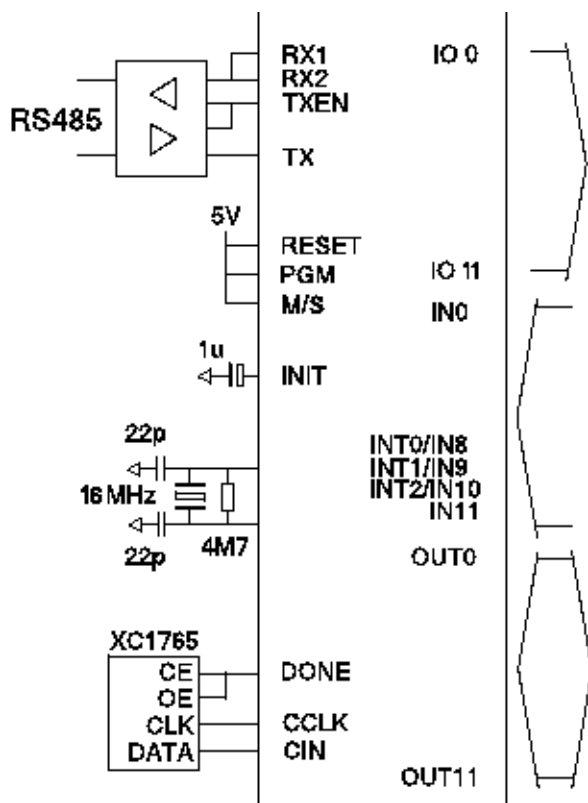
Nach dem Engagement am FRP1600 folgt ein weiterer Anlauf von DELTA t als Halbleiterschmiede. Durch klare Zielanwendung und einen kostenoptimierten Chip sind die Voraussetzungen für den kommerziellen Erfolg des neuen Feldprozessors diesmal günstig.

Stichworte: iX1 Prozessoren Feldbus DELTA t

Hardware

Der Controller ist in einem preiswerten 68 Pin PLCC untergebracht. Bereits der Standardtyp eignet sich für den

Temperaturbereich -40 ... +85 C. Die Versorgungsspannung beträgt 5V. Niedrigere Betriebsspannungen sind nicht vorgesehen. Durch die hohe Taktfre-



quenz von bis zu 24 MHz säuft das IC trotz CMOS wie ein Maurer: etwa 150mA bei den typischen 16 MHz. Für leitungsgespeiste Anwendung ist das etwas zuviel.

Es sind mehrere Betriebsarten wählbar. Im Singlechip Modus (Bild 1) ist die Außenbeschaltung minimal: Quarz, serielles EEPROM und typisch ein RS485-Treiber für den Feldbus. Für Anwendungen stehen dann drei Parallelports zu 12 Bit zur Verfügung. Einer davon ist bidirektional, die anderen sind je ein reiner Input- und Outputport. Das EEPROM ist hier immer nötig. Ähnlich wie ein Xilinx-LCA lädt der Prozessor aus ihm nach dem Reset den Programmspeicher. Typisch wird ein 8kByte-Typ XC1765 verwendet.

Man kann den IX1 auch als PeripherieIC in einem 8- oder 16 Bit-Mikroprozessorsystem einsetzen. Das EEPROM kann entfallen, wenn der Hauptrechner den IX1 über das serielle EEPROM-Interface lädt. Für den effizienten Datentransfer sind mehrere Varianten möglich. Insbesondere kann der IX1 ein DualPort-RAM simulieren. Während des Zugriffs muß die externe CPU jedoch durch ein Waitsignal gestoppt werden, weil der IX1 die Funktion in Software ausführt. Da man zwischen Intel- und MotorolaBusinterface umschalten kann, ist auch hier die externe Beschaltung minimal.

Architektur

Der IX1 ist ein Stackprozessor in Harvardarchitektur. Der Programmspeicher ist 4k x 12 Bit breit. Auch der Return- und Datenstack sind 12 Bit breit und je 32 Worte tief. Das erscheint überreichlich, da meist 16 Wort Tiefe

Bild 1: Der Singlechip-Controller iX1

Die Redaktion dankt der Firma DELTA t (Hamburg) für die freundliche Überlassung des Entwicklungssystems. Es würde uns freuen, wenn dieses Beispiel Schule macht und weitere Firmen ihre FORTH-Produkte für einen unabhängigen Test zur Verfügung stellen würden.

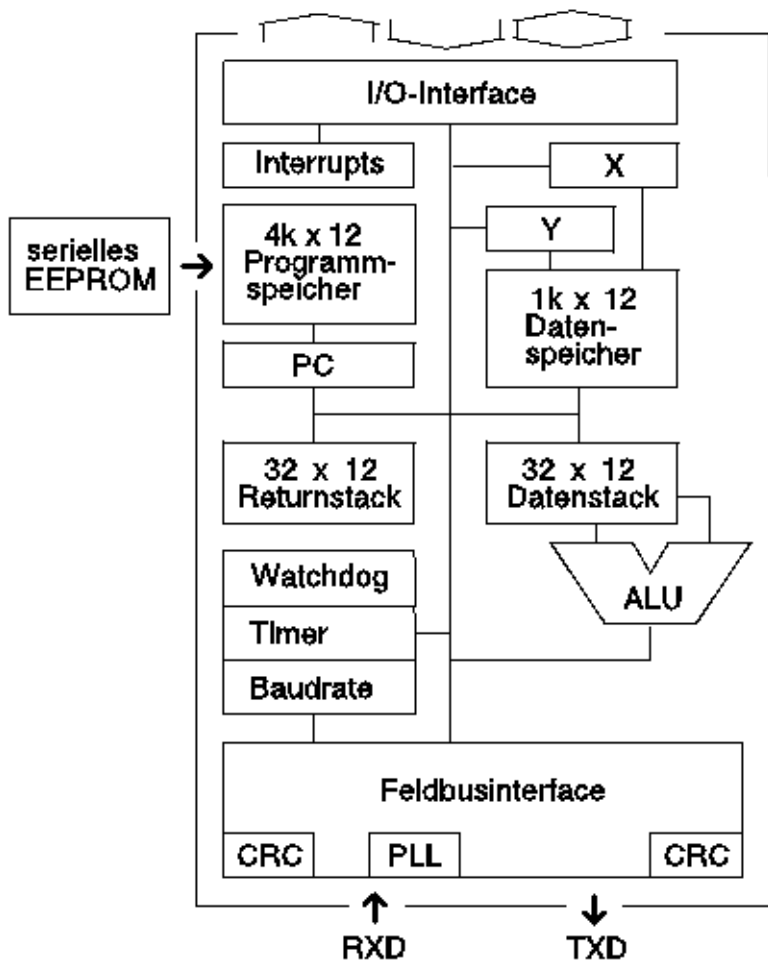


Bild 2: Blockschaubild des iX1

genügen. Der IX1 ist jedoch auf Multitasking mit 2 Tasks ausgelegt: die MainTask und die BitTask. Letztere kümmert sich um den Feldbus. Die Daten der beiden Tasks wachsen von oben und von unten in den Stack. Im Mittel bleiben für beide damit nur 16 Worte verfügbar. Schließlich gibt es noch einen Datenspeicher mit etwa 1k x 12 Bit. Auf ihn wird über die beiden Indexregister X und Y zugegriffen. Durch das reichlich vorhandene RAM kann der IX1 den Datenstrom aus dem Feldbus blockweise puffern.

Befehlssatz

Die meisten Befehle sind 6 Bit breit. Da einige Opcodes nur in BitTask oder MainTask verwendet werden können, haben sie die identische binäre Darstellung. Der Befehlssatz ist dadurch nicht

auf 64 Befehle beschränkt, sondern es gibt 67 Befehle. Lange Befehle haben neben dem 6 Bit Opcode einen 12 Bit Operanden. Trotzdem werden sie meist in einem Zyklus ausgeführt, da der Befehlsfluß in einer 24 Bit Instruction-Queue gepuffert wird.

Der 4k Programmspeicher enthält somit wesentlich mehr als 4k Befehle. Zieht man Eigenschaften wie die Wiederverwendbarkeit von Befehlen in FORTH in Betracht, oder die besondere Unterstützung von gefädeltem Code, so kann man im IX1 wohl genausoviel Software wie im 8k Speicher eines konventionellen Controllers unterbringen. Ein Befehlszyklus entspricht einem Taktzyklus des Quarzes. Typisch 63ns bei 16 MHz. Also DSP-Rechenleistung.

Die meisten Opcodes (vgl. Tabelle) entsprechen dem was man in einem Stackprozessor üblicherweise findet. Ein

Unterprogrammaufruf ist auch hier mit nur 2 Zyklen Verzögerung sehr schnell. FOR ... NEXT ist wieder die effizientere Alternative zu DO ... LOOP. Der Befehl INTERPRET dient der Verarbeitung von gefädeltem Code. Da der Programmspeicher im IX1 begrenzt ist, ist die Unterstützung von gefädeltem Code sinnvoll, damit man wahlweise besonders kompakte Programme erzeugen kann. Multiplikation und Division sind als die Schrittbefehle MULU und DIVU vorbereitet. Man kann damit die benötigte Wortlänge und damit die verbrauchte Zeit optimieren. Leider hat die CPU kein Overflowflag. Für 2er-Komplementarithmetik ist deshalb ein geeigneter Software-Patch nötig. Die unorthodoxe Datenwortbreite von 12 Bit entspricht der typischen Auflösung von A/D- und D/A-Wandler. Sie ist damit gut an Sensoranwendungen im Feldbus angepaßt.

Peripherie

Der Chip hat die übliche Controllerperipherie integriert. Verfügbar sind, außer Ports, ein Watchdog und ein Timer. Der Timer ist als programmierbarer 12-Bit Zähler ausgeführt. Sowohl Timer als auch Watchdog sind recht einfach aufgebaut, ihrer Aufgabe aber durchaus angemessen. Auf einem Stackprozessor sind Interrupts sehr effizient, weil man kaum Register sichern muß. Der IX1 macht deshalb reichlich Gebrauch von ihnen. Es sind 12 Interruptquellen vorhanden. Auslösen können BREAK-Befehle, Flanken an den Ports, Timer, Überlauf des Datenpuffers im RAM und der Feldbus. Nominell gelangt ein Interrupt innerhalb von etwa 5 Zyklen zur Ausführung. Die BitTask hat jedoch maximale Priorität und kann deshalb andere Interrupts verzögern. Sie ist normalerweise aber kürzer als 16 Befehle, wodurch die Verzögerung immer noch unter 1,5µs bleibt.

Feldbusinterface

Dieses komplexe Thema kann hier nur gestreift werden. Oberhalb der Bitebene erfolgt die Bearbeitung in Software und ist damit extrem flexibel. Es wurden eigens Opcodes dafür in den Befehlssatz integriert. Wo Software nicht möglich

DataStack

DROP (n ---)
 DUP (n --- n n)
 NIP (n1 n2 --- n2)
 OVER (n1 n2 --- n1 n2 n1)
 SWAP (n1 n2 --- n2 n1)
 TUCK (n1 n2 --- n2 n1 n2)
 UNDER (n1 n2 --- n1 n1 n2)

ReturnStack

RFETCH (--- n) R(n --- n) alias of R@
 NRFETCH (--- n1) R(n1 n2 --- n1 n2)
 variant of R@
 RDROP R(n ---)
 RPOP (--- n) R(n ---)
 RPUSH (n ---) R(--- n)

Arithmetic

ADD (n1 n2 --- n3) c alias of +
 SUB (n1 n2 --- n3) c alias of -
 DIVS (n1 --- n2) cR(n3 --- n3) divisionstep
 MULS (n1 n2 --- n1 n3) cR(n4 --- n4)
 multiplicationstep
 ADDI hhhhhh (n1 --- n2) c alias of hhhhhh +
 INC (n1 --- n2) c alias of 1+
 DEC (n1 --- n2) c alias 1-

Booleans

AND (n1 n2 --- n3)
 OR (n1 n2 --- n3)
 XOR (n1 n2 --- n3)
 ANDI hhhhhh (n1 --- n2)
 MASK hhhhhh (n1 --- n1 n2) AND-mask

Shift

LSR (n1 --- n2) c Logic Shift Right
 SL (n1 --- n2) c Logic Shift Left
 ROL (n1 --- n2) c Rotate Left
 ROR (n1 --- n2) c Rotate Right
 DROL (d1 --- d2) c ROL double
 DROR (d2 --- d2) c ROR double

Die Instruktionen des Feldbusprozessors IX1

ist, ist geeignete Hardware vorhanden. Sei es ein Baudrategenerator, eine programmierbare CRC-Berechnung für CRCs bis 16 Bit, oder eine digitale PLL im Empfangskanal. Die Obergrenze des Durchsatzes liegt typisch bei 500k bis 1M Bit/sec. Und dabei bleiben meist immer noch 50% der Rechenleistung für Anwendungsprogramme verfügbar.

Memory

LD (addr --- n) alias of @
 LDD hhhhhh (--- n) alias of hhhhhh @
 ST (n addr --- n) alias of !
 STD hhhhhh (n ---) alias hhhhhh !
 LIT hhhhhh (--- n)
 LDCM (-- n) R(addr ---) @ ProgramMemory
 Harvard-Patch
 IXLD (--- n) @ indirect via IX
 IXST (n ---) ! indirect via IX
 IYLD (--- n) @ indirect via IY
 IYST (n ---) ! indirect via IY

Compares

CMPI hhhhhh (n --- flag) c Compare
 ZEQU (n --- flag) 0=

Jumps

CALL hhhhhh (---) R(--- addr)
 RET (---) Return from CALL
 QRET (Flag ---) Return if Flag set
 BREAK (---) SoftwareInterrupt
 INTRET (---) Return from Interrupt
 BRA (---) Branch always
 BCC (---) c Branch on Carry Clear
 BTOS (n ---) Branch if n = 0
 used in IF, UNTIL,
 DBR (---) R(n1 --- n2) bzw R(r1 ----)
 used in FOR ... NEXT

Fieldbus

BITSTUFF (---) c
 RXBIT (n1 --- n2) c
 TWIST (n1 --- n2)
 RXCRC (---) c
 TXCRC (---) c
 BITRET (---) Switch to MainTask
 BCRC (---) Branch on CRC
 BSC (---) Branch on StuffCounter
 BUC (---) Branch on User Clear

Misc

HALT (---) stop CPU
 INTERPRET (---) bzw (--- addr) R(addr ---)
 support for threaded code
 CC (---) c Complement Carry
 SC (---) c Set Carry
 NOP (---)

Dokumentation

Gut und reichlich vorhanden. Der Text ist überwiegend in Englisch gehalten. Geordnet nach Themenschwerpunkt erfolgte eine Aufteilung in 7 spiralgebundene Hefte im A4-Format. Die Schrift ist dadurch angenehm groß und lesbar geblieben. Die Hefte sind dünn und handlich. Auffallend ist, daß soweit

Steckbrief:

- * 12 Bit - Stackprozessor: FORTH in Silizium
- * primäre Anwendung: Feldbusse und andere serielle Protokolle.
- * Firmware für diverse Feldbusse verfügbar.
- * Rechenleistung: 16 - 24 MIPS
- * SingleChip-Controller: IC enthält 4k Programm & 1k Datenspeicher sowie Feldbusinterface, Watchdog, Timer, Ports.
- * Alternativ einsetzbar als: Peripheriebaustein in Mikroprozessorsystem
- * Quasi-OTP: Programmspeicher wird aus externem seriellen EEPROM geladen
- * Preis: bald <50DM / 100 St.

Bezugsquelle:

DELTA t GmbH
Adenauerallee 54
20097 Hamburg
Tel. 040 / 280 152-0
Fax: 040 / 280 152-90

irgend möglich die Erwähnung des gefürchteten F-Wortes vermieden wurde. In den Mnemonics hat man sicherheitshalber sogar @ und ! in LD und ST umbenannt.

Entwicklungssystem

Das Demoboard enthält außer dem IX1 kaum ICs. Der IX1 hat einige Portpins zusammen mit den seriellen Leitungen des EEPROMs auf einer Steckerleiste herausgeführt. Diese führt über ein Verbindungskabel zum Parallelport des PC. Die Signalumsetzung im Kabel erfolgt durch einen ST62xx-Controller, der auch als Dongle dient. Will man auf die Portpins nicht verzichten, kann die Kommunikation auch über den Feldbus und die V24 realisiert werden. Ins Anwenderprogramm muß in der Testphase ein Debugprogramm integriert werden. Es benötigt etwa 200 Speicherzellen. Dann ist FORTHartiges, interaktives Arbeiten möglich. Zusätzlich zum isolierten Testen der FORTH-Befehle kann man ihr internes Verhalten mit einem Singlestepper ver-

folgen. Die Definition eines neuen Befehls resultiert in kompletten Nachladen des Programmspeichers. Bei einer Harvardmaschine kann man keine Software im DatenRAM ausführen. Offensichtlich ist sowas kein Echtzeit-In-Circuit-Emulator. Aber es wäre ohnehin keine Freude bei einer 60nsec-Maschine mit Flachbandkabel kontaktieren zu wollen. So kann man die Stiftleiste des Downloadkabels ins eigene Board integrieren und dann dort wie gewohnt testen.

Programmiert wird der IX1 in Assembler oder FORTH. Das FORTH ist ein ANS-Subset, aber nicht völlig kompatibel. Man kann beim Compilieren einen Peephole-Optimierer zuschalten. Auch ein Decompiler ist vorhanden. Erweitern kann man es durch das getrennt erhältliche Arithmetikpaket, das auch eine Float enthält. Die Software läuft auf PC-AT ab MS-DOS 3.0. Neben dem Parallelport wird durch den Feldbus meist auch noch die serielle Schnittstelle benötigt. Die Oberfläche stützt sich auf Menüs und Functionkeys und ist dadurch effizient und benutzerfreundlich.

Angekündigt ist ein C-Compiler unter Windows. Nominell verlangt der Markt das zwar. Ob der Anwender damit auf einem Stackprozessor glücklich wird ist eine offene Frage.

Preise

Das Entwicklungssystem (HW, SW & Dokumentation) kostet zusammen mit einem implementierten Feldbusprotokoll 5000DM. Abgemagerte Versionen für Hochschulen liegen bei 3000 bzw. 2500DM. Der Preis für Protokolle variiert. P-Net kostet 25DM und ist PublicDomain. Es wird typisch zur Vernetzung von Sensoren, d.h. für billige Endprodukte eingesetzt. Die Version für Interbus oder Profibus sind (bald) zertifiziert und enthalten eine Lizenz für unbegrenzte Stückzahlen. Kostet dann aber auch bis 8000DM. Diese Netze haben als Anwendung aber z.B. Werkzeugmaschinen. Der Preis fürs Protokoll orientiert sich also am Preisniveau des Endprodukts. Protokolle werden prinzipiell in kompilierter Form geliefert und können vom Anwender nicht mehr modifiziert werden. Letzteres

würde ja auch die Zertifizierung hinfällig machen.

Das derzeit verfügbare Vorserien-IC kostet einzeln 140DM. Es entspricht in Details noch nicht dem Sollzustand und einige Funktionen müssen derzeit noch durch Software nachgebessert werden. Ab Anfang nächsten Jahres sind Serientypen verfügbar und der Preis liegt zwischen 50 - 100DM abhängig von der Stückzahl. Ende 96 sinkt der 100 Stück-Preis deutlich unter 50DM. Das IC wird von SGS-Thomson gefertigt. Außer für Großanwender erfolgt der Vertrieb jedoch exklusiv durch DELTA t.

Fazit

Das IC ist nicht billig, aber kosteneffektiv. Es minimiert die externe Beschaltung und ermöglicht durch seine hohe Rechenleistung Hardware durch Software zu ersetzen. Es minimiert auch die Softwareentwicklungskosten. Hier kann ohne große Einbußen an Effizienz in Hochsprache entwickelt werden wofür konventionell Assembler nötig ist.

Die Anwendung ist nicht auf Feldbusse beschränkt. Weitere, noch unerschlossene, Einsatzgebiete finden sich in der Telekommunikation. Weniger in ISDN als bei den inkompatiblen Protokollen der digitalen Altlasten. Oder auch bei Massenspeichern in Nischenanwendungen wie Streamern. Da der Materialeinsatz im Entwicklungssystem gering ist, bestehen auch Chancen, daß hier noch die Preise sinken und sich die Zahl der Anwender dadurch erhöht.

Literatur:

- K. Schleisiek-Kern, H. Pfüller "Das Feldbus-Chamäleon" Franzis-"Elektronik" 20/1992 S.50
- L. Brackmann "Fünfkämpfer" ELRAD 1/1995 S.21

Forth-Systeme: 486-Assembler für F-PC

The assembler I wrote for Andrew and Tom's FORTH for Windows 95 and NT is now available via anonymous ftp. To get it, ftp to `ftp.netcom.com', login as anonymous, and get these files:

```
/pub/ja/japs/486read.me  
/pub/ja/japs/486asm.zip
```

The file 486read.me contains a very brief description of the archive contained in 486asm.zip, a PKZipped DOS archive. I was working on a Linux port of FORTH for ..., but I lost interest when I realized how Win32 dependant the kernel was. Still, I'll put the sources (tarred and gzipped) in my ftp directory when I get around to it.

By the by, does anyone know if FORTH for Windows 95 and NT is available anywhere via anonymous ftp? If not, I'll get in touch with Andrew and Tom about sticking it in my ftp directory.

*japs@netcom.com (Jim Schneider) in
comp/lang/forth, nov'95*

Forth-Systeme: 32-bit protected Mode Forth

In article <48332e5da5@nntp.crl.com> Ed Tannenbaum <et@crl.com> writes:

"Can anybody recommend a 32 bit for DOS? My applications have outgrown the 16 bit environment, and EMS is slowing me down too much. I am doing image processing on some large files."

LMI has been selling a 32-bit protected mode Forth for DOS since 1987. It is called 80386 UR/FORTH and it is a complete development system including multitasker, block editor, assembler, software and hardware floating point support, native code compiler, and so on. The 80386 UR/FORTH system is built on the Phar Lap TNT DOS Extender and is fully compatible with all memory managers and even Windows 3.1. All LMI Forth systems include a full host file interface and can compile from either block files or ASCII text files.

For more information call Tammy at 310-306-7412 or FAX 310-301-0761.

You can also leave a message on our BBS at 310-306-3530 or send EMAIL to lmi@cerf.net or sales@lmi.la.ca.us.

*duncan@nic.cerf.net (Ray Duncan) in
comp/lang/forth, nov'95*



Serie: PC-Meßtechnik, Teil III

Der A/D-Wandler am Druckerport

Autor: Klaus Kohl / BMC Systeme GmbH
 Zeppelinstr. 10; 86406 Mering; Tel.: 08233 / 30 524

Nach dem Einführungsartikel und dem im letzten Teil vorgestellten Interrupttreiber kommen wir jetzt zur PC-Hardware. Anhand eines A/D-Wandlers am Druckerport wird die Verwendung dieser auch bei Laptops verfügbaren Schnittstelle zur Datenerfassung gezeigt. Der Artikel beschreibt dazu neben dem Aufbau des PC-Druckerports auch den Aufbau und die Programmierung des kommerziell verfügbaren Meßsystems μ Meter4.

Stichworte: Meßtechnik, PC, A/D-Wandlung, Druckerport

1. Der PC-Druckerport

Meistens wird die 25polige Buchse des Centronics-Port direkt mit dem Drucker verbunden. Jedoch kann man diese Schnittstelle auch als reinen Digitalport ansehen, der über eine Reihe von Ein- und Ausgängen verfügt. Die Pegel sind zwar nicht so genau definiert, liegen aber meist bei 5V (bei neueren PCs manchmal nur bei 3.3V). Neuere Schnittstellenkarten haben sogar die Möglichkeit zur Umschaltung der 8 Datenleitungen als Eingang. Da dies aber nicht immer garantiert ist, wurde es hier nicht genutzt. Eine zusätzliche Option des Druckerports ist die Auslösung eines Interrupts durch einen externen Impuls. Dies wird hier auch nicht verwendet, könnte aber für andere Meßsysteme z.B. zur Erzeugung eines externen Timerinterrupts für "gerade" Frequenzen herangezogen werden.

1.1. Pinbelegung

Folgende Liste zeigt die Maximalbelegung eines Druckerports. Bei den Leitungen, die mit einem "*" gekennzeichnet sind, ist das Signal Low-Aktiv und deshalb im Ruhezustand auf 1 (entspricht 5V). Zusätzlich muß man bei der Programmierung darauf achten, daß einige Portleitungen über Inverter gehen und deshalb

der im Programm verwendete Wert (1 oder 0) umgekehrt zu Status der Leitung ist.

1.2. Speicheradressen

Beim Bootvorgang sucht der PC automatisch auf drei Adressen (\$3BC, \$378 und \$278) nach einem Druckerport. Damit aber intelligente Karten weitere Druckerports installieren und Anwenderprogramme diese Adressen ermitteln können, gibt es in den BIOS-Variablen vier Speicherstellen

Pin	Richtung	Verwendung
1	Ausgang*	/Strobe Ein Low-Impuls zeigt an, daß Daten verfügbar sind
2-9	Ausgang*	Datenbit 0-7
10	Eingang	/Acknowledgment Ein >0.5us dauernder Impuls zeigt an, daß der Drucker bereit ist
11	Eingang	Busy Ein High-Pegel an dieser Leitung zeigt an, daß der Drucker blockiert ist
12	Eingang	Paper End Bei einem High-Pegel hat der Drucker kein Papier mehr
13	Eingang	Select Status Bei einem High-Pegel ist der Drucker angewählt
14	Ausgang	/Auto Feed Bei Low-Pegel erzeugt der Drucker automatisch bei CR ein Line-Feed
15	Eingang	/Error Bei Low-Pegel besteht ein Fehler am Drucker(Papier, Offline ...)
16	Ausgang	/Initialize Ein Low-Pegel von mindestens 50us initialisiert den Drucker
17	Ausgang	/Select (nicht immer vom Drucker genutzt) Mit einem Low-Signal wird der Drucker angewählt
18-25		Masse Im Extended Mode werden diese Pins als Eingänge verwendet

Tabelle 1: Belegung der Centronics Schnittstelle

für die Basisadresse der vorhandenen Schnittstellen. Nicht verfügbare Ports werden mit dem Wert 0 gekennzeichnet.

Adresse der

BIOS-Variable Verwendung

- \$0040:\$0008/9 Basisadresse von Druckerport 1 (meist \$0378)
- \$0040:\$000A/B Basisadresse von Druckerport 2 (meist \$0278)
- \$0040:\$000C/D Basisadresse von Druckerport 3
- \$0040:\$000E/F Basisadresse von Druckerport 4

Bei jedem angegebenen Druckerport werden bis zu 8 Adressen für die Programmierung belegt. Davon sind aber nur 3 hier beschrieben, da über sie die gesamte Schnittstelle gesteuert wird. Man muß darauf achten, daß beim Schreiben und Lesen eines Ports die Bits manchmal unterschiedlich belegt sind. Beim Lesen des Werts der Datenleitungen (Offset 0) erhält man den tatsächlichen externen Status und kann dadurch z.B. Kurzschlüsse feststellen.

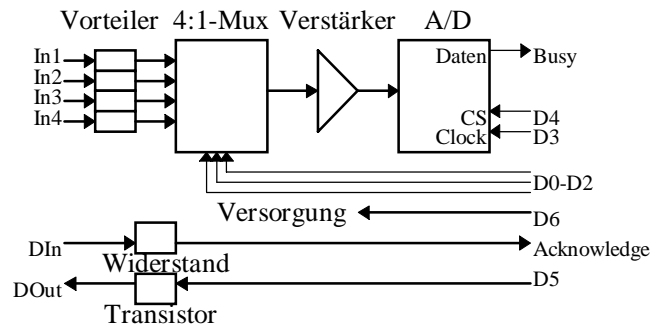
Offset	Richtung	Bit	Bedeutung
0	Schreiben	0-7	Wert an den Datenleitungen Pin 2-9
0	Lesen	0-7	Wert von den Datenleitungen Pin 2-9
1	Lesen	0/1	nicht verwendet
		2	0: Interrupt steht an (Acknowledge empfangen)
		3	Invertierter Status der /Error-Leitung (Pin 15)
		4	Status der Select-Leitung (Pin 13)
		5	Status der Paper-End-Leitung (Pin 12)
		6	Status der /Acknowledge-Leitung (Pin 10)
		7	Invertierter Status der Busy-Leitung (Pin 11)
2	Lesen	0	Invertierter Status der /Strobe-Leitung (Pin 1)
		1	Invertierter Status der /Auto-Feed-Leitung (Pin 14)
		2	Status der /Initialize-Leitung (Pin 16)
		3	Invertierter Status der /Select-Printer-Leitung (Pin 17)
		4	Bei 1 ist Interrupt durch /Acknowledge-Leitung möglich
		5-7	nicht verwendet
2	Schreiben	0	/Strobe-Status (invertiert zum Pin 1)
		1	/Auto-Feed-Status (invertiert zum Pin 14)
		2	/Initialize-Status (zum Pin 16)
		3	/Select-Printer-Status (invertiert zum Pin 16)
		4	1: Interrupt aktivieren
		5	1: Pins 2-9 sind als Eingänge verwendet
		6/7	nicht verwendet

2. Das Meßsystem µMeter4

Bei dem hier vorgestellten Meßsystem handelt es sich um ein an den Parallelport anschließbares Dongle, der über folgende Eigenschaften verfügt:

- 4-Kanal-Multiplexer (100KOhm Eingangswiderstand)
- 10Bit-A/D-Wandler
- Einen durchgeschleiften Eingang
- Einen Open-Collector-Ausgang

Durch Verwendung anderer Multiplexer und eines genaueren A/D-Wandlers kann natürlich auch ein 8Kanal-12Bit-Wandler gebaut werden (unter dem Namen µMeter8 verfügbar). Man hat aber dann das Problem, daß der Strombedarf nicht mehr durch nur ein Datenbit gedeckt



werden kann. Die Geschwindigkeit des A/D-Wandlers ist meist höher als die Auslesegeschwindigkeit, da daß Toggeln der Bits zum Auslesen der Daten auch auf einem 100MHz-Pentium-Prozessor sehr lange dauert.

Damit ein evtl. angeschlossener Drucker nicht wahllos Zeichen ausgibt und/oder bei späterer Verwendung auch reagiert, wurde auf die Umprogrammierung des Druckerports verzichtet. Statt dessen wird die Steuerung nur über die Datenleitung realisiert.

Bit Bedeutung der Datenleitungen (Offset 0)

7	Reserviert (auf 0 lassen)
6	Versorgung des µMeter4 (muß immer auf 1 sein)
5	Steuerleitung zum Transistor des Digitalausgang (invertiert)
4	Chip Select des A/D-Wandlers (Low-Aktiv)
3	Clock für den A/D-Wandler
2..0	Multiplexer für die Eingangskanäle
%000:	Eingang 1
%001:	Eingang 2
%010:	Eingang 3
%011:	Eingang 4
%1xx:	Reserviert (Bit 2 deshalb immer auf 0 lassen)

Über die Eingänge für Busy und Acknowledge werden dann die Daten des A/D-Wandlers bzw. der Status des Digitaleingangs abgefragt. Eine weitere, hier als "reserviert" bezeichnete Leitung erlaubt dem mitgelieferten Meßprogramm NextView die Kontrolle, ob das µMeter4 oder das µMeter8 verwendet wird.

Bit Bedeutung der Statusbits (Offset 1) als Druckerport bei µMeter4

7	Busy	invertierte A/D-Datenleitung
6	Acknowledge	invertierter Digitaleingang
5	Paper fail	für µMeter-Erweiterungen reserviert
4	Select-Signal	nicht verwendet
3	Error-Signal	nicht verwendet
2..0	reserviert (nicht durch PC unterstützt)	

Ein in den Anleitungen der µMeter abgedruckten FORTH-Listing beschreibt sowohl die Grundroutinen zur Abfrage des A/D-Wandlers und des Digitaleingangs als auch ein Wort zum Setzen des Digitalausgangs. Meist muß vor



Verwendung des Programmes noch das Timing in DELAY angepaßt oder die Schnittstellenadresse umgestellt werden.

- 21µs vor nächsten Zugriff auf A/D-Wandler warten (nächste Wandlung läuft)

\ Variablen und Konstanten (21.07.94/KK)

\$0378 Constant base \ Basisadresse für LPT1
Variable a \ Zustand des Ausgabeports

```
: delay ( u -- ) \ u Mikrosekunden warten
  0 DO LOOP ;
```

\ Grundroutinen (21.07.94/KK)

```
: init ( -- ) \ Initialisierung des Druckerports
  $60 a !
  $60 base pc! ;
```

```
: getad ( kanal -- mw )
  a @ $f8 and + 1- a !
  \ Kanal in Ausgabewert einbinden
  a @ base pc!
  &500 delay
  a @ $10 + base pc!
  a @ base pc!
  &21 delay
  0
  &10 0 DO
    2*
    base 1+ pc@ $80 and 0=
    IF 1+ THEN
      a @ $08 + base pc!
      a @ base pc!
  LOOP ;
```

```
: getdi ( -- flag )
  base 1+ pc@ $40 and
  IF false ELSE true THEN ;
```

```
: setdo ( flag -- )
  a @ $20 or
  swap IF $df and THEN
  dup a !
  base pc! ;
```

Der Ablauf einer A/D-Abfrage:

Der eigentliche Ablauf für eine Einzelmessung ist wie nachfolgend beschrieben. Das Ergebnis im 16Bit-Wert mw ist eine 10Bit-Zahl (Bereich 0..1023), wobei 0 den Spannungswert 0.00V und 1024 den Spannungswert 10.00V repräsentiert.

- Wahl des gewünschten Analogeingangs (Bit 0..2, Rest unverändert lassen)
- ca. 500µs zur Stabilisierung der Spannung warten
- Chip-Select setzen und löschen (Bit 4, startet die Wandlung)
- 21µs bis zum Abschluß der Wandlung warten
- Wert auslesen
 - Wert 0 vorgeben
 - Folgende Schleife 10 mal wiederholen:
 - Wert durch linksschieben verdoppeln (Bit 0 = 0)
 - Wert erhöhen, wenn Busy-Bit gelöscht ist
 - Clock-Leitung setzen und sofort wieder löschen (nächste Datenbit anfordern)

3. Weitere Möglichkeiten des Druckerports

Natürlich sind viele Verbesserungen sowohl bei Aufbau als auch bei der Programmierung des µMeter möglich. Jedoch funktioniert der hier vorgestellte Ansatz und stellt damit eine verwendbare Lösung für fast alle PCs dar. Da das Ziel dieses Artikels eine ausführliche Beschreibung des Druckerports ist, wurde der Aufbau des Meßsystems (auch wegen rechtlicher Gründe) nur schematisiert dargestellt. Ich werde aber in kommenden Folgen dieser Serie aber noch selbst zu bauende Meßsysteme und Sensoranschlüsse vorstellen.

Außer dem Anschluß eines A/D-Wandlers sind mit einem Druckerport viele andere, zum Teil in dieser Zeitschrift schon angesprochene Anwendungen möglich. Einige Beispiele werden wir nach den Einführungsartikeln noch in dieser Serie kennenlernen.

- Ausgabeeinheiten:
 - 8*8-LED-Matrix
 - Schrittmotorsteuerung
 - (optoentkoppelte) Ausgänge mit Relais und Thyristoren
- Eingabeeinheit:
 - Selbstgebauten A/D-Wandler nach dem Joystick-Prinzip
 - (optoentkoppelte) Eingänge (bis 24 V)
 - Anschluß von Sensoren mit digitalen Ausgängen (z.B. pulswertenmoduliert Temperatur- und Lichtsensoren)
- Schnittstellen zu externen Systemen (nicht nur Meßtechnik)
 - Interface zu IEC, SCSI, Ethernet und anderen Bussystemen
 - Downloader für EPROM-Simulator
 - Parallel angeschlossene Digital-Oszilloskope
 - Die Background-Debugger-Schnittstelle zu Motorola 68XXX-Prozessoren

4. Literatur zur Programmierung des Druckerports

- (1) c't-Serie PC-Bausteine
Details über die Centronics-Parallel-Schnittstelle
c't 1988, Heft 6, Seite 166ff
- (2) Thom Hogan
Die PC-Referenz für Programmierer
Microsoft Press
- (3) Frank van Gilluwe
The undocumented PC
The Andrew Schulmann Programming Series
Addison-Wesley Publishing Company

5. Bezugsquelle für Meßsysteme µMeter4/8

Dr. Schetter BMC IGmbH	Conrad Electronic GmbH
Boschstr. 10	Klaus-Conrad-Straße 1
82178 Puchheim/München	92240 Hirschau
Tel. 089/8002403	Tel. 0180/5312111



VD quo vadis?

Ein Jahr Forth-Magazin aus der Hauptstadt

von Claus Vogt

clv@forth-ev.de; Ebersstr.10; D-10827 Berlin; 030/782 81 79

Dies ist die vierte Ausgabe der VD aus Berlin. Ein kompletter Jahrgang bieten einen guten Zeitraum für eine Zwischenbilanz: Was waren die Ziele, was wurde erreicht? Welche Probleme müssen wir 1996 angehen, um im Jahre 2000 nicht als eine Gruppe überalterter Jugendlicher mit einem antiquariatsreifen Vereinsblatt dazustehen. Zum Mitstreiten und Mitdiskutieren laden wir alle Jungen und Altgebliebenen auf: Stellungnahmen bitte ins Brett de.com-p.lang.forth und druckreife Beiträge zum Leserforum direkt an Claus

Stichworte: Forth-Magazin

Das verflixte zehnte Jahr...

Der Jahrgang 1994 gehört sicher zu den interessantesten der VD. Im Heft 1/95 'Alteditor' Rolf Kretzschmar vor, wie ein Schwerpunktheft auch ohne den üblichen Corpus von separater Schwerpunktreaktion und explizitem Konzept machbar ist.

VD 2/95 wurde von Jens Axel Wilke zusammengetragen. (Un-)vorhersehbare Einflüsse verzögerten die Fertigstellung, bis

das Direktorium im Herbst die Endproduktion übernahm. Das Heft strahlt mit vielfältigsten Vorschlägen vieler Personen viel Optimismus aus.

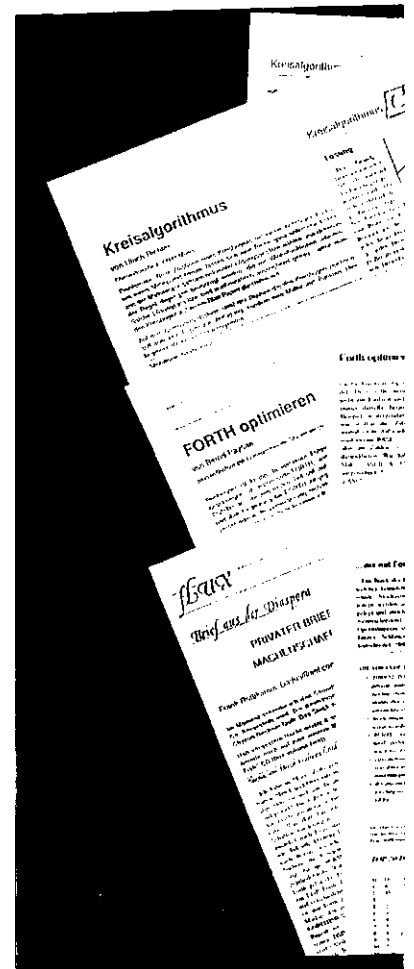
VD 3+4/95 ist von Umfang und Inhalt eines der stärksten Hefte überhaupt. Das Direktorium höchstselbst brilliert durch Fachkompetenz gepaart mit hochkarätigem Humor. Einzige offenkundige Ente: Auf dem Titelblatt steht die Jahreszahl '1994'.

Idee und Startloch

Die Verspätung währte und währte. e Claus Vogt - designerter Editor für- brütete Utopien aus und wartete Seine Vorstellung im Heft 3+4/95, Seite 2, läßt die zukünftige Richtung bereits ahnen: Grafische Gestaltung / Witz / fachliche Öffnung / moderne Kommunikationstechnik / keine explizite Neuorientierung - dazu die Werbekampagne 'Echtes Forth', die unschwer als Persiflage der Titanic-Kampagne 'Echtes Geld' zu identifizieren ist. Während die kommenden AutorInnen noch identifizierten statt ihre Artikel auf den Weg zu bringen, saß Claus an ganz anderen Problemen. Immerhin hatte der Editor auch die Layouterstellung zu tragen - eine massive Mehrbelastung. Gestützt auf langjährige Erfahrungen mit Schüler- & Studentenzeitungen, linksradikalen Pamphleten und einer Reihe von Originaldokumenten der DDR-Opposition wußte Claus, welches Layoutkonzept mit seinen beschränkten personellen Resourcen sinnvoll war: Gar keines! Die Kraft liegt in den Konzepten, die äußere Form ist Nebensache.

DTP in TV?

Ich wählte das nächstliegende bekannte Programm WinWord. Im Problemfall wäre



ich sofort auf Papierumbruch umgestiegen. Bloß kein Einarbeitungsaufwand, denn von DTP hatte ich (damals) keinen blassen Dunst. Leider ist der Job des Editors der VD ein sehr einsamer. Wird der Leser vom 'lonely Ranger' angerufen, nutzt er die Gelegenheit verständlicherweise, um dem Editor seine guten Vorschläge und sein Auffassung über hohe Anzeigenpreise (80.-DM) loszuwerden. Der Editor träumt von endlosen Prärien ohne Telefon und ihre Fernzone-III-Tarife.

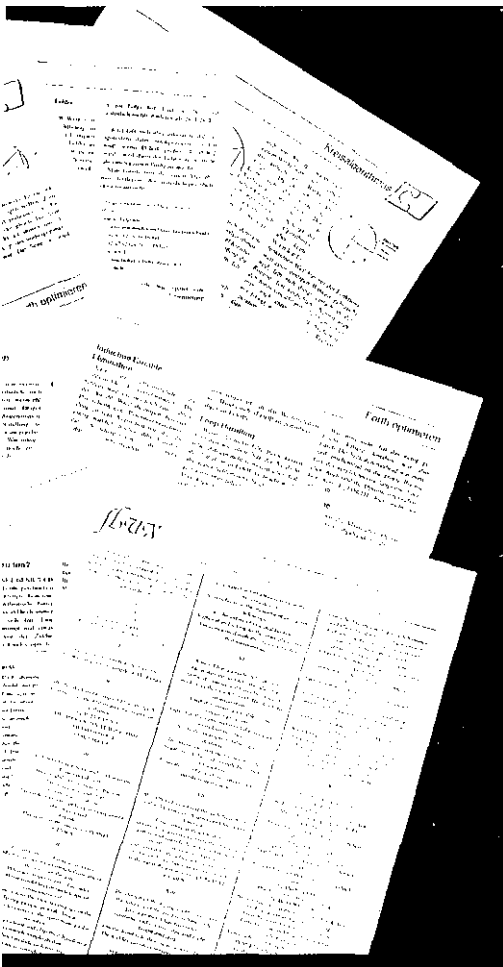
Alle möchten Rolfs aufwendige Layouts behalten. Na gut. Einige verzweifelte Nächte später Umstieg auf das nächstliegende 10.-DM-DTP-System namens Pressinternational - besser bekannt als TimeWorks-Publisher, dessen Nachfolger PressWorks seine treuen Dienste und untreuen Abstürze bis heute in der Redaktion zelebriert. Es kostete 50.-DM und ist nebenbei die einzige Investitionsgüteranschaffung der Redaktion (natürlich Privatschatulle, wenn man mal vom Duden (39.-DM, Famila) absieht.

Die vornehmste Aufgabe

Was ist die vornehmste Aufgabe des Editors der VD? Es ist der Spagat zwischen den berechtigten hohen Ansprüchen der Vereinsmitglieder und ihren (und vor allem seinen) knappen personellen und finanziellen Ressourcen. Sein Ziel muß sein, in diesem Paradoxon seine persönliche Substanz unbeschadet zu erhalten. Sein Ziel ist nicht, das geschilderte Dilemma lösen zu wollen. Nie macht er die Ziele der Mitglieder zu seinen eigenen. Immerhin sind sie mit 300:1 in der Mehrheit, da der eine vor allem an sich selbst denken.

- Daher sollte das Heft
- pünktlich erscheinen,
 - regelmäßige Kolumnen enthalten,
 - den Kontakt unter den Lesern fördern,
 - Vielfältigkeit und offen für alle Themen sein,
 - lesbar und hübsch anzugucken sein,

Dahinter steckt des Editors einziges Ziel: 'Editor, wenn Dir Dein Leben lieb ist, dann organisiere alle Arbeit von Dir weg!'. Also, Leser, beteilige auch Du Dich! Geht dabei etwas schief ist das nicht schlecht, sondern gut, weil es eine Rückkopplung anstößt, die noch mehr Mitarbeit (und noch mehr Fehler) produziert.



Die Highlights 1995

Ulrich Richter begeistert mit seinem 'Kreisalgorithmus' (vd 2/95, S.5) alle gleichermaßen: Profis auf der Suche nach nützlichen Programmen, Hobbyisten mit numerischen Interessen und Menschen im Ausbildungssektor

Bernd Paysans Auftakt zur Serie 'Forth optimieren' (VD 3/95, S.21) zeigt, daß auch die hohe Kunst des Programmierens einfach, übersichtlich und verständlich dargestellt werden kann.

'Laurie Anderson und Forth in einem Atemzug!' lesen wir bei Frank "Genius" Rothkamm in 'flux' (VD 3/95, S.24), den seine Schaffenswanderungen mit seiner Planet-Genius-CD Vol.I auf Platz 5 der Toronto Charts gebracht haben. Wer sich die CD kaufen möchte: Es handelt sich um *'experimentelles, mit Samples behaftetes Ambient Trance'*. Eine kurze Rezension bitte an die VD.

Realität

Es wurde ein fester Mitarbeiterstamm gewonnen, sicher der wichtigste Erfolg des Jahres 1995. Die Beteiligung an den Kurzmeldungen ist allerdings noch immer zu gering. Offenbar ist die Hemmschwelle immer noch zu hoch.

Die leidige Technik

Die Redaktionstechnik ist kommunikativ mit Fax/Anrufbeantwortermodem (vgl. VD 1/95, S.28) ausreichend ausgestattet. Die Vernetzung der zwei Redaktionsrechner (386sx16-4 und 386dx33-4) hat zwar viel Arbeit gekostet, aber wenig gebracht. Die Layoutvorbereitung von Grafikdateien steht jetzt (mit Macken). Ihr könnt jetzt also mehr Illustrationen zu Euren Artikeln als Datei schicken. Gute Papiervorlagen gehen auch und sollten möglichst immer beiliegen. Ich kann sie scannen. Letztlich hat sich der Aufbau der Technik - allen Technikfreaks zum Trotz und den Arbeitspsychologen zur

Ehre - als organisatorisches Problem herausgestellt und muß auch als solches angegangen werden.

Ehrenamt halbtags

Meine Arbeitszeit für die VD beträgt derzeit etwa 20 Stunden pro Woche, die sich zu gleichen Teilen auf die drei Bereiche

- Layout
- technische Umgestaltung
- Autorenkontakt, Verwaltung

Diese Arbeitszeit muß massiv sinken.

Das Layout der Kurzmeldungen muß möglichst automatisch werden. Die Artikel erfordern weniger Aufwand. Hier fehlen vor allem Grafikprogramme. Bei der 'technischen Umgestaltung' brauche ich massive Hilfe, nicht nur von Spezialisten.

Als Zielvorstellung bis Ende 1996 würde ich sagen: Insgesamt maximal 10 Stunden pro Woche, zu gleichen Teilen in Layout, Autorenkontakt und konzeptueller Arbeit. □

Folgende Personen arbeiten verantwortlich an der VD mit:

- Ulrike Schnitter (Anzeigenverwaltung)
- Fred Behringer (Kolumne 'Forth International' und Ressort 'Internationales')
- Ulrich Hoffmann (Kolumne 'ANS Forth')
- Rafael Deliano (fester Autor mit Spezialgebiet 'Hardware-Konzepte, -implementierung, Geschichte')
- Friederich Prinz (fester Autor z.B. für Lokales, Anfänger und Z-Netz)
- Rolf Kretzschmar (Zeichnungen)
- Birgit Steffenhagen als Korrektorin

Es laufen folgende Artikelserien:

- 'Forth optimieren' von Bernd Paysan
- 'Meßtechnik mit dem PC' von Klaus Kohl

Es gibt folgende Rubriken:

Wenn wir Inhaltsverzeichnis, Titelblatt und Anzeigen mal herauslassen, sind die festen Rubriken:

- Leserforum,
- Korrektur,
- Autorengeschenke,
- Termine (unregelmäßig),
- Adressen/Ansprechpartner,
- Forth-Gesellschaft (schlecht unterstützt!"),
- Bücher, Rezensionen, Presseschau,
- Was noch? Unernthaftes etc..

und die sich herauskristallisierenden Info-Rubriken:

- Forth-Systeme
- Metasprachen
- Forth inside

sehr schlecht bestückt sind die Info-Rubriken:

- Anfänger
- ANS
- Prozessorgeflüster
- Produktinfos
- Projekte, wird ab 1996 mit dem Leserforum verschmolzen

Ab 1996 kommt hinzu:

- Olaf Stroyke als Online-Verantwortliche. Unter anderem wird er die Meldungsrubrik 'Forth Online' aufbauen.

Kundenwunsch: Ein namhaftes Magazin im Forth-Bereich enthält kurze Meldungen, die die Leserinnen und Lesern aufgrund sehr schätzen. Diese vielen kleinen Texte müssen verwaltet, sortiert, redigiert und zum Druck vorbereitet werden. Die Autoren müssen umworben werden, News des Internets werden von vielen Personen aufbereitet und zusammengetragen. Die Autoren arbeiten mit verschiedenen Programmen und senden ihre Meldungen meist per E-Mail und (ganz aktuell) weit nach Redaktionsschluß an die Redaktion. In Zukunft sollen statt bisher 5 bis 10 Autoren etwa 50 bis 100 Personen in diese Arbeit einbezogen werden. Der Datenverkehr zwischen Zentralredaktionen, (neu zu schaffenden) Rubrikverantwortlichen und Autoren erfolgt hauptsächlich per E-Mail. Die Umstellung wird im ersten Quartal 1996 erfolgen und im Laufe des Jahres vollendet. Die Umstellung darf den Geschäftsbetrieb nicht belasten. Computerkenntnisse sind auf allen Ebenen vorhanden.

Projektkontext: Aufgrund personeller und finanzieller Erwägungen entschließt sich der Systemanalytiker zur Form eines öffentlichen Wettbewerbs. Ein wichtiger Gedanke hierbei war, den Kunden hierdurch von direkten Rückfragen der Softwareentwickler zu entlasten. Als oberste Zielvorgabe an die zu erstellende Software formuliert er Portabilität, Modularität, Wartbarkeit, Dokumentation. Die Anforderungen des Kunden werden sich nach Abschluß der ersten Betriebsphase wahrscheinlich stark ändern. Vielleicht aufgrund von Änderungen der Schnittstellen zum E-Mail- und Publishing-system bis hin zum Wechsel der verwendeten Hardware haben. Wahrscheinlicher sind Änderungen der Kundenwünsche aufgrund organisatorischer und inhaltlicher Vorgaben, die der Kunde bisher nicht formulieren konnte oder wollte. Nebenbei wird der Kunde derzeit als launisch und gereizt geschildert. Er übt sich sachgerechten Termindruck aus, sodaß seine direkten Vorgesetzten bereits von 'Panikmache' sprechen.

Die Aufgabe: Zu Erstellen sind verschiedene Konverter z.B. zum Umwandeln verschiedener verwendeter Zeichensätze, damit auch Sonderzeichen und Umlaute korrekt im Druck erscheinen. Dazu sind einzelne Meldungen zu separieren und ein Protokoll zur Verwaltungsvereinfachung zu generieren. Hervorhebungen (Überschriften, Aufzählungen, Tabellen, Hinweise, Quelltexte) müssen erkannt und mit Steuerzeichen für das DTP-Layout-Programm versehen werden. Schwierig wird die Spezifikation an den Stellen, wo die Software in die Arbeitsorganisation übergeht. Hier ist Phantasie und Kreativität gefragt. Hier muß der Benutzer Möglichkeit zur schnellen Anpassung haben. Die genauen Spezifikationen und die Beschreibung des Formats werden mit dem Weihnachtspäckchen verschickt.

Wer für die Feiertage etwas wirklich schönes und nützliches haben möchte, der muß sich jetzt beeilen:

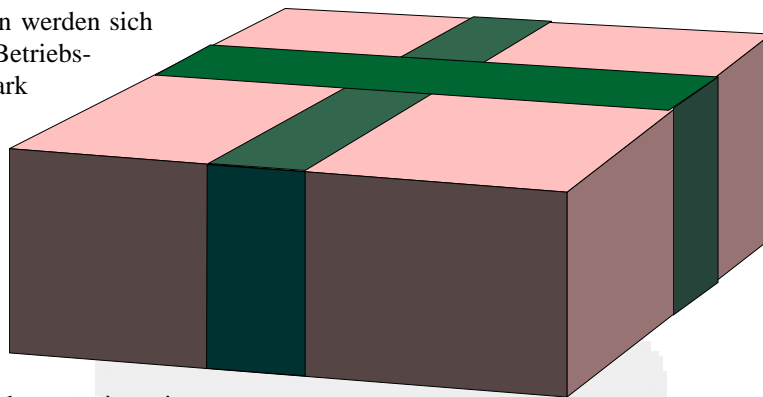
Das große Weihnachtspäckchen

mit Wettbewerb und zig Megabyte Forth und allem, was das Forthherz begehrt.

Aber es ist jetzt brandeilig, denn das gibt es

Nur noch bis 15. Dezember

(Datum des Eingangs, eine zweite Auflage ist nicht geplant. Zugegeben es ist sehr spontan, aber wer nicht gleich bestellt, guckt in die Röhre)



Bestellt wird so:

Verrechnungsscheck über
45.-DM an:

Redaktion 'Vierte Dimension'
c/o Claus Vogt
Ebersstr. 10
D-10827 Berlin

Das Päckchen enthält

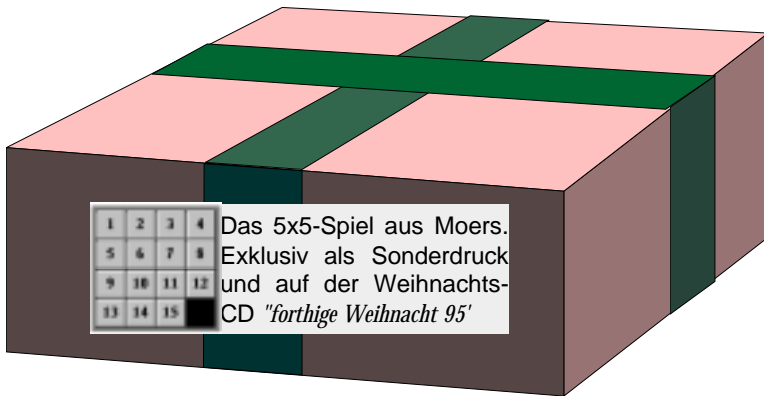
- eine Broschüre: Spiel 5x5 aus Moers (ein Spiel-Projekt für die Feiertage)
- eine Broschüre mit den offiziellen Wettkampfunterlagen und Beispielen zum VD-Weihnachtswettbewerb'95
- Die CD '*forthige Weihnacht 95*'

Die CD '*forthige Weihnacht 95*' enthält voraussichtlich: Die Unterlagen und Beispiele für den Weihnachtswettbewerb; Spiel 5x5 aus Moers; Alle vorhandenen Listings aller 11 Jahrgängen Forth-Magazin VD; Der komplette Jahrgang'95 der VD in mehreren Formaten mit allen Bildern; Die Usenet-Bretter de.comp.lang.forth, z-netz/sprachen/forth und comp.lang.forth mit zusammen bald 10 Mb; Der komplette Ftp-Server Taygeta aus USA mit bald 100 Mb; Der komplette Ftp-Server Asterix aus Portugal mit bald 100 Mb; Demoversion comForth der Firma FORTECH; Quellen, Forth-Systeme und Sammlungen von Claus Vogt, Ulrich Hoffmann, Bernd Paysan, Friedrich Prinz und anderen. Zahlreiche nützliche Programme zum Lesen von Texten, Mails und zum Betrachten von Bildern.

Ein ev. Überschuß wird für einen zukünftigen Wettbewerb verwandt. Ev. Verluste tragen Claus Vogt und Ulrich Hoffmann. Wir bedanken uns bei ihnen und allen Spendern, die mehr als 45.-DM auf ihren Verrechnungsscheck schreiben.



Wenn einem so viel schönes wird beschert...



wer dann beim Wettbewerb unter den ersten ist, der bekommt noch einen schönen Preis ...

Preis A: comFORTH für Windows

comFORTH für Windows ist ein FORTH-83 kompatibles 16-Bit-Forth-System, das für den Betrieb unter Windows und Windows für Workgroups ausgelegt wurde. Sein segmentiertes Speichermodell stellt einen vernünftigen Kompromiß zwischen verfügbarem Wörterbuchspeicher und Geschwindigkeit der Programmausführung dar. comFORTH für Windows besitzt eine grafische Bedienoberfläche, die der Systemphilosophie folgend auch durch den Benutzer erweitert werden kann. Aufgrund ihrer Integration sind alle Rufe des Windows-API interaktiv benutzbar, wobei nicht nur die Funktionsrufe, sondern auch alle übrigen Vereinbarungen wie Konstanten und Strukturbeschreibungen in comFORTH vorhanden sind. comFORTH für Windows ist seit Sommer 1995 in der Version 1.0 verfügbar.

Preis B: kkFORTH für PC, Z80, RTX

kkFORTH ist ein Forth83-System für die Programmentwicklung auf PC, Z80 und RTX. Es eignet sich speziell für die Entwicklung von Anwendungen im Mikrocontrollerbereich.



Preis C: bigFORTH 95 für PC

bigFORTH ist ein native-code-Forth für DOS-PCs mit einem 386er oder höher. Es läuft unter einem DOS-Extender im 32-bit-Protected-Mode. ANSI-Konformität, viele Libraries inclusive einem Text-'GUI', eine komplette Entwicklungsumgebung in ebendieser Text-'GUI', ein umfangreiches Handbuch und alle Sourcen sind natürlich dabei.

Preis D: F-PC - Das Buch zum Forth

Das F-PC ist ein sehr großes Public-Domain-Forth mit vielen zusätzlichen Utilities und allen Quellen. Beim Einstieg in dieses im PC-Bereich sehr verbreitete Forth-System hilft das Buch von Jörg Staben. Vom Forth-Zweizeiler bis zur Optimierung der win.ini kann hier jeder was lernen.

Forth. Ein Geschenk für gute Freunde. Zu Weihnachten und das ganze Jahr über. **Das**

Ein Forth-Programmierer ...

... ist höchst produktiv. Woran ganze Abteilungen von Fortran-66-Programmierern schon seit 1950 gescheitert sind, erledigt er wunschgemäß und völlig kostenlos (!) in einer einzigen Nacht. Ganz zum Leidwesen seiner Freundin. Den Großteil seiner Arbeitszeit verbringt er damit, sein Werkzeug 'Forth' zu einer noch besseren Entwicklungsumgebung zu machen. Seine Kollegen verstehen nie, woran er arbeitet.

Zwei Forth-Programmierer ...

... arbeiten gut zusammen. Dann vollbringen sie Wunder und werden von Kollegen und Vorgesetzten gleichermaßen geschätzt. Den Kollegen stehen sie gerne zur Seite. Erstaunlicherweise schlagen sie nach einem 'Blick über die Schulter' oft ganz kurze Lösungen vor, an die der C-Programmierer nicht gedacht hatte. Da nimmt es jeder gerne in Kauf auch zum zwanzigsten Mal zu erläutern, daß das Semikolon in C nicht (!) bedeutet, daß hier irgendwas anderes als die Zeile zueende ist. Vor wichtigen Abgabeterminen verbreiten sie entspannte Laune und mildern so Hektik und Mobbing.

Drei Forth-Programmierer ...

... kann man einstellen, wenn die 'Chemie zwischen den dreien' stimmt. Wenn die Chemie zu gut stimmt, sitzen sie immer in der Kaffeeküche und plaudern über Lösungen von Problemen, von deren Existenz die Menschheit bisher nichts wußte! Wenn die Chemie zu schlecht stimmt, fangen sie einen fürchterlichen Krach untereinander an. Meist geht es um die verschiedenen Lösungen von Problemen, von deren Existenz die Menschheit bisher nichts wußte - und das während der Arbeitszeit!

Sieben Forth-Programmierer ...

... gründen - sofern sie einen festen Wohnsitz im Bereich der Bundesrepublik Deutschland haben - einen Verein. Was sie in den mehr als hundert anderen Ländern des Planeten Erde und auf den unzähligen bewohnten Planeten dieser Galaxis so tun, wissen wir nicht. Wir hoffen, zu einem späteren Zeitpunkt ausführlicher berichten zu können.

Dreihundert Forth-Programmierer

... gibt es in der Forth-Gesellschaft.

Claus Vogt

Betreff: Forth Magazin sucht Programmierer

Die Aufgabe

Hallo, Forthler,

das Forth-Magazin 'Vierte Dimension' braucht mal wieder Eure Hilfe. Als kleine Belohnung gibt es dafür sogar ein Geschenk. Diesmal geht es um ein einfaches (aber nicht ganz triviales) Dateikonvertierungsproblem, das ich mal LST2PW nenne. (s.u.)

Ansonsten wird die Ausgabe 3/95 voraussichtlich ab Anfang Oktober ausgeliefert.

So, und nun das Problem LST2PW 'Konvertierung von Listings'. Bisher habe ich lange Listings mehr schlecht als recht und mit viel Handarbeit ins Heft übernommen. Dabei ist oft genug die Schönheit und Funktion eines mühsam geschriebenen Programms auf der Strecke geblieben.

Um diesen häßlichen Zustand zu beseitigen brauche ich einen Textkonvertierer auf PC. Ein Beispiel:

File: TEST.LST enthalte die folgenden 5 Zeilen:

.... es folgen 60 Zeilen Problembeschreibung ...

Wer kann mir sowas basteln? Als Belohnung gibt es ein Geschenk zur freien Auswahl aus der Liste der 'Autorengeschenke'.

Mit bestem Dank im Voraus

- Claus Vogt, Editor des Forth-Magazins Vierte Dimension - 8.9.1995

Die erste Lösung

Hallo clv,

Bitte schoen:

```
-----8<-----8<-----schnipp-----8<-----
UUencode version 1.51 (MSDOS)
begin 644 lst2pw.exe
```

... Es folgen 10 kB UUEncodetes Programm s. Listingdiskette LST2PW-P.EXE...

```
end
5344 bytes
encoded into 1 file(s)
----->8----->8-----schnapp----->8-----
```

Ich hoffe, es ist Dir so recht. Falls Du noch irgenetwas dazu brauchst (zB. Tabulatorerweiterung) melde dich ruhig. Übrigens, leider nicht in Forth (da bin ich noch nicht firm genug), sondern Borland Pascal :-(.

so long Gunter, 10.9.95

Danke schoen. Und dazu ging es noch dermassen schnell. Wenn ich das gewusst haette, haette ich mit dem Layout noch etwas gewartet, das haette mir viel Zeit gespart. "Ich hoffe, es ist Dir so recht. Falls Du noch irgenetwas dazu brauchst" Doch, es macht

genau das, was ich wollte. Kann ich da vielleicht auch die Quelle kriegen?

Nochmals vielen Dank,
- Claus - 13.9.95

Bitte schön.

... es folgt das nebenstehend abgedruckte Listing s.Listingdiskette LST2PW-P.PAS ...

Könntest Du mir „als Belohnung“ ein Exemplar der VD senden?

so long Gunter, 15.9.95
Zeitreisen haben Zukunft!

Was? Wie bitte? Turbopascal? Also wenn Du *das* nimmst, dann ... dann....

- Joerg, 15.9.95

Tut mir ja leid, aber in Forth hätte ich mehr als eine halbe Stunde gebraucht, evtl. das nächste mal... :-)

so long Gunter
Was man nicht begreift, kann man auch nicht verlernen.

Hm, muss mal gucken... Joerg, waere das nicht ein Fall fuer unseren (in der VD veroeffentlichen) Filtergenerator? Damit haben wir schon schwierigere Dinge geloesst..Claus, es gibt noch Hoffnung, but don't hold your breath.

- Joerg Plewe, 10.9.

Vielen Dank, aber es ist schon zu spät. Grade hat mir ein gewisser Gunter eine fertige funktionierende Lösung in Turbopascal geschickt.

Vielen Dank an alle, die sich Gedanken gemacht haben.

- Claus -, 14.9.95

Es juckte mir denn ja nun doch in den Fingern und so ist folgendes ANS-Forth Programm entstanden. Verbesserungsvorschlaege sind herzlich willkommen.

```
-lst2pw.ans-----
\ LST2PW Filter in ANS Forth Ulrich Hoffmann 12. Sep
95 \
\ Dieses Programm ist Public Domain Software.
\ Dieser Filter konvertiert Listings in ein Desk-
Top-Publishing-Format.
\ Dies ist ein ANS Forth Standard-Programm.
\ Es benutzt
\ - die CORE-Wortgruppe
\ - die CORE-Erweiterungs-Wortgruppe:\
\ - die DOUBLE-NUMBER-Wortgruppe: D=
\ - die FILE-ACCESS-Wortgruppe:
WRITE-LINE WRITE-FILE
\ - die EXCEPTION-Wortgruppe: THROW
\
... es folgen 150 Zeilen ANS-Forth und
Kommentare,
s.Listingdiskette LST2PWF.ANS...
```

Viele Gruesse,
Ulrich Hoffmann, 12.9.95

Oh, vielen Dank. Demnächst werde ich das Programm testen, aber wie gesagt: Eine Turbopascal-Lösung war schneller. Auf wel-

File: Ist2pw-P.pas vom 10.9.1995

```
1 program vd;
2 uses dos;
3
4 var inp,out: text;
5 Ch: char;
6 outCh: string[15];
7 ftime: Longint; { For Get/SetFTime}
8 dt : DateTime; { For Pack/UnpackTime}
9 nr: word;
10 N,maxN: longint;
11
12
13 function LeadingZero(w : Word) : String;
14 var
15 s : String;
16 begin
17 Str(w,0,s);
18 if Length(s) = 1 then
19 s := '0' + s;
20 LeadingZero := s;
21 end;
22
23 function DisplayNr(w: word): String;
24 var
25 nrStr: string[10];
26 nrStr1: string[10];
27 i: word;
28 begin
29 Str(w,3, nrStr);
30 nrStr1:= '';
31 for i:=1 to 3 do
32 case nrStr[i] of
33 ' ': nrStr1:=nrStr1+'<>';
34 '0'..'9': nrStr1:=nrStr1+nrStr[i];
35 end;
36 nrStr1:=nrStr1+' ';
37 DisplayNr:=nrStr1;
38 end;
39
40 begin
41 nr:=1;
42 n:=0;
43 writeln(Listingkonvertierprogramm für VD 1.0 (c) 1995 gunte
r@top.east.de);
44 if ParamCount < 2 then
45 begin
46 Write('Aufruf: ');
47 Writeln(ParamStr(0),' INFILE.LST OUTFILE.PW');
48 halt(1)
49 end;
50 assign(inp,Paramstr(1));
51 assign(out,Paramstr(2));
52 reset(inp); rewrite(out);
53 GetFTime(inp,ftime); { Get creation time }
54 UnpackTime(ftime,dt);
55
56 writeln(out, '<$ListTit>File: 'Paramstr(1),' vom ',dt.day,',',dt.m
onth,',',dt.year);
57 write(out, '<$ListTst>');
58 Write(out,DisplayNr(nr));
59 inc(nr);
60 while not Eof(inp) do
61 begin
62 Read(inp, Ch);
63 case Ch of
64 #13: begin
65 outCh:=<.>+#13+#10; read(inp, Ch);
66 if not eof(inp) then begin
67 outCh:=outCh+DisplayNr(nr); inc(nr)
68 end;
69 if N > maxN then maxN:=N;
70 N:=0;
71 end;
72 ' ': begin outCh:= ' '; inc(N); end
73 else
74 begin
75 outCh:=Ch; inc(N)
76 end
77 end;
78 Write(out, outCh); { Dump text file }
79 end;
80 Write(out, '<$ListUTit>Längste Zeile hat 'maxN,' Zeichen');
81 close(inp); close(out)
82 end.
83
```

Längste Zeile hat 86 Zeichen

Das Programm wurde zum Abdruck von seinem Nachfolger Ist2pw 1.1 konvertiert, die <J> restauriert und die <_> wieder in < zurückverwandelt.

chem System kann ich denn Dein Programm testen?

- Claus -14.9.

Ich denke dass Du da jedes ANS-Forth-System benutzen kannst. Oder war Deine Frage, welches spezielle System ich Dir fuer den PC empfehlen wuerde?

Ulli, 18.9.95

Ja, so ähnlich. Um ehrlich zu sein: Bisher habe ich weder ein ANS-Forth auf einem meiner Rechner noch den aktuellen Standard im Schrank. Meine Standard-Ausgabe ist von 92 oder so. Ich benutze meistens F-PC.

Also, welches ANS-Forth wäre für mich geeignet? Wo kriege ich es her?

- Claus - 23.9.95

Fuer das obige Problem gibt's ein Forth-macs tool, ich habe eben mal nachgesehen, der Aufwand betraegt eine Anpassung weniger code zeilen. Und sonst riecht das doch nach Perl, oder?

Hanno Schwalm, 12.9.

begin 644 LST2PW.EXE
.... Es folgen etwa 100 kB UUEncodetes Programm
s. Listingdiskette LST2PW-C.EXE....
end

Hier mein Vorschlag in C (Lex hilft):

... es folgen etwa 50 Zeilen Listing,
s. Listingdiskette LST2PW-C.C ...

Viele Gruesse,
Matthias Zierbock

Schoenen Dank. Du warst der dritte Einsender. Nie haette ich gedacht, daß so schnelle so tolle Reaktionen kommen.

- Claus -

Eigentlich ist das so eine Aufgabe für SED, AWK und ein Shell-Script. Aber wir sind ja FORTHer, wir verwenden FORTH dafür!

Verbesserungsvorschlag: Zeilennummern "tiefgestellt", also kleiner, und noch etwas Abstand zwischen Zeilennummer und Text (dazu müßte man wissen, wie die dazugehörigen Kommandos des DTP-Systems heißen). Und als FORTHer natürlich: Zeilennummern in HEX ;-)! Dann hat auch der AWK als Took verschissen ;-).

Und bei den <>-Markierungen: Müssen da nicht wieder die < und > gesichert werden? Sonst gibt's wieder unlesbare Programme, bei denen >R und R> nur durch Raten ermittelt werden können.

Was bin ich froh, daß ich LaTeX verwende. Da gibt's für solche Fälle verschiedene Listing-Styles, die manchmal sogar Schlüsselwörter besonders schreiben (ist bei FORTH nicht so wichtig).

Bernd Paysan, 14.9.95

"Late answers are wrong answers!"

Hallo, Bernd,

grade habe ich Dir auf Deine Mail hin mal 'Pressworks erweiterte Formatierungssprache' zugeschickt. So völlig bin ich mit den Experimenten rund um die Listings auch

noch nicht zufrieden. Also, wenn Dir was gutes einfällt. Schick mir doch mal den passenden Filter rüber.

Den schönsten Ist2pw-Filter von Gunter Schenck habe ich mit dem ersten Preis prämiert. Er hängt auch unten dran.

File: Ist2pw.c

```

1  %{
2  #include "string.h"
3  #include <stdio.h>
4  #include <sys/stat.h>
5  #include <time.h>
6  int Cmax = 0;
7  int row = 0;
8  int column = 0;
9  extern FILE *yyin;
10 %}
11 %%
12
13
14 [t]+ ;
15 [^\n] { dru(row); column++; ECHO; }
16 [ ] { dru(row); printf(" "); column++; }
17 ^\n { if(row==0)row++; dru(row);
18 printf("<J>\n"); row++; }
19 \n { printf("<J>\n"); row++; column=0; }
20
21 %%
22
23 void main(int argc, const char *argv[]){
24     struct stat buf;
25     struct tm fil;
26
27     if(argc!=2){printf("LST2PW input \nLST2PW input >output");
28         exit(1);}
29
30     if( stat(argv[1],&buf) != -1 ){
31         _localtime(&buf.st_mtime,&fil);
32         printf( "<ListTit>File: %s vom %d:%d:%d\n", argv[1],
33             fil.tm_mday, fil.tm_mon, fil.tm_year);
34     } else {
35         printf( "kann Datei %s nicht oeffnen !", argv[1] );
36         exit(1);
37     }
38     yyin=fopen( argv[1], "r" );
39     yylex();
40     printf( "<ListUTit>Längste Zeile hat %d Zeichen", Cmax );
41
42
43 }
44
45 void dru(int row){
46     if( row==1 & column==0 )
47         printf("<ListTxt>");
48     if( column > Cmax )
49         Cmax=column+1;
50     if( column > 0 )
51         return;
52
53     if( row < 10 )
54         printf("<_>%d",row);
55     else
56     if( row < 100 )
57         printf("<_>%d",row);
58     else
59     if( row < 1000 )
60         printf("%d",row);
61 }
62

```

Längste Zeile hat 66 Zeichen

So long, - Claus -17.9.

PS: Dein 'Optimierungsartikel' hat schon vor Erscheinen mehrere sehr positive Echos gefunden. Das gibt's sonst kaum mal.

"als FORTHer natürlich:
Zeilennummern in HEX ;-)"

Wenn schon, in base-36. Aber den Emacs etc. erweiterst Du, damit er mit solchen Zeilennummern umgehen kann.

"Was bin ich froh, daß ich LaTeX verwende.
Da gibt's für solche Fälle verschiedene Listing-Styles, die manchmal sogar Schlüsselwörter besonders schreiben (ist bei FORTH nicht so wichtig)."

Kannst Du welche empfehlen?

Abgesehen vom verbatim-environment find' ich Listings recht muhsam.

M. Anton Ertl, 18.9.95

Some things have to be seen to be believed-
Most things have to be believed to be seen

Ich habe mal mit einem Scheme-Style gearbeitet, den fand ich ganz brauchbar (da waren die Listings auch Verbatim). Allerdings mußte ich noch einen Tab-Expander (in Forth ;-)) schreiben.

Für die Zwecke der VD würde sich verbasef.sty empfehlen, damit kann man Dateien (auch ausschnittsweise) in Figures importieren (immer Verbatim) und Zeilennummern in anderen Fonts schreiben wie den Text, und sogar (!) Zeilennummern als Labels deklarieren lassen und so passende Seiten angeben.

Latürlich braucht man nicht alle Zeilen zu nummerieren, sondern etwa Zeile 4, 8, 12, 16 (oder normal 5, 10, 15, etc.). Hex geht's leider nicht.

Bernd Paysan, 19.9.

"Late answers are wrong answers!"

"Natürlich braucht man nicht alle Zeilen zu nummerieren, sondern etwa Zeile 4, 8, 12, 16 (oder normal 5, 10, 15, etc.). Hex geht's leider nicht."

Doch, man muß alle Zeilen nummerieren! Weil: Wenn ein paar Zeilen zu lang sind, soll der Leser sofort sehen, daß eine Druckzeile die Fortsetzung der letzten Listingzeile ist. Die Fortsetzungszeile hat dann

F-PC - Das Buch zum Forth
Für Ein-, Um- und Aufsteiger

Das F-PC ist sicher das derzeit m...
Forth für PC. Es stammt wie...
Hand von Tom Zimmer...
Gemeinde neue Dinge...
Staben ist...
Einführung...
monatlich in der c't
bework...
dem Verkauf zu retten.
we...
hier jeder was lernen.

bereits vergeben

- gestiftet vom Autor Jörg Staben

nämlich keine Nummer. So macht das auch die c't. In VD 3/95 habe ich teilweise eine andere Variante probiert. Ich habe ein spezielles Zeilenendezeichen '<' eingeführt. Das gefiel mir nicht so gut.

Vorschläge und Kritik sind jederzeit willkommen.

- Claus, Editor der VD - 23.9.

Der Hauptgewinn

Der Hauptgewinner des 'Listing-filterwettbewerbs' der VD steht fest. Es ist.

Gunter Schenck mit seinem LST2PW Turbopascalprogramm

Sein Programm war als erstes hier, sofort lauffähig, ist total kurz, funktioniert und was noch besser ist: Es sieht hübsch aus. Gunter hat ein Probeheft der VD gewonnen und außerdem ein Geschenk aus der Liste der Autorengeschenke. (z.B. in VD3/95 auf S.4 zu finden.)

Bitte sendet keine Lösungen mehr ein!!!!

Eigentlich wollte ich gar keinen Programmierwettbewerb veranstalten, sondern nur ein nutzbares Programm haben. Wenn irgendwer gerne die diversen Mails & News zum LST2PW in einen Artikel verwandeln möchte: Bitte melden! Mir fehlt leider die Zeit.

Die zahlreichen Reaktionen zum Thema haben mich sehr erfreut. Vor allem hatte ich tatsächlich wenige Tage nach der Bitte in

de/comp/lang/forth zwei lauffähige Programme vorliegen. Vielen Dank nochmal.

Aus der Fülle der vielen Diskussionsbeiträge möchte ich hier nur mal einen wahllos herausgreifen und stellvertretend für alle anderen beantworten:

joerg.plewe@jpsforth.forth-ev.de wrote in de/comp/lang/forthon 15.09.95 :

"Claus, es gibt noch Hoffnung, but don't hold your breath."

Warum sollte ich mir denn die Brust halten? Oder heißt 'Breath' noch was, was ich ohne Wörterbuch nicht verstehe?

"Was? Wie bitte? Turbopascal? Also wenn Du *das* nimmst, dann ... dann...."

Ja, Jörg. Das finde ich wirklich gut, daß Du Deine Meinung hier so offen äußerst. Ich bin stolz auf Dich, weiter so!

Tschüß,

- Claus - 17.9.

Hallo Gunter,

Ein Probeexemplar ist an Deine obige Adresse bereits angewiesen. Es duerfte bis spaetestens Mitte Oktober da sein. Sonst ist was schief gegangen, dann melde Dich noch mal.

Ausserdem erenne ich Dich hiermit zum Gewinner des Programmierwettbewerbs. Eigentlich sollte das ja gar kein Programmierwettbewerb werden. Aber bereits nach zwei Tagen waren zwei lauffaehige Programme und ausserdem zwei Quellen in verschiedenen Sprachen eingetroffen. Ich traue mich kaum, in de/comp/lang/forth zu schauen, was inzwischen noch alles gekommen ist. Die Resonanz war jedenfalls hervorragend.

Deine Loesung war die erste, die eingegangen ist, das lauffaehige Programm das kuertzeste und damit geht der Hauptgewinn an Dich.

Der Hauptgewinn ist ein Geschenk aus der Liste der 'Autorengeschenke', die Du z.B. in vd3/95 gegenueber vom Inhaltsverzeichnis finden wirst. Es sind keine grossartigen Sachen, aber doch ganz huebsche Saechelchen dabei. Mail mir gelegentlich, was Du Dir davon ausgesucht hast. Dann bring' ich's auf die Reise.

Nochmals vielen Dank fuer Deine prompte Reaktion. Das Programm spart mir sehr viel Zeit beim Layouten.

Tschuess,
- Claus -

Gunter wünschte sich als Hauptgewinn das Buch von Jörg Staben.



Das Werkzeug Win32Forth

von Jörg Staben

Hagelkreuzstr.23; D-40721 Hilden

Win32Forth - kurz Win32For genannt - ist ein 32Bit-Forth System für Windows NT und Windows'95. Es wurde, ebenso wie sein erfolgreicher Vorgänger F-PC, federführend von Tom Zimmer und Andrew McKewan in Zusammenarbeit mit Jim Schneider (ASM486), entwickelt. Win32For ist, wie F-PC auch, public domain und wird mit sämtlichen Quellen zur Verfügung gestellt. Die zum (Neu-)Generieren notwendigen Werkzeuge sind allerdings diesmal nicht mit dabei, da dazu u.a. ein C-Compiler von Borland gehört. Damit wird sich die Schwemme der Win32For-Clones in einem überschaubaren Rahmen halten.

Stichworte: Forth-Systeme, Tom-Zimmer, win32Forth, 32bit, Spiel

Win32For zeichnet sich durch 7 Punkte aus, auf die Tom Zimmer besonders großen Wert legt:

- I Ein einziger linearer 32bit Adreßraum für Programm und Daten (flat memory model).
- II Unterstützung von anwenderdefinierten Fenstern.
- III Unterstützung von Pulldown Menüs und Dialog Boxen.
- IV Zusammengefaßte Entwicklungsumgebung (IDE) mit Editor, Debugger, Compiler, Decompiler, Assembler, Disassembler, Hypertext.
- V Blitzschnelles Compilieren (wichtig bei sehr, sehr großen Projekten).
- VI Vollständige Schnittstelle für alle Windows-Prozeduraufrufe und -Konstanten.
- VII Gesamter System Quelltext verfügbar.

Die Bedeutung dieser Punkte liegt auf der Hand, wenn man über 64K-Grenzen gestolpert ist (alle 16Bit-Forth) oder wenn man in einer komfortablen Entwicklungsumgebung arbeitet, die eigene Anwendung aber nicht über diesen Komfort verfügen kann. Genauso ärgerlich ist es, wenn Aufgaben nicht lösbar sind, weil das betreffende Forth-System nicht über eine hinreichend ausgelegte Windows-Schnittstelle

verfügt (LMI) oder die Fehlersuche zu lange dauert, wenn Decompiler, Debugger und Disassembler fehlen (MPE).

Windows und OS/2

Möchten Sie Win32For auf der langsam aussterbenden Kombination DOS/Windows 3.1 betreiben, ist dazu ein kleines Microsoft-Päckchen namens Win32s notwendig, das dem 16-bittigen Windows 3.1 zu einer 32Bit-Programmierschnittstelle verhilft. Dies ist kein großes Hindernis, da Win32s kostenlos auf nahezu jeder CD-ROM der maßgeblichen Computerzeitschriften wie PC-GAMES enthalten ist.

Übrigens - für die Schachspieler unter Ihnen lohnt sich nach der Installation von Win32s ein Blick auf Virtual Chess, ein sehr schön gemachtes Schachspiel mit der parallelen Darstellung der schachprogramm-üblichen 3D-Ansicht und der Ansicht auf xStellungsx

Die OS/2-Anwender unter Ihnen können sich leider nicht auf ihr mitgelieftertes VWIN32S verlassen, auch sie müssen Win32s installieren. Aber was tut man nicht alles für ein gutes Forth...

Win32For präsentiert sich funktionell und leistungsstark, wie man es vom F-PC gewohnt ist. Es wertet beim Start eine Datei Win32For.CFG aus, stellt sofort 5830 Wörter zur Windows- und Forth-Programmierung bereit, kennt VIEW, SEE, DEBUG und verfügt über die oftmals heißbegehrten objektorientierten Erweiterungen.

Ein Forth also, mit dem sich gut leben läßt.

Die Aufgabe

Eines der Spiele, die wohl eher mit dem WATCOM C++ Compiler geschrieben wurden (Sie sollten sich auch mal die Software Developer CD von Intel Architecture Labs library ansehen; 50DM, ein C-Compiler, eine halbe Stunde und Ihr eigener DxxM-Clone läuft) als mit einem Forth-System, ist die Strategiesimulation COMMAND&CONQUER.

Dieses Spiel, kurz C&C genannt, steht in der Tradition der Spiele DUNE und DUNE2. Der Inhalt geht auf den - auch verfilmten - Zyklus "Der Wüstenplanet" von H.W.Franke zurück, einem oftmals ausgezeichneten ScienceFiction-Autor. Der Wüstenplanet Arrakis ist öde und unwirtlich, seinen einzigen Bodenschatz stellt SPICE, das Gewürz dar. Dieses Gewächs befähigt die Gilde der Raumschiffpiloten dazu, mittels mentaler Kräfte ihre Schiffe in den Hyperraum zu portieren. Sein unschätzbare Wert macht das Gewürz zum Streitobjekt zwischen verfeindeten Familien, die sich gegenseitig den Gewürzabbau streitig machen. Und schon tobt eine erbitterte Fehde, die in C&C als Roboterschlacht ausgetragen wird.

Soweit zum literarischen Hintergrund, Einzelheiten möge der geneigte Leser der auch als Taschenbuch erhältliche SF-Reihe entnehmen.

Einige Missionen innerhalb des Spieles prüfen den Spieler hart, indem man gezwungen wird, mit wenig Androiden, wenig Material und - vor allem - wenig Geld seinen Auftrag zu erfüllen.

Da kommt die Veröffentlichung eines Patches gerade recht: In der Datei eines abgespeicherten Spielstandes läßt sich anhand der Zeichenkette 'Computer' ein Byte benennen, das den verfügbaren Geldbetrag in einer Mission beschreibt. Dieses Byte gilt es nun, in der Datei "SAVEGAME.0xx" von 0 auf 1 umzusetzen.

Die Lösung

Als erprobter F-PC-Anwender denkt man sofort an eines des vielen - wirklich einsetzbaren - Beispielprogramme für F-PC/TCOM, den Datei-Editor BINEDIT. Dieser Editor erlaubt in

Dateien die Suche nach Bytes, Zeichen und -ketten in HEX und ASCII, das Ändern von Bytes und natürlich das Zurückschreiben auf Disk. Aber leider, leider ist BINEDIT an die magische 64K-Grenze gefesselt, eine 150KB große "SAVEGAME.023" läßt sich nicht editieren!

0 value file

: open-savegame

s" savegame.023" R/W open-file . to file ;

: os open-savegame ;

: close-savegame

file close-file drop ;

: cs close-savegame ;

: read-savegame (a l --)

file read-file 2drop ;

: rs read-savegame ;

decimal

: read-buffer (--)

pad 1024 rs ;

: rb read-buffer ;

: check-buffer

rb pad 1024 dump ;

: cb check-buffer ;

Da Forth mit dem verabschiedeten Standard ANS auch eine übertragbare Dateischnittstelle bekommen hat, fällt positiv auf, daß Win32For hier ANS-konform ist.

Schnell sind Wörter für den etwas komfortableren Zugriff auf C&C-Spielstandsdateien geschrieben, kurz wird kontrolliert, ob PAD existiert und schon wird die Datei von Hand eingelesen und von Hand PAD gedumpte, bis 'Computer' irgendwo im hinteren Drittel der Datei zu erkennen ist. Nun wird die

aktuelle Stellung des Dateizeigers mit FILE-POSITION ausgelesen, die Bytes sorgsam abgezählt und mit REPOSITION-FILE der Dateizeiger rückwärts richtig positioniert. Von dieser Stelle wird - zur Kontrolle - noch einmal in PAD eingelesen, die Bytes gecheckt, der Dateizeiger erneut positioniert, das magische Byte von 0 nach 1 geändert und PAD mit WRITE-FILE zurückgeschrieben.

Spannung! Forth wird verlassen, Windows wird verlassen und C&C gestartet. Die betreffende Mission wird gestartet und schier unbegrenzte Geldvorräte sorgen für eine sturmreif geschossene gegnerische Festung.

Fazit

Win32For hat in einer kleinen Aufgabe gezeigt, wie funktionell es ist, da auch hier wieder zu jedem Wort der Quelltext angezeigt werden kann. Ebenso hat sich gezeigt, daß die Programmierung mit Win32For komfortabel ist - z.B. hat nicht jedes Forth eine Eingabewiederholung.

Zudem ist Programmieren unter Windows etwas bediensicherer, ein hier und da interaktiv vergessener Parameter zerschießt nicht eine ganze Datei, sondern die Win32s nutzt auch bei seinem Abbruch die Chance, alle Dateien unaufgefordert und ordentlich zu schließen.

Let's go forth - geben auch Sie der Windows/Forth-Programmierung mal eine Chance, es muß ja nicht direkt Windows-Programmierung sein. □

Forth-Systeme:

WIN32FOR läuft auch in OS/2 - WINOS/2 Sitzungen !

Ich bin, fälschlicher Weise, immer davon ausgegangen, daß WINOS/2 die von MS gelieferte WIN32S Erweiterung nicht benötigt, weil WINOS/2 ein virtuelles VW32S ab der OS/2 Version 3.0 WARP grundsätzlich in das System mit einbindet. Dank der 'Überredungskünste' von Bill McCarthy, der mir via DFÜ und COMP/LANG/FORTH mehrfach geraten hat,

win32forth on taygeta

Aref Nammari wrote:

"I recently downloaded the file win-32for.zip from the taygeta site. I unzipped it and was ready to run it. When I did nothing, it did not run. I looked in the files and there was no install.com among the files . Did anybody run into the same problem ? Can somebody please help me try to make the thing work ?"

Win32Forth requires either Windows95 or WindowsNT to function. You run it by just executing Win32For.EXE. The file Win32For.IMG must be in the same directory as the .EXE file. You may have gotten a corrupted archive so something.

I am about to release a new version (within a couple of weeks) that includes an installation program and a Windows based editor to replace the NEWZ editor included with the system.

You might want to wait for this new release "Version 1.6" which will have DLL support and (I Hope) OLE support (Andrew are you listening?).

ZForth@ix.netcom.com(Tom Zimmer) in /comp/lang/forth

Wer ernsthaft in die (traditionelle 16 Bit) Windows-Programmierung einsteigen will, bei dem darf der "Petzold" im Bücherregal nicht fehlen, da eine sinnvolle Nutzung des Windows API wahrscheinlich erst nach dieser Lektüre möglich ist: "Programming Windows 3.1", Charles Petzold, Microsoft Press, Redmond, 1992, ISBN 1-55615-395-3

WIN32S zusätzlich zu installieren, habe ich dem OS/2 WIN32S in der Version 1.15 'spendiert'. Und die Wochen des Ärgerns (weil ich das neue Spielzeug von Tom nicht nutzen konnte) sind vorbei! Jetzt geht es erst einmal auf Entdeckungstour...

Vielleicht ist das ein weiterer Anstoß für die 'DFÜ-Muffel' - Kontakte, Hilfe und Anregungen zu FORTH gibt's längst weltweit - und öfter als über die CD ;-)

F.PRINZ@MHB.GUN.DE

Parallel-Forth: Der neue Weg

von Michael B. Montvelishsky

mic@fclub.mordovia.su; ave. Lenin 21-29; Saransk; Russia 43000

Eine neue Programmiersprache erscheint im allgemeinen immer dann, wenn sich eine neue Programmiermethode oder ein neues Paradigma aufgetan hat. ALGOL-60 ging mit dem Aufkommen der strukturierten Programmierung einher. PASCAL läutete die komplexeren benutzerdefinierten Datentypen ein. Objektorientiertes Programmieren kam in enger Verbindung mit Smalltalk auf. C++ ebnete ihm den Weg in die professionellen Programmierstuben. Neue Programmierparadigmen rufen neue Programmiersprachen auf den Plan. Die einzigen, die auf keine neue Programmiersprache umzusteigen brauchen, sind die Forth-Benutzer. Forth ist erweiterbar und kann sich leicht jedem neuen Paradigma anpassen. Im vorliegenden Artikel wird beschrieben, wie Forth auch für die Methodik der Parallelprogrammierung fitgemacht werden kann.

Stichworte: Parallelverarbeitung Transputer F-PC

Linda und Occam

Eines der aktuellsten Gebiete der Wissenschaft des Programmierens ist das des Parallelrechnens. C.A.R. Hoare, der Verfasser von "Communication Sequential Processes", ist der Begründer der Theorie des Parallelrechnens. Die

eigentliche Programmiersprache, sondern eine Erweiterung zum parallelen Ausbau bestehender Programmiersprachen. Es gibt C-Linda, Fortran-Linda und Forth-Linda. Lindas Paradigma stützt sich auf die Idee der aktiven und passiven Tupel. Aktive Tupel sind

Forth-Benutzer brauchen auf keine andere Sprache umzusteigen. Forth ist erweiterbar und kann sich leicht neue Programmier-Paradigmen

Programmiersprache OCCAM stützt sich auf die Theorie von Hoare. Sie kann als eine der wichtigsten Programmiersprachen für das Parallelrechnen betrachtet werden. Zwei Dinge sind an dieser Sprache wesentlich:

- 1. Kanäle zum Austausch von Informationen und zur Synchronisation der beteiligten Prozesse.
- 2. Die Fähigkeit des gerade laufenden Prozesses, eine große Zahl von Tochterprozessen parallel abarbeiten zu lassen. Der Mutterprozeß hält inne und wartet solange, bis alle Tochterprozesse beendet sind. Dann erst nimmt er seine Arbeit wieder auf.

Ein anderes weitverbreitetes Modell der Parallelprogrammierung ist Linda von D. Gelernter. Linda ist keine

Prozesse, die von einem anderen Prozeß gestartet werden und mit diesem zusammenarbeiten. Passive Tupel sind Mittel zum Austausch von Informationen, "Austausch" so ähnlich wie über kleine Zettel an einer Anschlagtafel.

Die Paradigmen von OCCAM und Linda arbeiten nicht gegeneinander. Sie ergänzen sich. Es ist schwierig, Linda in OCCAM nachzubilden. Es ist aber auch genauso schwierig, OCCAM in Linda wirksam darzustellen.

Forth

In diesem Artikel wird über eine Parallel-Erweiterung von Forth gesprochen. Wenn man sich die Wortliste weiter unten ansieht, so erkennt man, daß sich mit diesem Parallel-Forth eigentlich nichts Neues ergeben hat.

Zunächst ein paar Bemerkungen über den Quelltext:

- * 1. Alle USER-Variablen in einem Tochterprozeß haben denselben Wert wie im Mutterprozeß vor einem $||(\dots)$ - oder $EV(\dots)$ -Konstrukt. Man kann also USER-Variablen zur Parameterübergabe von Mutterprozeß zu Tochterprozeß oder zu einem dem Mutterprozeß beigeordneten Prozeß verwenden.
- * 2. Man stelle sicher, daß die USER-AREA und die Stackgröße groß genug gewählt wurden, so daß der Prozeß nicht hängen bleibt. Es lassen sich leicht Testworte schreiben, die die betreffenden Speicherplätze mit Zeichen belegen, so daß man nachsehen kann, wieviele davon aufgebraucht wurden.
- * 3. Wenn man einen Tochterprozeß mit $||(\dots)$ einrichtet, so kann man nicht gleichzeitig mit $EV(\dots)$ zusammenarbeiten, bevor nicht PAR den Ring mit $EV(\dots)$ startet.
- * 4. Für $||(\dots)$ - und $EV(\dots)$ -Prozesse ist eine automatische Speicherzuteilung vorgesehen. Automatische Speicherfreigabe ist aber nur für $||(\dots)$ -Prozesse (und alle darin enthaltenen Prozesse) vorgesehen. Können Sie sich vorstellen, warum?
- * 5. Alle Prozesse beginnen mit leeren Stacks.

□

Der Abdruck dieses Artikels erfolgt mit freundlicher Genehmigung der amerikanischen Forth Interest Group.

Der ursprüngliche Artikel erschien in der "Forth Dimensions" vom März/April 1994.

Die Übersetzung für das Forth-Magazin 'Vierte Dimension' besorgte Fred Behringer, München.

Den Quelltext dieser Parallel-Forth-Erweiterung findet man im folgenden Listing. Ich habe F-PC in der Version 3.53 verwendet. Das ist eines der vollständigsten Forthsysteme für IBM-PCs und Kompatible, und es ist Public Domain! Der Quelltext enthält keine Code-Definitionen, sodaß man das Ganze mit nur ganz kleinen Änderungen leicht auch auf andere Forth-Plattformen übertragen kann. Zur Erhöhung der Wirksamkeit kann man PAUSE in Machinensprache neu fassen. Am besten implementiert man Parallel-Forth aber auf irgendeiner Mehrprozessor-Anlage.

File: parallel.seq

```
1 \ Dr. Michael B. Montvelishsky, Saransk/Rußland, 1993
2
3 \ Zusatzworte für den Parallelbetrieb von Michael Montvelishsky, 25.9.93
4 ANEW PARALLEL
5
6 CR .FREE
7
8 \ =====
9 \ It is just a simple troock :-) Wortspiel: "TROOCK" ist die
10 \ To replace the pause hoog :-) russische Version von "TRICK"
11 \ PAUSE liegt zunächst als Code-Definition vor
12 ' PAUSE \ OLD
13 DEFER PAUSE ' NOOP IS PAUSE
14 ' PAUSE \ OLD NEW
15 2DUP SWAP - \ OLD NEW SHIFT
16 OVER 1+ +! \ OLD NEW ( ADJUSTED!)
17 TUCK \ NEW OLD NEW
18 HERE SWAP - \ NEW OLD SIZE
19 CMOVE
20 FORGET PAUSE
21 \ PAUSE ist jetzt DEFERiertes Wort!
22 \ =====
23
24 USER
25 VARIABLE (NEST) \ Zeiger zum zuletzt eingerichteten Prozeß
26 VARIABLE (PARENT) \ Zeiger zum Mutterprozeß
27 VARIABLE (NEXT) \ Zeiger zum nächsten Prozeß im Ring
28 VARIABLE RP \ Zur Aufbewahrung
29 VARIABLE SP \ der Stackwerte
30 FORTH
31
32 VARIABLE RETURN-STACK-SIZE \ Zur Veränderung der Return-Stack-Länge
33 VARIABLE DATA-STACK-SIZE \ Dito für Datenstacklänge
34 VARIABLE USER-AREA-SIZE \ Dito für Länge der USER-AREA
35
36 \ Werkzeug zur globalen Speicherzuteilung:
37 DP CONSTANT (HEAP)
38 : MALLOC (HEAP) +! ;
39 : HEAP (HEAP) @ ;
40
41 \ Variablen vorbelegen:
42 UP @ (PARENT) !
43 UP @ (NEXT) !
44 0 (NEST) !
45 64 USER-AREA-SIZE !
46 128 RETURN-STACK-SIZE !
47 64 DATA-STACK-SIZE !
48
49 : PROC-SIZE ( - current-proc-size )
50 USER-AREA-SIZE @ RETURN-STACK-SIZE @ + DATA-STACK-SIZE @ +
51 ;
52 : GO-NEST SP@ RP@ RP ! SP ! (NEST) @ UP ! RP @ RP! SP @ SP! ;
53 : (PAUSE) SP@ RP@ RP ! SP ! (NEXT) @ UP ! RP @ RP! SP @ SP! ;
54
55 \ Alle laufenden Prozesse beenden und zum Mutterprozeß gehen
56 : STOP-ALL (PARENT) @ (NEXT) ! (PAUSE) ;
57
58 \ Parallelbetrieb einschalten/ausschalten
59 : MULTI [ ] (PAUSE) IS PAUSE ;
60 : SINGLE [ ] NOOP IS PAUSE ;
61
62 \ Sich selbst beenden
63 : STOP ( --- : Nimmt Prozeß aus dem Prozeßring )
64 UP @ DUP (NEXT) @ - IF \ Letzter im Ring ?
65 UP @ BEGIN (NEXT) 2DUP @ - WHILE @ UP ! REPEAT
66 SWAP UP ! (NEXT) @ SWAP ! (PAUSE)
67 ELSE
68 STOP-ALL
69 THEN
70 ;
71
72 \ Erzeugung eines neuen Prozesses mit vorgegebener
73 \ pfa, Mutterprozeß und vorhergehendem Prozeß
74 : MOUNT ( pfa parent prev --- : Neuen Prozeß erzeugen und einbinden )
75 HEAP PROC-SIZE MALLOC \ Platz schaffen für Prozeß
76 UP @ DUP >R OVER USER-AREA-SIZE @ CMOVE \ Kopier Mutterprozeß-
USER-Variablen
77 OVER UP ! (NEXT) @ SWAP UP ! (NEXT) ! \ (NEXT) anpassen
78 UP @ USER-AREA-SIZE @ +
79 RETURN-STACK-SIZE @ + \ An Prozeß anpassen:
80 DUP DUP RP0 ! 4 - RP ! \ RP0 & RP
81 DATA-STACK-SIZE @ + DUP SP0 ! SP ! \ SP0 & SP
82 SWAP (PARENT) ! \ (PARENT) anpassen
83 (NEST) 0! \ (NEST) anpassen
84 -ROT RP @ 2! \ Start-pfa
85 UP @ SWAP UP ! (NEXT) ! \ Vorhergehendes (NEXT)
86 R> UP ! \ an neuen Prozeß anknüpfen
87 ;
88
89 \ Runtime-Worte zur Einrichtung und Berechnung eines neuen Prozesses
90 : (||)
91 2R@ 4 + UP @ (NEST) @ ?DUP 0=
92 IF HEAP DUP DP ! THEN
93 HEAP (NEST) ! MOUNT
94 ;
95 : (EV)
96 (NEST) @ 0= IF
97 2R@ 4 + (PARENT) @ (NEXT) @ MOUNT
98 THEN
99 ;
100
101 : |( COMPILER (||) COMPILER BRANCH ?>MARK ; IMMEDIATE
102 : EV( COMPILER (EV) COMPILER BRANCH ?>MARK ; IMMEDIATE
103 : ) COMPILER STOP ?>RESOLVE ; IMMEDIATE
104 : PAR (NEST) @ IF GO-NEST DP @ (HEAP) ! (NEST) 0! THEN ;
105
106 \ Einsatz von Kanälen zum Informationsaustausch und zur Synchronisation
107 : CHAN 2VARIABLE ;
108 : C> ( CHAN -- W : Wort von Kanal holen )
109 >R BEGIN R@ @ 0= WHILE (PAUSE) REPEAT R@ 2+ @ R> 0!
110 ;
111 : >C ( W CHAN -- : Wort auf Kanal legen )
```

```

112 >R BEGIN R@ @ WHILE (PAUSE) REPEAT R@ 2+ ! R> -!
113 ;
114
115 \ Einsatz von Puffern zum wirksamen Informationsaustausch
116 \ BUFF-SIZE und MASK können größer gemacht werden!
117 16 CONSTANT BUFF-SIZE 15 CONSTANT MASK
118 : BUFF CREATE 0 , BUFF-SIZE ALLOT ;
119 : .OUT 1+ ; ( IN +0 )
120 : .BUF 2+ ;
121 : B> ( BUFF - W : Wort von Puffer holen )
122 >R BEGIN R@ C@ R@ .OUT C@ = WHILE (PAUSE) REPEAT
123 R@ .OUT C@ DUP R@ .BUF + @ SWAP 2+ MASK AND R> .OUT C!
124 ;
125 : >B ( W BUFF -- : Wort in Puffer legen )
126 >R R@ C@ 2+ MASK AND
127 BEGIN DUP R@ .OUT C@ = WHILE (PAUSE) REPEAT
128 SWAP R@ C@ R@ .BUF + ! R> C!
129 ;
130
131 : 4* 2* 2* ;
132
133 \ Definierende Worte zur Einrichtung von Wort- und Kanalfelder
134 : []CHAN
135 CREATE 4* HERE OVER ERASE ALLOT ?STACK
136 DOES> SWAP 4* +
137 ;
138 \ []WORD ist nützlicher als ARRAY von F-PC
139 : []WORD
140 CREATE 2* HERE OVER ERASE ALLOT ?STACK
141 DOES> SWAP 2* +
142 ;
143
144 MULTI
145
146 CR .FREE

```

Das nächste Listing enthält ein Programmbeispiel. Ich habe darin zur Erläuterung der Verwendung des Parallel-Forth-Pakets die Berechnung eines Skalarproduktes gewählt. Die Aufgabe entspricht der Aufgabe X4 aus Kapitel 4.3 von C.A.R. Hoares Buch "Communication Sequential Processes". Ich habe die den Forth-Worten entsprechenden OCCAM-Befehle als Kommentare hinzugefügt. Leider habe ich keinen Transputer, um die OCCAM-Programmteile auszuprobieren. Ich hoffe aber, sie gehen.

File: par-skal.seq

```

1
2 30 CONSTANT SIZE \ DEF SIZE = 30 :
3 USER VARIABLE U.I FORTH \ VAR U.I ,
4 SIZE []WORD V1 \ V1[SIZE],
5 SIZE []WORD V2 \ V2[SIZE]:
6 SIZE 1+ []CHAN N \ CHAN N[SIZE+1],
7 SIZE []CHAN W \ W[SIZE] ,
8 SIZE []CHAN E \ E[SIZE] :
9 \ VAR T1,T2,T3,T4:
10 \

```

Wortliste

MULTI	Parallelbetrieb einschalten
SINGLE	Parallelbetrieb ausschalten
(...)	Zusammenfassung von Worten (alle innerhalb der Klammern) zu einem neuen parallelen Zweig in dem zur Zeit eingerichteten (noch nicht laufenden) Prozeßring.
EV(...)	Zusammenfassung von Worten zu einem neuen parallelen Zweig in dem gerade laufenden Prozeßring, ähnlich wie die aktiven Tupel in LINDA. Die Syntax stammt aus FORTH-LINDA von Jeff Fox.
PAR	Den laufenden Prozeß anhalten und einen Ring von Tochterprozessen starten, der über einen -Konstrukt eingerichtet wurde. Warten, bis alle Tochterprozesse abgearbeitet sind, und dann den Mutterprozeß wieder aufnehmen.
PAUSE	(Nunmehr)DEFERiertes Wort. Im MULTI-Modus: Übergang zum nächsten Prozeß im gegebenen Ring. Im SINGLE-Modus: NOOP .
STOP	Beende Prozeß und entferne ihn vom gegebenen Ring.
STOP-ALL	Alle Prozesse im gegebenen Ring beenden und zum Mutterprozeß zurückkehren. Sehr nützlich zur Nachbildung des ALT-Konstrukts von OCCAM .
DATA-STACK-SIZE	
RETURN-STACK-SIZE	
USER-AREA-SIZE	Variablen zur Anpassung der Größe der Stacks und der USER-AREA. Je nach Prozeß können dafür verschiedene Werte beansprucht werden.
BUFFER	Definierendes Wort zur Erzeugung von Puffern. Puffer sind die wirksamste Art, den Datenaustausch von Prozeß zu Prozeß (ohne Synchronisation) vorzunehmen.
>B (n b --)	Ausgabe eines 16-Bit-Wertes auf Puffer b.
B> (b -- n)	Eingabe eines 16-Bit-Wertes von Puffer b.
CHAN	Definierendes Wort zur Erzeugung von Kanälen. Ein Kanal ist ein Puffer von vorgegebener Wortlänge. Kanäle sind für den Austausch von Informationen mit Synchronisation gut.
[]CHAN	Definierendes Wort zur Einrichtung eines Feldes von Kanälen
>C (n c --)	Ausgabe eines 16-Bit-Wertes auf Kanal c.
C> (c -- n)	Eingabe eines 16-Bit-Wertes von Kanal c.
[]WORD	Genau wie ARRAY in F-PC und mit demselben Ziel.



```
11 : VECT-MULT                \ PROC VECT.MULT =
12                             \ SEQ
13   SIZE 0 DO                 \ U.I = [0 FOR SIZE]
14     I 1+ I V1 !             \ V1[U.I] := U.I + 1
15     I 1+ I V2 !             \ V2[U.I] := U.I + 1
16   LOOP                       \ PAR
17   SIZE 0 DO I U.I !         \ U.I = [0 FOR SIZE]
18     I U.I !                 \ PAR
19     ||( U.I @ V1 @ U.I @ W >C ) \ W[U.I] ! V1[U.I]
20     ||( U.I @ V2 @ U.I @ E >C ) \ E[U.I] ! V2[U.I]
21     ||(                       \ SEQ
22       U.I @ W C>           \ W[U.I] ? T1
23       U.I @ E C> *         \ E[U.I] ? T2
24       U.I @ N C> +         \ N[U.I] ? T3
25       U.I @ 1+ N >C       \ N[U.I+1] ! T1*T2+T3
26     )                       \
27   LOOP                       \
28   ||( 0 0 N >C )           \ N[0] ! 0
29   ||( SIZE N C> U. )       \ SEQ
30                             \ N[SIZE] ? T4
31                             \ WRITE(T4) :
32   PAR
33 ;
34
```

Michael Montvelishsky stammt aus Saransk, 600 km südöstlich von Moskau. Er machte 1982 an der Mordwinischen Landesuniversität sein Diplom in Elektrotechnik. 1989 promovierte er daselbst. Mit Forth arbeitet er seit 1988. Es ist seine Liebessprache, aber "gelegentlich programmiere ich auch in C und PASCAL". Michael Montvelishsky ist Dozent an der Universität von Mordowien und verwendet Forth in seinem Programmierkurs.

Das zweite Programmbeispiel enthält etwas weniger an Parallelverarbeitung, ist dafür aber etwas mehr von Nutzen. Es handelt sich dabei um eine Weckuhr. Man gebe ein:

21 00 ALARM" Es ist Zeit, nach Hause zu gehen!"

und schon kann es losgehen mit dem Experiment in Parallel-Forth !

File: par-uhr.seq

```
1
2 \ Weckuhr
3
4 ANEW ALARM
5 HIDDEN ALSO EDITOR ALSO
6
7 USER
8 VARIABLE ALARM-HM
9 VARIABLE ALARM-STRING
10 FORTH
11
12 : ALARM" ( stunden minuten | <text>" )
13   SWAP FLIP + 0 0. B>T DROP ALARM-HM ! \ Zeit holen
14   HERE , ALARM-STRING ! \ Wecktextstringholen
15   EV(
16     ALARM-HM @
17     BEGIN                       \ Warten ...
18       1000 FOR PAUSE NEXT       \ Raus zur besseren Ausnutzung
19       DUP GETTIME DROP U<=
20     UNTIL
21     DROP SINGLE                 \ Parallelverarbeitungsaufheben
22     SAVESCR SAVECURSOR          \ Bildschirm und Cursor aufbewahren
23     DKGRAY >BG WHITE >FG       \ Farben wählen
24     TRUE ALARM-STRING @ COUNT ?SOFTERROR \ Alarm !!
25     RESTCURSOR RESTSCR MULTI    \ Bildschirm und Cursor wieder holen
26   )                               \ Parallelverarbeitungeinschalten
27 ;
28
29 ONLY FORTH ALSO DEFINITIONS
30
```

Gehaltvolles

zusammengestellt und übertragen von Fred Behringer

Het Vijgeblad 1, Notausgabe von der HCC Forth-Gebruikersgroep, Holland

Die holländische Forth-Gebruikersgroep ist immer noch auf der Suche nach einem Redakteur. In der Zwischenzeit gibt der Vorsitzende, Willem Ouwerkerk, eine Notausgabe heraus: Het Vijgebladje (Das Feigenblättchen).

Eine interessante Erscheinung tut sich auf: Auf den knapp 4 Seiten des Feigenblättchens steht nicht etwa nur ein Zehntel dessen, was im Feigenblatt auf mehr als 40 Seiten stand. Der Zwang zur Konzentration leistet Erstaunliches:

2 Over ANSI Forth Albert Nijhof

Sieben Jahre haben die dazu gebraucht Umfragen, Antworten langwierig jeder konnte mitmachen, aber das Sagen haben Vertreter amerikanischer Großfirmen Gibt es eigentlich schon ein ANSI-Forth ? Die Forth-Gebruikersgroep wird das herausfinden Lesen Sie das Feigenblättchen...

3 Analooq naar digital met 8051-ANS-Forth

Willem Ouwerkerk

Es geht um das 8051-ANS-Forthkurs-Projekt der Forth Gebruikersgroep. Der 8051 hat alles für die AD-Wandlung Nötige an Bord. Den Rest erledigt das 8051-ANS-Forth.

4 LCD aansturing met de BAMBOE

Willem Ouwerkerk

Aus dem 8051-ANS-Forthkurs. Es geht um die Ansteuerung von LCD-Displays über die BAMBOE-Karte. Das ist eine 16-Bit-E/A-Karte, die an den Parallelport eines weitgehend beliebigen Computers angeschlossen werden kann (50,- Gulden). Besprochen wird die Funktion OR unter 8051-ANS-Forth.

□



Forth International

von Fred Behringer



Holland: Forth auf der Suche nach einem Redakteur

In der Nummer 46/47 (1.Quartal 1995) von "Het Vijgeblad", der Clubzeitschrift der holländischen "Forth-Gebbruikersgroep" schreibt der (scheidende) Redakteur, Lennart Benschop: "Das ist die letzte von mir redigierte Nummer und vielleicht sogar auch die allerletzte überhaupt keine Artikel keinen neuen Redakteur Het Vijgeblad gestorben, Forth bestimmt nicht viele ihr eigenes Forth geschrieben, keiner einen eigenen Cobol-Compiler Forth ist überall, vom 8051 bis zur Unix-Workstation"

Amerika: "Definitions", eine neue Forth-Zeitschrift

Willem Ouwerkerk macht in "Het Vijgeblad" Nr.46 die folgende kurze Mitteilung, die ich hier in Übersetzung wiedergebe:

In Amerika ist eine neue Zeitschrift über Forth und forth-artige Sprachen herausgekommen, die sich "Definitions" nennt. Sie wird sechsmal pro Jahr erscheinen und kostet für Besteller außerhalb Amerikas 30,- Dollar pro Jahr. Als Redakteur zeichnet der vielen bekannte Lawrence P.G. Forsley. Die erste Nummer enthält folgende Artikel:

- * *Oberservation, Feyman on Top*
- * *Down Design*
- * *1993 Rochester Forth Conference*

on Process Control

- * *Vendors column: New Micros, Inc*
- * *Forth in Space*
- * *Vendors column: Dash, Find & Associates*
- * *Consultant's Spotlight: Gary Bergstrom*
- * *Forth, C and C++ (part 1 of 3)*
- * *Paleography of Extensible Languages (part 1 of 2)*
- * *Postscript vs. Forth (part1)*

Die Zeitschrift wendet sich an "Fachleute" und ist dementsprechend auch auf Firmeninserate eingestellt. Alles in allem macht diese Forth-Zeitschrift einen recht guten Eindruck. Redaktionsadresse:

Institute for Applied Forth Research
70 Elmwood Avenue
Rochester, NY, 14611

FORTHWRITE, die Zeitschrift unserer englischen Nachbarn

Im "Het Vijgeblad" 43 berichtet Marcel Hendrix von der "HCC-Forth-Gebbruikersgroep" in Holland über die Zeitschrift FORTHWRITE der englischen Forth-Gruppe FIGUK. Ich übersetze den ersten Abschnitt und gebe den Rest im Telegrammstil wieder:

"FORTHWRITE hat das Format (DIN-) A5, ist also $\sqrt{2}$ mal kleiner als unser Vijgeblad (Anmerkung des Rezensenten: Das Feigenblatt ist DIN-A4-groß), oder anders gesagt, so lang wie

das Vijgeblad breit ist. Es (FORTHWRITE) enthält durchschnittlich 35 Seiten. Die Zeitschrift wird nicht mit einem Programm wie LaTeX oder Ventura mit speziellem Layout angefertigt, die Artikel werden so abgedruckt, wie sie die Mitglieder einreichen. Ich kann Ihnen nur sagen, im Vereinigten Königreich sind noch eine ganze Reihe von 9-Nadel-Druckern in Umlauf, wahrscheinlich noch mit ihren ersten Farbbändern Eddy and Tu (editorial and tutorial) Leserbriefe, Mitteilungen mindestens zwei Seiten Anzeigen der Rest eingereichte Artikel Niveau hoch Anfänger scheint es nicht zu geben keine F-PC-Anhänger größtenteils 9-Nadler mit FIG-Forth, die anderen ATARIsten Alte Garde mit ANS-Forth wenig im Sinn....."

Fred Behringer, München

Ich möchte in dieser Rubrik in unregelmäßiger Folge über Forth-Aktivitäten in anderen Ländern berichten. Ich bin fleißig am Sammeln, würde mich aber natürlich auch über Zusendung von Material von anderer Seite freuen.

Fred Behringer
Planegger Str. 24
81241 München.

Verständlich sind für mich die Sprachen Englisch, Holländisch, Italienisch und Französisch.



Nun ist auch 1995 beinahe oder auch doch schon Vergangenheit. Die Eindrücke von der euroFORTH-Konferenz, die 1995 mit Unterstützung der GI (Gesellschaft für Informatik) Fachgrup-

werden. (Hiermit winke ich wieder einmal mit dem Zaunpfahl!)

Es läßt sich ohnehin feststellen, daß die VD als vierteljährliches Vereinsmagazin natürlich eine wesentlich gemächlichere Berichterstattung macht

als es die Diskussionen in Usenet sind. So ist zwar, zur der Zeit als ich diese Zeilen noch reichlich Zeit an dem in comp.lang.forth von

Rick VanNorman (rick@whytel.com) ausgeschrieben ANS-Forth Programmierwettbewerb teilzunehmen - aber wenn diese Zeilen gelesen werden, ist der Wettbewerbsschluß schon lange Vergangenheit. Aber nichts desto trotz: Alle Standard Forth Programmierer sind aufgerufen, das ihrer Meinung nach beste ANS-Forth Programm, das das Spiel Sokoban implementiert, bis zum 1. Dezember 1995 bei Rick VanNorman einzureichen. Eine unabhängige Jury nimmt sich der Einsendungen an und prämiert sie. Zu gewinnen gibt es das Taygeta Forth CD-ROM, das Mitte Dezember produziert sein soll und neben zahlreichen Forth bezogenen Files und einem Abzug des Taygeta-FTP/WWW-Archivs eben auch die Lösungsvorschläge des Programmier-

*Jetzt
geht's
ANS
FORTH!*

von Ulrich Hoffmann



File: ansi0495.lst

pen 2.1.4 (Alternative Konzepte für Sprachen und Rechner) und 3.5.5 (Architektur für hochintegrierte Schaltungen) im Informatik-Tagungszentrum Schloß Dagstuhl bei Saarbrücken stattfand, sind noch frisch. Nun - ich kann die Standardkritiker beruhigen: Standard Forth ist weitgehend akzeptiert aber eine daraus resultierende Einschränkung des Ideenreichtums bei der Verwendung von Forth kann ich nicht beobachten. Freizügig werden auch Konzepte eingesetzt, bei denen der Standard ausdrücklich keine Aussagen macht. Auf der anderen Seite ist es gut möglich gewesen die Standard-Begriffswelt in vielen Diskussionen zu verwenden und auch damit verstanden zu werden (jedenfalls meistens). Gut, wenn Begriffe wie 'word list' oder 'execution token' Folklore werden.

Es ist schon eine Weile vergangen, seit dem wir in de.comp.lang.forth nach deutschsprachigen Entsprechungen zu den im Standard verwendeten Begriffen gesucht haben. Recht schnell konnten wir uns auf eine Liste einigen in der rund 80% der Begriffe im Konsens übertragen werden konnten. Ich denke, diese Liste wird in der kommenden Vierten Dimension zu finden sein, und so als Ergebnis unser Diskussion auch den nicht in de.comp.lang.forth vertretenen Forth-Interessierten bereitgestellt

```

1
2 : CASE 0 ; IMMEDIATE
3 : ESAC BEGIN ?DUP WHILE [COMPILE] THEN REPEAT ; IMMEDIATE
4 : OF COMPILE OVER COMPILE = [COMPILE] IF COMPILE DROP ; IMMEDIATE
5 : ORIF COMPILE ?DUP COMPILE 0= [COMPILE] IF ; IMMEDIATE
6 : ANDIF COMPILE DUP [COMPILE] IF COMPILE DROP ; IMMEDIATE
7
8 "If the first two definitions don't work, then the following
9 should." W. Baden.
10
11 VARIABLE case-sys
12 : CASE case-sys @ DEPTH case-sys ! ; IMMEDIATE
13 : ESAC
14 BEGIN DEPTH case-sys @ > WHILE [COMPILE] THEN REPEAT
15 case-sys !
16 ; IMMEDIATE
17
18 Und so wird es benutzt:
19
20 : range ( x lo hi -- x flag ) 2>R DUP 2R> WITHIN ;
21
22 : char-class ( c -- )
23 CASE DUP 32 U<_>ORIF DUP 127 = THEN
24 IF DROP ." control character "
25 ELSE 32
26 OF ." space "
27 ELSE CASE
28 48 58 range ORIF 65 91 range ORIF 97 123 range
29 ESAC
30 IF DROP ." alphanumeric "
31 ELSE 128 0 RANGE
32 IF DROP ." not ascii "
33 ELSE DROP ." special character "
34 ESAC
35 ;
36

```

Längste Zeile hat 68 Zeichen



wettbewerbs enthalten soll. Man darf gespannt sein.

“Welches Forth System soll ich nehmen, wenn ich ANS-Forth Programme schreiben will?” Diese Frage wird mir nun doch immer wieder gestellt. Nun - ich selbst verwende dafür überwiegend das ‘Portable Forth Environment’ PFE von Dirk Zoller, das ich zur Zeit unter SUN-OS, dem Unix der Sun Workstations, und unter Linux (freies PC-Unix) verwende. Es läuft auch unter DOS und OS/2. PFE ist ein klassisch Textbasiertes sehr vollständiges ANS Forth Standard-System. Größtes Manko ist die weitgehend unvollständige Dokumentation. Um Standard-Programme zu schreiben, eignet sich PFE dennoch vorzüglich, da ja das Standard-Dokument die dafür benötigten relevanten Informationen bereithält. Ähnlich verhält es sich mit F68KANS, das als Standard-System zur Wahl steht, wenn es sich um Motorola 68xxx basierte (Desktop- oder eingebettete) Systeme handelt. Wem mehr der Sinn nach einem Handbuch steht, der kann auf bigFORTH (B. Paysan) oder iFORTH (M. Hendrix) zurückgreifen. Diese bei-

den preiswerten kommerziellen Standard Systeme laufen z. Zt. beide mit Hilfe eines DOS-Extenders als 32-Bit-Programme unter DOS. Wem mehr der Sinn nach MS-Windows steht, der kann mit Win32For, das nunmehr wohl F95 heißt, ANS-Forth konform Win32 (Windows 3.x mit Win32s; Windows95 oder Windows NT) erobern. F95 basiert auf einem ANS-Forth konformen 32Bit Kernel, den Andrew McKewan (mckewan@netcom.com) geschrieben hat, und es wird imitativ von Andrew und von Tom Zimmer (Mr. F-PC himself, ZForth@ix.netcom.com) entwickelt. Bei mir hat F95 zur Zeit das Prädikat “Hacker’s delight”. Wie vollständig es dem Standard genügt, entzieht sich meiner heutigen Kenntnis. Vielleicht kann ich darüber in einer der kommenden VDs berichten. Da F95 erklärtermaßen die Nachfolge von F-PC antreten soll, steht zu erwarten, daß es ähnlich hohen Zuspruch wie F-PC finden wird und wie dieses hoffentlich hohe Qualität und künftig eine ebenso vernünftige Dokumentation wird aufweisen können.

An den Standard-Grundlagen diskutieren Wil Baden (wilbaden@netcom.com)

und Jonah Thomas (JETHomas@ix.netcom.com) zur Zeit über das bessere CASE-Statement für zukünftiges Standard Forth. Wil Baden verfolgt dabei eine interessante Variante: Er bietet heute zwei Versionen einfachen Standard Source Codes mit Environmental Dependencies an. Beide Versionen sind zwar prinzipiell nicht vollständig portabel über alle Standard Systeme, aber dennoch paßt auf den zur Zeit verfügbaren Standard Systemen die eine oder andere Variante. Zum Selberexperimentieren hier die beiden CASE-Implementierungen:

Soviel vielleicht aus der “Grundlagenforschung”. Viel Spaß beim Knobeln.

Insgesamt ist meine Einschätzung, daß die Zeit der Begeisterungstürme für den Standard auf der einen Seite und der Stürme wehementer Ablehnung auf der anderen Seite weitgehend vorbei ist. Der Standard hat sich als solcher etabliert, seinen Platz in der Forth Gemeinde gefunden und ist auf vielen Ebenen akzeptiert.

May the Forth be with you!



Forth-Systeme:

hForth 0.95

I uploaded HF86V095.ZIP to Taygeta. hForth is small ANS Forth Standard system. hForth EXE model for 8086 is comparable ZEN and Pygme. I beleive that mine is better than them. :)

FILE and BLOCK words for MS-DOS were added as well as many bugs were fixed since v0.9.2. Now you can build application programs by redirecting "boot". Also I made all VALUE words and VARIABLE words IMMEDIATE with '@', '!', '+!' for EXE model. There is no speed advantage for VALUE words over VARIABLE words and choice between VALUE and VARIABLE is pure matter of style. You can find more detailed description about it and how to make stand-alone application in the following attached README file. I pick up 'h' in hForth for Han which means Korean in Korean language. Please let me know if you know the name hForth is used already by someone else.

wykoh@pado.kriect.re.kr (Wonyong Koh)
in /comp/lang/forth, nov'95
fax: 82-42-861-4245, South Korea

[Das angesprochene README file erwähnt, daß hForth ROM-fähig ist, die beta release frei verteilt werden darf, auf eForth beruht. Inkompatibilitäten mit ANS seien denkbar, alle Core words sind vorhanden (sogar mit dem Testprogramm der Johns Hopkins Universität getestet). Weitere ANS-Wortgruppen: TOOLS words, SEARCH ORDER words, SEARCH ORDER EXTENSION words. Das RAM von hForth wurde auch auf Z80 portiert. /clv]

Taygeta lebt

The changes at 'taygeta' are bascially complete.

The new FIG home page is at:
<http://www.taygeta.com/fig.html>
my Forth page is at:
<http://www.taygeta.com/forth.html>
my home page is at:
<http://www.taygeta.com/skip.html>
[jmrubin@ix.netcom.com (Joel Rubin)
ergänzt: You left out the ftp address:
<ftp://taygeta.com/pub/ForthNote> that there is no "ftp" in the domain address.]
skip@taygeta.com (Skip Carter) in
/comp/lang/forth, oct'95

2PW



Nur noch bis 15. Dezember..

kann man das Forth-Weihnachtspäckchen bestellen.

Die CD 'forthige Weihnacht 95'

enthält u.a. den kompletten Ftp-Server Taygeta aus USA mit bald 100 Mb; den kompletten Ftp-Server Asterix aus Portugal mit zahlreichen Forth-Systemen und in der Sammlung von Ulrich Hoffmann befinden sich sicher auch einige ANS Quellen, um dem neuen Standard zu Leibe zu rücken.

Sofort bestellen!



Forth-Gesellschaft: Das Forth-Büro

Leider sind diese Nachrichten nicht besonders interessant und leider auch kein FORTH-Programm, aber nicht ganz unwichtig für die FG. ok

Die Reaktionen auf den Datenbankauszug, den wir zur Korrektur der letzten VD beigelegt haben, war erfreulich. 13,47 % der Mitglieder haben uns geholfen die Mitgliederliste zu aktualisieren, dafür vielen Dank. Von den restlichen 86,53 % müssen wir annehmen, daß sie immer noch ihre alten Interessen, ihren alten Computer (C64 ;-)) und ihr (volks-;-)FORTH haben.

Adressänderungen: Wenn Sie in der nächsten Zeit einen Umzug oder gar einen Auslandsaufenthalt planen, vergessen Sie bitte nicht dies dem FORTH-Büro zu melden, telefonisch, per Fax oder als Brief, alles nehmen wir an. Es gibt Mitglieder, deren Jahresbeitrag wir am Jahresanfang abgebucht haben und die von uns jetzt keine VIERTE DIMENSION mehr zugeschickt bekommen können, da die Deutsche Post AG die letzten VDs, mit der Bemerkung "unbekannt verzogen" an uns zurückgeschickt hat! Ich staple diese Rückläufer hier im Büro noch bis zum Jahresende. Die Betroffenen können das natürlich wieder nicht lesen.....

Nachweise zur Beitragsermäßigung: Einmal pro Jahr ist es leider unumgänglich auch an das FORTH-Büro einen Nachweis zur Berechtigung für den ermäßigten Mitgliedsbeitrag zu schicken, ausgenommen von dieser Nachweispflicht sind Rentner, wenn Sie das entsprechende Rentenalter nachgewiesen haben.

Mitgliedsbeiträge: Die Mitgliedsbeiträge haben sich nicht verändert:

Schüler, Auszubildende, Studenten, Arbeitslose, Rentner 64,00 DM

ordentliche Mitglieder 96,00 DM

fördernde Mitglieder, Institutionen und Firmen 176,00 DM

Die Bankgebühren sind allgemein sehr hoch, besonders teuer kommt uns allerdings ein Lastschrift-Rückruf, weil meistens beide Banken (die Bank des Mitglieds und unsere Postbank) ihre Gebühren dafür verlangen, d.h. insgesamt 15,00 - 20,00 DM

Gemeinnützigkeit: Die FORTH-Gesellschaft eV ist ein gemeinnütziger Verein, d.h. sie ist von der Körperschaftsteuer, Gewerbesteuer und Vermögensteuer befreit. Der Freistellungsbescheid wurde dem FORTH-Büro vom Finanzamt München für Körperschaften am 24. Mai 1995 zugestellt. Nach diesem Freistellungsbescheid ist die FORTH-Gesellschaft eV berechtigt entsprechende Spendenbestätigungen für steuerliche Zwecke auszustellen. Die Berechtigung gilt unter dem Vorbehalt des Widerrufs bis zum 31.12.1997. Der Bescheid erging jedoch unter der Auflage, da unsere Satzung an die aus steuerlichen Gründen notwendigen Bestimmungen der Mustersatzung für gemeinnützige Vereine angepaßt wird. Diese Satzungsänderung muß im Rahmen der nächsten Mitgliederversamm-

lung durchgeführt werden. Die Mitgliedsbeiträge sind wie Spenden abziehbar, und als Nachweis gilt bis 100,00 DM der Bankbeleg.

Angebote der FORTH-Gesellschaft eV für ihre aktiven Mitglieder: Ein Verein lebt durch die Mitwirkung möglichst vieler Mitglieder

- kommen Sie zu unseren FORTH-Tagungen
- diskutieren Sie mit in unserer Mailbox
- schreiben Sie einen Artikel oder Anregungen zu FORTH, zu Spezialgebieten oder zum Verein
- haben Sie schon Ihr Firmenprofil in der VD veröffentlicht; es kostet nicht die Welt und bringt sicher neue Kontakte?
- haben Sie schon unseren Mikrocontroller-Verleih frequentiert?
- haben Sie vielleicht sogar eine Literaturliste zu und über FORTH, die das FORTH-Büro bei Anfragen weitergeben könnte?

*Danke für das Lesen dieser Zeilen.
Ein erfolgreiches 1996 mit FORTH wünscht Ihnen Ihre
Ulrike Schnitter
- FORTH-Büro -*

Telefonisch erreichen Sie uns unter 089-317 37 84 (Anrufbeantworter und Fax, wir rufen gerne zurück, aus Kostengründen lieber nach 18:00 Uhr).

Wo ist das Forth-Magazin?

Immer wieder werden wir gefragt: Wo gibt es eigentlich die VD? Ulrike Schnitter gibt uns die Antwort:

unsere VD liegt natürlich bei unseren Mitgliedern und

- bei der Universitätsbibliothek Hannover und
- bei der Bayerischen Staatsbibliothek München und
- bei der FIGUK Grossbritannien und
- bei der FIG in USA und
- bei der FIG Dutch Holland und
- bei den Abonnenten ??? und selbstverständlich im FORTH-Büro.

Griße Ulrike

Forth-Gesellschaft: Microcontrollerverleih

Neu im Vertrieb:

- VD95-1 - Diskette zum Forthmagazin
- VD95-2 "
- VD95-3 "
- Dokumentation ANS X3.215
- 1994 im Word 2.0-Format
- Crossassembler für Z8 und PIC von Bernd Hinze
- EFORTH für PIC 1742

je Diskette: 10,-DM

Ing. Büro Klaus Kohl
PF 1173 Mering
Zeppelinstr. 10
86415 Mering
Tel.: 08233 / 30 524
Fax.: 08233 / 99 71

Seit 1.1.1995 hat Thomas Prinzen Microcontroller-Verleih

übernommen. Bis jetzt hat er zwei Anfragen bekommen.

Dem Verleih stehen folgende unterschiedliche Geräte zur Verfügung: Super-8 / Mini-Muck und zur Abschreckung 8052-AH-Basic. Es sind jeweils enthalten eine Leiterplatte mit dem entsprechenden Microcontroller, Netzteil, Beschreibung und Software. Demnächst wird noch ein 68HC11 hinzukommen. Dieser wird zur Zeit von Rafael Deliano zusammengebaut.

Der F65k wird in alter Form nicht mehr hergestellt und ist nicht mehr erhältlich.

Wer dem Verleih Geräte zur Verfügung stellen könnte, sei hiermit ermuntert.

Forth-Gesellschaft: Forth-Magazin

Das Forth-Magazin sucht ...

An Sachmitteln fehlen noch:

- Ein Englisch-Wörterbuch

An Hardware fehlt noch:

- Ein Rechner, schneller als 386dx33
- Eine Gigabyte-Festplatte
- Ein A4-Scanner

An Software fehlt noch:

- Ein DTP-System, teurer als 50 DM

- Ein Archivsystem, vielleicht auf Basis des F-PC-Hypertextes?

An Beratern fehlen noch:

- Spezialisten aus dem Satz-, Grafik-, Layoutbereich
- DTP-Erfahrene zur Beratung bei der Neuanschaffung
- Wer kann mir Hinweise zum Belichten von Diskette geben?
- Wer kennt sich mit Macs aus und kann beim Anschaffen helfen

Claus 030 / 782 81 79



Umlaute in den Namen von Forth-Worten

von Fred Behringer

Planegger Str. 24, D-81241 München

Eigenartigerweise dürfen wir Deutsche in Forth keine Umlaute (ä,ö,ü) verwenden. Wir wissen alle, woher das kommt. Die Franzosen haben ähnliche Probleme mit den Akzent-Buchstaben. 1988 habe ich unter dem Motto "Forth ist für alle da - auch für uns Deutsche" in der Zeitschrift JEDI der gleichnamigen französischen Forth-Gruppe einen Leserbrief über das Umlautproblem geschrieben, der Leserbriefe von anderen zur Folge hatte [1]. Bei JEDI ging es mir um UPC. (Ich spreche in diesem Beitrag über Turbo-Forth oder F83 und über mein Transputer-Forth-System F-TP.) Ulrich Hoffmann [5] hat 1991 das Umlautproblem in bezug auf den Editor in F-PC besprochen, Claus Vogt [7] hat 1992 und Arndt Klingelberg [6] 1993, sie arbeiten alle mit F-PC (!), die Frage ganz allgemein behandelt.

Stichworte: Umlaute, F-TP, Turbo-Forth, F83

Es ist nicht neu

Das Problem mit den Umlauten in der amerikanisch orientierten elektronischen Datenverarbeitung ist nicht neu. Ich habe mich 1962 bei EURATOM im Zusammenhang mit der Automatischen Dokumentation damit beschäftigt. Damals waren es lochstreifengesteuerte Schreibmaschinen vom Typ Flexowriter, deren Umlautverhalten uns nicht gefiel.

Tastatur und Editor kein Problem

UPC habe ich für mein Transputer-Forth-System von Anfang an so modifiziert, daß bei CAPS ON ä in Ä übergeht, ö in Ö und ü in Ü. Diese Veränderung erfordert keinen großen Aufwand und stört auch in einem System ohne Umlaute nicht. Das Editorproblem besteht für mich nicht, da ich in meinem Transputer-Forth-System (ganz in Turbo-Forth-Manier) schnell aus dem System heraus einen "Fremdeditor" aufrufen kann und von dort aus dann genauso schnell wieder ins Transputer-Forth-System zurückgelange. Tastatureingaben waren wegen der Hostsystem-Transputersystem-Beziehung sowieso so stark zu beachten, daß ich mich nicht zu scheuen brauchte, das Umlautproblem gleich mit zu bereinigen. Die Tastatur selbst läuft bei mir über den Gerdes-

Treiber [2],[3],[4], den ich für meine Zwecke modifiziert und ergänzt habe.

Umlaute in den Wort-Namen

Was für den vorliegenden Beitrag übrigbleibt, ist das Problem mit Zeichen von ASCII-Werten über 127 in den Namen von Forth-Worten. Es ist TRAVERSE, das Schwierigkeiten macht, indem es nach einer 1 im höchsten Bit des letzten Zeichens im Namensfeld sucht. Hinzu kommt FIND und die beiden Worte >NAME und NAME>, die beide TRAVERSE enthalten. Schließlich benötigt auch .ID das besagte 1-Bit, um zur ASCII-7 gerechten Anzeige der Namen zu gelangen. Das ist dann aber auch schon (fast) alles.

Meine Lösung

Im Gegensatz zu anderen Autoren brauche ich in meinem Transputer-Forth-System F-TP nichts zu patchen. Das System ist immer noch in der Entwicklungsphase, wenn auch schon im Endstadium, und es bereitet überhaupt keine Schwierigkeiten, die Mühle des Metacompilierungsprozesses immer wieder von neuem mahlen zu lassen. Es folgt meine Lösung:

Vektorisieren statt Patchen

Ich DEFERiere die obengenannten Forth-Worte und ersetze sie jeweils durch ein Paar der Form XXX-ENG und XXX-GER. Zwei Umschalte-Worte, ENGLISH und GERMAN, sorgen dafür, daß die eine oder andere Sorte eingeschaltet wird. Wird ENGLISH eingegeben, dann verhält sich das System so, wie wir es von Turbo-Forth oder F83 her kennen: ASCII-7 konform. Wird GERMAN eingegeben, dann werden in dem so entstehenden System alle ASCII-8 Zeichen in den Namen zugelassen. WORDS, SEE und derartige Werkzeuge verarbeiten das ohne Murren.

Byte mit Länge auch am Ende

Das einzige Zugeständnis, das das ursprüngliche System, also das, zu dem man über ENGLISH zurückgelangt, an das über GERMAN einzuschaltende System, das sämtliche ASCII-Zeichen in den Namen akzeptiert, in meinem F-TP macht, ist die Einführung eines Extrabyte am Ende des Namensfeldes. Genau wie das erste Byte des Namensfeldes enthält auch dieses die Stringlänge. Das reicht dann, um bei >NAME und dergleichen vom letzten auf das erste Byte zu gelangen, ohne die 1 im höchsten Bit der Namensbytes testen zu müssen.

Extrabyte ist zusätzliches ALIGN-Füllbyte

Das Extrabyte am Ende des Namensfeldes fällt nicht weiter auf, da ich ja sowieso auf die 32-Bit-Wortgrenzen des Transputers T800 achten mußte und an vielen Stellen ein (voll funktionstüchtiges) ALIGN eingefügt habe. (Mein Transputer-Forth-System F-TP ist natürlich ein 32-Bit-System.) Das Extrabyte wird als zusätzliches Füllbyte betrachtet. Die Füllbytes werden mit dem Wert 32 d besetzt. Das ist ein Wert, der von keiner Namenslänge erreicht wird. Also kann ich beim Übergang von cfa zu nfa (bei >NAME) erkennen, wo die Füllbytes aufhören und wo das Namensfeld, von hinten her, anfängt.

Kurzbeschreibung meines Transputer-Forths

Bevor ich nun die einzelnen Worte der hier zu besprechenden Umschalt-Vorrichtung vorstelle, möchte ich noch kurz ein Bild meines Transputer-Forth-Systems zeichnen.

Server, Metacompiler und Cross-Assembler in Turbo-Forth

Im Hostsystem (PC) habe ich einen Server und einen Metacompiler eingerichtet. Beide sind in Turbo-Forth (16 Bit) geschrieben. Turbo-Forth (seit 1988) stammt von Marc Petremann et al. von der französischen Forth-Gruppe JEDI. Turbo-Forth gründet sich auf F83, ersetzt aber das Screen-Konzept durch eine konsequente DOS-Datei-Anbindung. Der Server enthält einen Cross-Assembler, ebenfalls in Turbo-Forth programmiert. (Der Cross-Assembler läuft im Host-Forth-System und erzeugt im Transputer-RAM Maschinencode.)

Cross-Metacompilation per Transputer-PEEK und -POKE

Der Metacompiler, eigentlich ein "Cross-Metacompiler", sorgt mit Hilfe des Cross-Assemblers für die Metacompilation des ersten Teils des Transputer-Forth-Systems, TKERNEL.FTH. Der Datenverkehr wird über ein Transputer-Link-Adapter mit Hilfe des Servers abgewickelt und zwar während der gesamten Metacompilation über den PEEK-POKE-Mechanismus des Transputers.

Aus 16 Bit mach 32 durch Patchen von CONVERT

Der Übergang von 16 Bit zu 32 Bit wird durch ein zeitweiliges Patchen von CONVERT im Host-Forth-System erreicht, und zwar so, daß alle einfachen Werte als doppeltgenaue interpretiert werden und doppeltgenaue im System-Quelltext, also in TKERNEL.FTH, für die Zeit der Metacompilation untersagt bleiben.

Zweiten Teil compilieren, nicht mehr metacompilieren

Nach der Metacompilation von TKERNEL.FTH wird der zweite Teil des System-Quelltextes, TCORE.FTH, vom dann schon funktionsfähigen F-TP-Kernsystem per INCLUDE hinzugeladen und schon compiliert - compiliert, nicht mehr metacompiliert.

Fertiges F-TP als Binärdatei jederzeit einladbar

Das fertige System F-TP wird per SAVE-SYSTEM TFORTH.FYS in eine DOS-Binärdatei gespeichert. Vom SERVER im Hostsystem aus kann diese dann jederzeit per LOAD-SYSTEM TFORTH.FYS in das Transputer-RAM geladen werden. Dieses Hinzuladen wickelt sich per Batch-Dateien von der DOS-Ebene aus automatisch ab.

ESC-Zwischenlandung Server im Host

Per [ESC] oder per BYE [RET] kann das Transputer-Forth-System jederzeit unterbrochen und verlassen werden. Man landet dann im Host-Forth-System, und zwar im Server. Von dort aus gibt es zwei Möglichkeiten.

F-TP-Fehleranalyse vom Host-Server aus

Entweder (1) (nach eventuellen Untersuchungen, z.B. per TDUMP, des Transputer-Forth-Systems von Server aus) per START [RET] wieder ins Transputer-Forth-System springen oder (2) das Hostsystem per BYE [RET] verlassen und später gegebenenfalls per TRESUME [RET], TRESUME.BAT ist eine Batch-Datei, von der Dos-Ebene aus wieder ins Transputer-Forth-System springen. Beide Male wird das Transputer-Forth-System wieder so angegangen, wie es verlassen wurde, also unter Beibehaltung der Stack- und Return-stackinhalte, der Variablenwerte usw.

Forth-Worte zur Umschaltung von Englisch nach Deutsch

Ich hätte natürlich mein System F-TP gleich zu einem "deutschen" System umbauen können. Es stört ja nicht. Und wer keine Umlaute verwenden will, braucht das ja nicht zu tun. Und ob nun ein Forth-System unter Verwendung von TRAVERSE aufgebaut wird oder nicht, das kann der Sprache selbst, also Forth 83 oder den ANS-Weiterentwicklungen, eigentlich egal sein. Und wer TRAVERSE verwendet, greift sowieso ins Systemgeschehen ein. Und das kann gar nicht anders als implementationsabhängig sein. Ich dachte aber, es ist sicherer, für eine gewisse Zeit, also sozusagen in der Betatestphase, ein Zurückschalten in die von F83 und Turbo-Forth her gewohnte Umgebung zu ermöglichen. Es folgen die für eine solche Umschaltung modifizierten oder neu hinzugenommenen Worte.

HEADER

Im Metacompiler TRANSMET.FTH wurde HEADER so geändert, daß jetzt in TKERNEL.FTH in den Namensfeldern ein Extrabyte angehängt wird, in dem (ein zweites Mal) die schon im ersten Byte des Namensfeldes enthaltene Länge des Namens enthalten ist. Die 1 im höchsten Bit des davorliegenden Bytes bleibt zunächst stehen, d.h., der Metacompiler erzeugt ein "englisches" System, das keine Umlaute in den Namensfeldern zuläßt.

"CREATE

Analog zu HEADER im Metacompiler wurde auch "CREATE in TKERNEL.FTH so geändert, daß jetzt Extrabytes an die Namensfelder angehängt werden. HEADER wirkt sich auf die Forth-Worte in TKERNEL.FTH aus, "CREATE wirkt sich auf die Forth-Worte in TCORE.FTH (das ja per INCLUDE hinzugeladen wird) und auf alle eventuellen Anwendungsworte aus. Das Extrabyte stört im "englischen" System nicht, jedoch mußte eine Version geschaffen werden, die nach Umschalten per GERMAN bei den neu zu erzeugenden Worten die 1 im höchsten



Bit des letzten Zeichens im Namensfeld wegläßt. "CREATE wurde also vektorisiert und per ENGLISH nach "CREATE-ENG und per GERMAN nach "CREATE-GER umleitbar gemacht.

(FIND)

In TKERNEL.FTH wurde (FIND) DEFERiert und per ENGLISH nach (FIND)-ENG oder per GERMAN nach (FIND)-GER umschaltbar gemacht. Das ursprüngliche (FIND) stimmt weitestge-

Forth ist für alle da

hend mit dem jetzigen (FIND)-ENG überein. In (FIND)-ENG mußte noch der Umstand berücksichtigt werden, daß die Namensfelder jetzt alle noch ein Zusatzbyte am Ende enthalten (FIND)-GER wurde so programmiert, daß es nicht mehr, wie noch das ursprüngliche (FIND), nach der 1 im höchsten Bit des letzten Zeichens im Namensfeld Ausschau hält, sondern beim Durchsuchen in Vorwärtsrichtung von der im ersten Byte des Namensfeldes gespeicherten Namenslänge Gebrauch macht und beim Suchen in Rückwärtsrichtung von der im Zusatzbyte gespeicherten Namenslänge.

NAME>

In TKERNEL.FTH wurde NAME> DEFERiert und per ENGLISH nach NAME>-ENG oder per GERMAN nach NAME>-GER umschaltbar gemacht. Das ursprüngliche NAME> stimmt weitestgehend mit dem jetzigen NAME>-ENG überein. Nur wurde aus

```
: NAME> 1 TRAVERSE 1+ EVEN ;
```

jetzt

```
: NAME>-ENG 1 TRAVERSE 2+ EVEN ;
```

um das Extrabyte zu berücksichtigen. NAME>-ENG kann nur für die per ENGLISH umgeschaltete ursprüngliche Form des Systems verwendet werden, die keine Umlaute in den Namen zuläßt. In NAME>-GER wurde TRAVERSE in ähnlicher Form vermieden wie bei (FIND)-GER .

>NAME

In TKERNEL.FTH wurde >NAME DEFERiert und per ENGLISH nach >NAME-ENG oder per GERMAN nach >NAME-GER umschaltbar gemacht. Das ursprüngliche >NAME stimmt mit dem jetzigen >NAME-ENG überein. Das Extrabyte gliedert sich in die ohnehin per ALIGN eingefügten Füllbytes ein und braucht nicht gesondert berücksichtigt zu werden. NAME>-ENG kann nur für die per ENGLISH umgeschaltete ursprüngliche Form des Systems verwendet werden, die keine Umlaute in den Namen zuläßt. In NAME>-GER wurde TRAVERSE in ähnlicher Form vermieden wie bei (FIND)-GER .

.ID

gibt den Namen des Wortes, das zu der Namensfeldadresse nfa gehört, auf dem Bildschirm aus. .ID wurde vektorisiert und kann per GERMAN nach .ID-GER und per ENGLISH nach .ID-ENG geschaltet werden. Weiteres siehe dort.

.ID-GER

Das ist das neue .ID, das über GERMAN (siehe dort) eingeschaltet wird. Es ist für Systeme gedacht, die die 1 im höchsten Bit der Zeichen im Namensfeld nicht testen. Es werden alle Zeichen so wiedergegeben, wie sie im Namensfeld stehen. Insbesondere erscheint also bei ASCII-Wert 142 ein Ä und nicht das Notenzeichen mit ASCII-Wert 14 d. Alles weitere siehe .ID-ENG und .ID.

.ID-ENG

gibt den Namen des Wortes, das zu der Namensfeldadresse nfa gehört, auf dem Bildschirm aus. Wird per ENGLISH (siehe dort) über ' .ID-ENG IS .ID mit .ID verkoppelt. Es wird auf 1 im höchsten Bit der einzelnen Zeichen getestet. Sobald ein Zeichen mit ASCII-Wert ab 128 auftritt, wird dasjenige Zeichen ausgegeben, das zu dem Anteil char 127 AND gehört. Die darauf folgenden Zeichen des Namensfeldes werden durch _ ersetzt. Vom letzten

Byte, das in einem System, das auf 1 im höchsten Bit testet, immer eine solche 1 trägt, wird grundsätzlich der Anteil byte 127 AND ausgegeben. .ID-ENG entspricht dem ursprünglichen .ID . Wenn sich das anzuzeigende Wort im Einschritt-Modus (per DEBUG einschaltbar) befindet, dann wird es hervorgehoben dargestellt.

GERMAN

schaltet die vektorisierten Worte NAME> >NAME "CREATE .ID nach NAME>-GER >NAME-GER "CREATE-GER .ID-GER, das Wort (FIND) erst zur "Zwischenstation" (FIND)-XXX und dann nach (FIND)-GER, und führt das Wort NAME-GER aus. Damit wird aus dem System ein System, in dem auch Zeichen mit ASCII-Werten über 127 in den Wortfeldern vorkommen dürfen. Alle TRAVERSE-ähnlichen Funktionen werden in der sich damit ergebenden Form des Systems umgangen.

ENGLISH

schaltet die vektorisierten Worte NAME> >NAME "CREATE .ID um nach NAME>-ENG >NAME-ENG "CREATE-ENG .ID-ENG, das Wort (FIND) erst zur "Zwischenstation" (FIND)-XXX und dann nach (FIND)-ENG, und führt das Wort NAME-ENG aus. Damit wird aus dem System ein System, in dem keine Zeichen mit ASCII-Werten über 127 in den Wortfeldern vorkommen dürfen. Das System arbeitet dann an verschiedenen (eigentlich nur wenigen) Stellen mit TRAVERSE . Das ist der sich auf ASCII-7 stützende amerikanische Standard, der von F83 und damit auch von Turbo-Forth her bekannt ist. (TRAVERSE testet auf 1 im höchsten Bit der Zeichen im Namensfeld.) Das im vorliegenden System F-TP eingeführte Extrabyte am Schluß der Namensfelder, das genau wie das erste Byte die Stringlänge enthält, wird allerdings auch nach dem Umschalten per ENGLISH beibehalten. Es besteht keine Veranlassung zur Vermutung, daß das irgendwo stören könnte. Es wird im vorliegenden System genauso behandelt wie die "Füllbytes", die überall per ALIGN erzeugt werden,

um die Transputer-Wortgrenzen (32 Bit) einzuhalten. Die eigentlichen Füllbytes enthalten das Leerzeichen (ASCII 32 d).

NAME-GER

Mit diesem Wort wird in sämtlichen Worten des gerade im Speicher befindlichen Systems (eigentliches System und eventuell per INCLUDE "angehängte" Anwendungen) das höchste Bit im letzten Zeichen im Namensfeld auf 0 gesetzt. Das betreffende System ist dann

Das Machbare sofort machen, das Nichtmachbare später.

"germanifiziert" (siehe auch GERMAN und ENGLISH). Bis auf CRESET (in Maschinensprache) ist NAME-GER eine Kopie von NAME-ENG. NAME-GER ist im "Umschaltewort" GERMAN enthalten. An der Funktionsweise von NAME-GER ist (FIND) maßgeblich beteiligt. Damit das bei einem "englischen" System eingeschaltete (FIND)-ENG keine Schwierigkeiten mehr und ein einzuschaltendes (FIND)-GER noch keine Schwierigkeiten macht, wird zunächst erst (FIND)-XXX eingeschaltet, dann NAME-GER und dann erst (FIND)-GER.

NAME-ENG

Mit diesem Wort wird in sämtlichen Worten des gerade im Speicher befindlichen Systems (eigentliches System und eventuell per INCLUDE "angehängte" Anwendungen) das höchste Bit im letzten Zeichen im Namensfeld auf 1 gesetzt. Das betreffende System ist dann "anglifiziert" (siehe auch GERMAN und ENGLISH). Bis auf CSET (in Maschinensprache) ist NAME-ENG eine Kopie von NAME-GER. Schwierigkeiten mit (FIND)-ENG beim Umschalten werden durch Zwischenschalten von (FIND)-XXX (siehe auch NAME-GER) umgangen.

NAME-CLEAR

Mit diesem Wort wird in sämtlichen Worten des gerade im Speicher befindli-

chen Systems (eigentliches System und eventuell per INCLUDE "angehängte" Anwendungen) das höchste Bit im ersten bis zum vorletzten Zeichen im Namensfeld auf 0 gesetzt. Namen, die Zeichen mit ASCII-Werten über 127 enthalten (wie sie beispielsweise in einem per GERMAN umgeschalteten System erzeugt sein könnten) stören dann in einem auf ENGLISH geschalteten System (zunächst bei WORDS usw.) nicht. Natürlich ist es schwierig, sie dort dann sinnvoll weiterzuverwenden, aber zur Fehlersuche halte ich es für sinnvoll, diese Möglichkeit bereitzustellen. (Motto: Alles Machbare wird sofort gemacht. Über das Nichtmachbare kann man dann in Ruhe nachden-

ken.)

(FIND)-XXX

Diese Version ist als Zwischenversion für das Umschalten von ENGLISH nach GERMAN gedacht. Sie testet bei den Vergleichen der einzelnen Zeichen der Namen nicht nach 1 im höchsten Bit, sondern verwendet als Abbruchkriterium die im allerersten Byte gespeicherte Länge des Namens. Sie wertet aber im letzten Byte das Zeichen nur als Zeichen von ASCII-Wert bis zu 127. Diese Version funktioniert also sowohl in einem System mit 1 im höchsten Bit des letzten Namensbytes als auch in einem System ohne. Vorausgesetzt, besagte Systeme verwenden als letztes Zeichen im Namensfeld keins mit einem ASCII-Wert über 127. Mit anderen Worten, die Zurückschaltung per ENGLISH kann auch in einem "germanifizierten" System, das schon Anwendungsworte mit Umlauten in den Namen enthält, vorgenommen werden. Nur müssen die Worte mit Umlauten dann "nachbehandelt" werden. Sie lassen sich im Regelfall nicht mehr aufrufen. Sicher ist es auf alle Fälle, das "germanifizierte" System erst per EMPTY vom Anwendungsteil zu befreien und diesen dann in "bereinigter" Form, also ohne Umlaute in den Namen, erneut (per INCLUDE) hinzuzuladen.

(;USES)

Die bisherige Definition war

```
:(;USES)R> @LAST @NAME>!;
```

Ich habe diese durch

```
:(;USES)R> @LAST @  
DUP C@ 31 AND + 2 + EVEN!;
```

ersetzt. Diese neue Version nützt für den Sprung von nfa zu cfa die im ersten Byte des Namensfeldes gespeicherte Länge des Namensstrings. Die in die jetzige Definition von (;USES) neu eingebauten Worte sind genau die Worte, die in NAME>-GER vorkommen. Eine solche Umgestaltung kann meiner Meinung nach nirgendwo schaden. Es kommt ja nur auf die Funktion an. Und die bleibt dieselbe. Das neue (;USES) kann nun aber auch bei Namen verwendet werden, die Zeichen mit ASCII-Wert über 127 (z.B. Umlaute) enthalten.

(;CODE)

Die bisherige Definition lautet

```
:(;CODE)R> LAST @NAME>!;
```

Ich habe diese in

```
:(;CODE)R> LAST @  
DUP C@ 31 AND + 2 + EVEN!;
```

umgewandelt. Für diese neue Version gilt dasselbe wie das, was bei (;USES) gesagt wurde.

RECURSE

Die bisherige Version lautet

```
:RECURSE LAST @NAME> , ;IMMEDIATE
```

Ich habe diese geändert zu

```
:RECURSE LAST @  
DUP C@ 31 AND + 2 + EVEN , ;
```

Für diese neue Version gilt dasselbe wie das, was bei (;USES) gesagt wurde.

Literatur

- [1] Behringer, F.: in JEDI (Zeitschrift der französischen Forth-Gruppe), Heft 45 (1988), S.8.
- [2] Gerdes, M.: Knöpfchen, Knöpfchen. Teil 1. c't (1988), Heft 6, 148-164.
- [3] Gerdes, M.: Knöpfchen, Knöpfchen. Teil 2. c't (1988), Heft 7, 178-199.
- [4] Gerdes, M.: Knöpfchen, Knöpfchen. Korrekturen. c't (1989), Heft 2, 12.
- [5] Hoffmann, U.: F-PC spricht Deutsch. Vierte Dimension 7 (1991), Heft 1, S.12.
- [6] Klingelberg, A.: KEYB8B. Vierte Dimension 9 (1993), Heft 1, S.17-20.
- [7] Vogt, C.: Internationaler Zeichensatz im F-PC. Vierte Dimension 8 (1992), Heft 3, S.7-12





Lehrbuch: Zech auch auf Englisch gut

About Zech's book: I learned Forth with this book (first German edition). I had Winfield's "The Complete Forth" first, but it really did not cover enough depth. Still, I do not recommend learning Forth with Zech's book alone, because it is somewhat unstructured in its presentation (it also does not cover the ANS Forth standard, AFAIK). I have the English edition in front of me right now. Its ISBN is 0-13-327040-8. It states it is a translation of "Die Programmiersprache FORTH" published in 1987 by FranzisVerlag GmbH. As for covering the ANS Forth standard, this is totally out of the question. The book was written in 1986 and the English translation finished in 1990. The ANS Forth standard did not exist at that time.

It early states that the book concentrates on fig-Forth because it is in the public domain and available for a wide range of processors. Many other Forth systems are mentioned, but they cannot be covered without making the book more confusing and unstructured. There is a conflict between the inventors of new Forth systems and standards on one hand and textbook writers on the other. A textbook writer must strive for clarity and understanding. Creators of new versions of Forth just scribble down whatever pops up in their heads if they even write any documentation at all. We had a textbook on the ANS Forth standard, "FORTH: The New Model" by Jack Woehr. It went out of print faster than any other Forth book I've ever seen.

If a book is going to sell and stay in print long enough to be worth the effort to write it, it has to be written very well indeed. A tutorial about computer programming should have a working programming system for students to use. At the time

Ronald Zech wrote his book, the Forth system most widely available was fig-Forth. It is still widely used. If anyone wanted to write a new Forth textbook today,

would he use an ANSI Forth version?

I think not. If a useful and popular version of Forth following the ANSI standard is ever created, it will be time for the committee.

**Forth for Professionals
by Ronald Zech
translated by Prof. Rex
Last published by Ellis Hor-
wood Ltd. 1990**

[Ergänzung von Anton Ertl: The second edition of the German version ("Die Programmiersprache >Forth") discusses F83. It appeared before 1990, so I assume that that >would be the basis of this translation.]

*mikc@gnu.ai.mit.edu (Mike
Coughlin) in comp/lang/forth,
Sep '95*

Presseschau: Byte A Brief History

Die Zeitschrift "Byte" feierte im Septemberheft ihr 20jähriges Bestehen mit einem umfassenden Rückblick. In "A Brief History of Programming Languages" wird Forth mehrmals angesprochen. Als typische Anwendung wurde aufgeführt, daß 1985 das U-Boot das die Titanic lokalisierte in Forth programmiert war.

*Rafael Deliano (jrd), Oktober
1995*

Bücher:

Starting Forth Polyglott

Laut dem Literaturverzeichnis des "MARC4-Programmer's Guide" ist "Starting FORTH" bereits in Deutsch, Französisch, Holländisch, Japanisch und Chinesisch erschienen. Letzteres hat wohl Dr. Ting verbochen.

*Rafael Deliano (jrd), Oktober
1995*

Forth-Gesellschaft: Brief aus der Provinz

Die FORTH - Gruppe Moers

Die Moerser FORTH-Gruppe geht auf eine Idee aus dem Anfang der 80er Jahre zurück. Damals hatten Teilnehmer eines VHS FORTH-Kurses die Idee, eine lokale Gruppe zu gründen. FORTH als Programmiersprache war noch weitgehend unbekannt, aber die ersten FORTHER gab es auch schon in Moers. Allerdings lebten und wirkten die ersten Moerser FORTHER unabhängig voneinander und auch untereinander unbekannt. Eine lokale Gruppe mit regelmäßigen Treffs und Diskussionen zu allem was den FORTHER bewegt, sollte zunächst die einzelnen Mitglieder aus ihren 'Elfenbeintürmen' herausholen und ihnen ermöglichen, ihre vielfältigen Erfahrungen untereinander nutzbar zu machen. Insbesondere die Anfänger und Quereinsteiger - meist quer von BASIC und/oder Turbo Pascal - erhofften sich in einer solchen Gruppe Rat und Hilfe.

Nach anfänglichen Schwierigkeiten bei der Suche nach geeigneten Räumen, 'konstituierte' sich die Moerser Gruppe 1987 in den Räumen des Moerser Arbeitslosenzentrums, wo sie heute noch regelmäßig an jedem Samstag zusammenkommt. Ganz bewußt entschloß man sich damals zu einem 'losen Verbund'. Die Gruppe ist kein eingetragener Verein und hat weder einen Vorsitzenden noch einen Präsidenten. Es gibt keine Mitgliederbeiträge und dementsprechend weder einen 'Kassenwart' noch eine Revision. Die Teilnahme an den wöchentlichen Treffs ist ebenso wenig Pflicht, wie es die Mitgliedschaft in der Deutschen Forthgesellschaft auch nicht ist - obwohl beides durchaus wünschenswert wäre.

Seit 1987 bietet die Moerser Forthgruppe Kurse für Anfänger und Einsteiger an, die semesterweise durchgeführt werden und von 'A bis Z' durch FORTH führen. Zusätzlich bietet die Gruppe Diskussionsforen zu 'forthigen' Themen

Forthgeschritteneran, sowie gemeinsame Experimente mit unterschiedlichster Hardware. Hilfe bei allen Problemen rund um den Computer sind ebenso selbstverständlich wie die kostengünstige, gemeinschaftliche Beschaffung von

Hard- und Software, oder wie gemeinsame Besuche z.B. der jährlich stattfindenden "Hobbytronic" in Dortmund.

Neben den 'elektronischen' Inhalten der 'Gruppenarbeit' kommt

der gesellige Teil nicht zu kurz. Exkursionen zum Rhein, wo sich forthige Themen auch im Biergarten diskutieren lassen, Radtouren mit den Familien und ein jährliches Sommerfest bringen die Moerser FORTHER immer wieder 'in die Realität zurück'. Gelegentlich lockert jemand etwas 'trockene' Themen mit selbstgebackenem Kuchen auf. Mit anderen Worten: Es ist (fast) immer etwas los, in Moers.

Um in der Moerser Forthgruppe mitzumachen, bedarf es weder einer Anmeldung noch irgendwelcher anderer Formalitäten. Einfach hingehen, Spaß am Computer, an FORTH und am Lernen mitbringen und den "Mitgliedern" sagen, was man sich von der Gruppe erhofft und wünscht - das ist schon alles. Auch werden von den Moerser FORTHERn keine 'Vorabbedingungen' an interessierte Computerfans gestellt. Weder Berufe, noch Ausbildungsstände, noch Weiterbildung in der EDV sind wichtig. Nur der Spaß am gemeinsamen Lernen zählt. Wen es aber trotzdem interessiert - in der Moerser Gruppe finden sich Berufe wie Bergleute, Drucker, Kaufleute, Schüler, Lehrer, Sozialarbeiter und Programmierer wieder, ebenso wie alle Ausbildungsstände. Hier stehen der Spaß am Computer und der Spaß an gemeinsamer Freizeitgestaltung im Vordergrund.

In der Moerser Forthgruppe wird zur Zeit ausschließlich mit PCs gearbeitet, was aber für Besitzer anderer Computer kein Hinderungsgrund sein sollte. Auf den PCs wird ZF eingesetzt, ein Public Domain System von Tom Zimmer, das auch professionellen Ansprüchen an ein vollständiges Entwicklungssystem genügt. Die Entscheidung für das ZF als 'das Moerser Forth' ist in der Gruppe bereits Ende der 80er Jahre gefallen. Nach einer Phase sehr gründlicher Tests mit unterschiedlichen Forth-Systemen - u.a. auch mit einem technisch sehr guten, aber leider relativ teurem, kommerziellen System - hat sich die Moerser Gruppe für das ZF entschieden, weil dieses Forth den gelungensten Kompromiß zwischen den Ansprüchen der 'Cracks' und den Forderungen der Einsteiger darstellt. ZF ist ein System das durchaus professionellen Ansprüchen genügt, aber dabei überschaubar (und durchschaubar) bleibt, und sich den Bemühungen um das Lernen niemals widersetzt. Das ZF ist von den Moerser Forthern in weiten Bereichen 'eingedeutscht' und an die

Rezensionen gesucht!

Das Forth-Magazin möchte gerne mehr Rezensionen veröffentlichen. Außerdem suchen wir noch Personen, die ihre Zeitschriften hinsichtlich forthrelevanter Themen lesen und darüber Kurzberichte verfassen!

Bitte bei der

Redaktion melden / Claus

Ingenieurbüro Dr. Maier-Schuler	Software-Entwicklung Prüfgerätebau Anlagensimulation
------------------------------------	--

Echtzeit-Simulation und Datenerfassung unter OS/2. Mit grafischer Oberfläche und Schnittstellen zu anderen Anwendungen.

cpFORTH

das FORTH-System für OS/2. Erweiterbar durch DLL- und REXX-Schnittstellen. Als C++ Objekt auch in User-Programme integrierbar.

Shareware- und Vollversionen verfügbar!

Dr. Maier-Schuler Wissenstraße 19 88690 Uhldingen-Mühlhofen Tel.: 07556-5618

Bedürfnisse deutschsprachiger Anwender angepaßt, sowie um einige Funktionen erweitert worden, die auch anderen Systemen 'gut zu Gesicht' stünden. Im Verlaufe der vielen Jahre, die in Moers bereits mit ZF gearbeitet wurde, sind dort eine Reihe von 'Lehr- und Kurstexten' zu Forth im Allgemeinen und zum ZF im Besonderen entstanden, die den Mitgliedern der Moerser Gruppe jeweils von Anfang an in gedruckter Form zur Verfügung standen. Im Verlaufe der vergangenen Jahre haben mehr und mehr FORTHer aus der Bundesrepublik

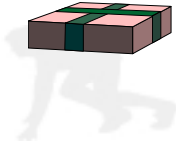
sowohl das ZF, als auch diese Texte von den Moerser Forthern angefordert und erhalten, wobei die inhaltlich hervorragenden und äußerlich fast professionell gestalteten 'Dokumente' (z.B. 100 seitige Dokumentation zu ZF's Assembler, WinWord Dokument mit Angaben zu CPU-Takten und Beispielen zu jedem Mnemonic) aus Zeit- und Kostengründen nur noch auf Disketten mitgeliefert werden. Selbstverständlich verschickt die Moerser Gruppe 'ihr' Zf kostenfrei, in der Hoffnung, daß FORTH auf diese Weise Verbreitung und Freunde findet.

FORTH ist für die meisten FORTHer mehr als "nur" eine relativ "mächtige" Computersprache. Das gilt auch für die Mitglieder der Moerser Forthgruppe. Hier wird FORTH sowohl im professionellen und semiprofessionellen Bereich genutzt, als auch einfach nur als Werkzeug mit dessen Hilfe sich das Gesamtsystem aus Maschine - Betriebssystem und Programmiersprache durchschauen und "erfahren" läßt. Überhaupt wird das LERNEN in Moers groß geschrieben. Kein Moerser Forther käme auf die Idee, zur Entwicklung z.B. einer Datenbank-Applikation 'sein' Forth herzunehmen. Zur Lösung solcher Aufgaben existieren geeignetere Tools. Aber wenn es darum geht, sich in dieser Applikation Strukturen deutlich zu machen und zu verstehen, "was da überhaupt passiert", ist und bleibt Forth die erste Wahl unter allen möglichen 'Werkzeugen'.

FORTH ist in gewisser Weise auch eine eigene Philosophie. "Keep it simple" ist die Botschaft von Mr. Moore, der dieses Grundprinzip wie folgt beschreibt: "Eine einfache Lösung ist elegant. Die elegante Lösung ist das Ergebnis einer gründlichen Anstrengung das wirkliche Problem zu verstehen und überzeugt durch das beeindruckende Gefühl der Richtigkeit. Ich betone diesen

Punkt besonders, weil er im Gegensatz zur vorherrschenden Meinung steht, die Leistungsfähigkeit steige mit der Kompliziertheit an. Einfachheit erzeugt Vertrauen. Zuverlässigkeit, Überschaubarkeit und Schnelligkeit"

F.PRINZ@MHB.gun.de
(Friedrich Prinz), oct '95



Nur noch bis 15. Dezember.

kann man das Forth-Weihnachtspäckchen bestellen. Es enthält die Broschüre zum Moerser Spiel 5x5

Die CD 'forthige Weihnacht 95' enthält das Zimmerforth und die Listings, um alles nachzuvollziehen, zahlreiche andere Forthsysteme, den kompletten Ftp-Server *Taygeta* aus USA mit bald 100 Mb; den kompletten Ftp-Server *Asterix*; genug um noch sehr, sehr lange damit zu spielen

Sofort bestellen! Info auf Seite 22

Dienstleistungen und Produkte von Forthlern und/oder für Forthler (Anzeige)

Ingenieurbüro Dipl.-Ing. Wolfgang Allinger

Tel. (+Fax.) 0+212-66811
Brander Weg 6
D-42699 Solingen

Entwicklung von µC, HW+SW, Embedded Controller, Echtzeitsysteme 1 bis 60 Computer, Forth+Assembler PC / 8031 / 80C166 / RTX2000 / Z80 ... für extreme Einsatzbedingungen in Walzwerken, KKW, Medizin, Verkehr / >20 Jahre Erfahrung.

Tel.: 0+381 -405 94 71 (Fax: -4059.471)
Joachim-Jungius-Str. 9
D-18059 Rostock

PC-basierende Forth-Entwicklungswerkzeuge, System comFORTH für DOS und Windows, Cross- und DownCompiler für diverse Microcontroller, Controllerboards mit 80C196, 80C537 und H8, Softwareentwicklung für Microcontroller und PC's, auch unter Windows (und fremdsprachig)

Tel.: 0+8233-30 524 (Fax: --9971)
Postfach 11 73
D-86406 Mering

FORTH-Software (volksFORTH, KKFORTH und viele PD-Versionen). FORTH-Hardware (z.B. Super8) und -Literaturservice. Professionelle Entwicklung für Steuerungs- und Meßtechnik.

ETA Elektrotechnische Apparate GmbH

Tel.: 0+9187 -10.0 (Fax: -10.397)
Industriestr. 2-8
D-90518 Altdorf (b. Nürnberg)

Produkte für Echtzeitanwendungen
FRP1600: Echtzeitprocessor optimiert für Forth
RP-PB1: FRP1600 Prototyping Board.

Tel.: 0+2404 -61648 (Fax: -63039)
Strassburgerstr. 12
D-52477 Alsdorf (b. Aachen)

Computergestützte Meßtechnik und Qualitätskontrolle, Fuzzy, Datalogger, Elektroakustik (HiFi), MusiCassette High-SpeedDuplicating, Tonband, (engl.) Dokumentationen u. Bed.-anl.

FORTech Software GmbH

Ing.Büro Klaus Kohl

Möchten auch Sie oder Ihre Firma hier aufgeführt werden? Bitte wenden Sie sich an die Anzeigenverwaltung (s. Impressum).

Ihre Anzeige plus 3 Zeilen je 45 Zeichen Text kosten 90.-DM (incl. 20.-DM Einrichtung/Änderung, je Zusatzzeile 10.-DM) und das komplett für ein ganzes Jahr.

Dienstleistungen und Produkte von Forthlern und/oder für Forthler (Anzeige)

Forth-Gruppen regional

Berlin	Claus Vogt Tel.: +30 -782 81 79 p Treffen: nach Absprache
Rhein-Ruhr	Jörg Plewe Tel.: +208 -49 70 68 p Treffen: jeden 1. Samstag im Monat im S-Bahnhof Derendorf Münstererstr. 199,
Moers	Friederich Prinz Tel.: +2841 -5 83 98 p Treffen: jeden Samstag 14:00 Arbeitslosenzentrum, Donaustr. 1, Moers
Darmstadt	Andreas Soeder Tel.: +6257 -27 44
Mannheim	Thomas Prinz Tel.: +6271 -28 30 p Ewald Rieger Tel.: +6239 -86 32 p Treffen: jeden 1. Mittwoch im Monat, Vereinslokal Segelverein Mannheim e. V., Flugplatz Mannheim-Neuostheim

µP-Controller Verleih

Thomas Prinz
Tel.: +6271 -28 30 p

Gruppen Gründungen, Kontakte

Regional
Stuttgart
Wolf-Helge Neumann
Tel.: +711 -8 87 26 38 p

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

Forth-Hilfe für Ratsuchende

Forth allgemein
Jörg Plewe
Tel.: +208 -49 70 68 p
plewe@mpi-dortmund.mpg.de

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

Jörg Staben
Tel.: +2103 -24 06 09 p

Spezielle Fachgebiete

Arbeitsgruppe Marc4
Rafael Deliano
Tel./Fax.: +89 -841 83 17 pf

**Anfänger und
Wiedereinsteiger**
Gerd Limbach
Tel.: +2051 -25 51 12 p
Mo.+Di. 20:00 - 22:00

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

FORTHchips (FRP1600, RTX, Novix...)
Klaus Schleisiek-Kern
Tel.: +40 -330 674 p

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

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

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

KI (Künstliche Intelligenz), OOF (Object Oriented Forth)
Ulrich Hoffmann
Tel.: +4351 - 712 217 pf

Birgit Steffenhagen
Tel.: +38204 - 129 33 pa
+381 - 498 35 52 g

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

Forth-Vertrieb
**volksFORTH/ultraFORTH,
RTX/FG/Super8/KK-FORTH**
Ingenieurbüro Klaus Kohl
Tel.: +8233 -3 05 24 p
Fax: +8233 -99 71 f

Forth-Mailbox (KBBS)
+431-533 98 98 :8N1
Sysop: Holger Petersen
Fax: +431-533 98 97 f
Tel: +431-533 98 98 p bis 22 Uhr
Mail: hp@kbbs.org

Hinweise

Zu den Telefonnummern

f == FAX
a == Anrufbeantworter, hier können Sie Ihren Ansprechpartner eventuell vorinformieren,
erwarten Sie bitte keinen (kostspieligen) Rückruf
g == geschäftlich, zu erreichen innerhalb typischer Arbeitszeiten
p == privat, zu erreichen außerhalb typischer Arbeitszeiten

Die Adressen des Forth e. V. (Forth Büro) und der Redaktion/ finden Sie im Impressum

FORTECH Software

- Forth-Entwicklungsumgebung comFORTH unter DOS oder Windows
- interaktive Crossentwicklungssysteme für Mikroprozessoren von Intel, Motorola, Zilog, TI ...
- Softwareentwicklung für PC und Mikrocontroller
- System- und Anwendungsprogrammierung unter Windows

comFORTH

- Forth-Entwicklungsumgebung für Windows
- interaktive Benutzbarkeit aller Windows-API-Funktionen und -Strukturen
- kombinierbar mit anderen Programmiersprachen
- Unterstützung von DDE, DLL, VBX, ...

fieldFORTH

- Forth-Entwicklungssystem für eingebettete Systeme
- interaktive Programmierung off-line und on-line
- verfügbar für diverse 8-, 16- und 32-Bit Mikrocontroller und -Prozessoren (TMS320C40, M68332, M68HC11,...)
- **NEU!!!** Evaluation-Kit M68HC11 inclusive Board MINI-HC11 296,70 incl. MwSt.

FORTECH Software GmbH

J.-Jungius-Str. 9 • D-18059 Rostock • Tel: (03 81) 4 05 94 72 • Fax: (03 81) 4 05 94 71