

# NAXOS

The screenshot shows the NAXOS compiler IDE interface. The main window displays a source file named 'D:\TEMP\X\DIVERS\CRTDEMO.FTH'. The code in the editor includes:

```

import< stdio clock xcrt >

< *** textmodus *** >
const bw40 0 ;
const co40 1 ;
const bw80 2 ;
const co80 3 ;
const mono 7 ;
const font8x8 256 ;
const c40 co40 ;
const c80 co80 ;

< *** Farben *** >

OBJECT Crt
var checkbreak byte ; < boolean >
var checkeof byte ; < boolean >
var directvideo byte ; < boolean >
var checksnow byte ; < boolean >
  
```

An 'Information' dialog box is overlaid on the code, displaying the following text:

```

Information
NAXOS-COMPILER U 2.12
(C) Peper, Zissis, Tossounidis
14.04.1993
OK
  
```

The IDE's status bar at the bottom shows function key shortcuts: F2 Speichern, F3 Laden, Alt-F3 Schließen, F5 Zoom, F6 Next, F10 Menü.

NAXOS ist ein FORTH-ähnliches System, das in der Zeitschrift „DOS TOOLBOX“ 3/92 vorgestellt wurde. Es besteht aus einem Kommandozeilen-Compiler und einer an Turbo-Pascal 6 angelehnten IDE. NAXOS wurde in Turbo Pascal 6.0 geschrieben, die IDE wurde mit Turbo Vision 1.0 implementiert.

NAXOS basiert in seinen Grundlagen auf dem Software-Projekt **SForth** von DOS International (C) 1987 Volker Everts und DOS International sowie den Vorgängerprojekten der Autoren, FCC u. MCC (C) 1989 bis 1992 K.Peper, I.Tossounidis & A.Zissis.

## SFORTH

SForth wurde in der DOS INTERNATIONAL 9/1987 in Version 2.0 vorgestellt (© 1987 Volker Everts und DOS International). SForth ist ein in Turbo Pascal 3.0 geschriebener Forth-Compiler für den 8086 und erzeugt COM-Dateien.

In der DOX EXTRA 2 (1988) wurde SForth letztmalig in Versionsstand 3.1 komplett beschrieben und abgedruckt. Federführend für die Neuerungen in SForth 3.x war Erwin Schomburg.

Zu SForth gehört neben dem Compiler auch das 'Vocabulary Forth' und das 'Vocabulary System'.

## Historie

- 1986 erste Ansätze zur Programmierung auf Schneider CPC 464 unter CP/M 80.
- Turbo Forth Versionen 0.x (April/Mai 1987)  
Programmierung erster Versuchssysteme auf einem IBM-kompatiblen PC (Versionen 0.x).  
Konzeption der grundsätzlichen Arbeitsweise des Compilers.
- Turbo Forth Version 1.0 (Juni 1987)
- Turbo Forth Version 2.0 (Juli 1987)

- Turbo Forth Version 2.1 (Juli 1987)  
In der DOS 9/87 wird die Version 2.1 des Turbo-Forth-Compilers abgedruckt. Gleichzeitig erfolgt eine Namensänderung in „SForth“, um Namenskonflikte mit Borland-Software zu vermeiden.
- SForth Version 3.0 (Dezember 1987)  
Der Compiler steht in einer völlig neu überarbeiteten Version zur Verfügung (-> Erwin Schomburg). Wichtigstes Merkmal des neuen Compilers ist die Fähigkeit, auch speicherresidente Utilities damit erzeugen zu können.

## Erinnerungen

### Erwin Schomburg

*.. ein kurzer Abriss aus meiner Sicht:*

*Das erste DOS Heft schlug ein wie eine Bombe, in dieser „goldenen“ Ausgabe fand man zum ersten Mal sonst nicht dokumentierte oder nur mühsam zu beschaffende Interna zum MS-DOS zum Nachschlagen in einer Quelle. In einer der nächsten Ausgaben stellte der damalige Chefredakteur Volker Everts den Forth-Compiler vor. Ein in Turbo-Pascal geschriebenes Programm konnte aus Quelltexten in Forth-Syntax unter MS-DOS lauffähige Programme erzeugen. Angeblich soll die Urversion auf der Arbeit einer französischen Gruppe basieren, ich habe dies aber nicht weiter verfolgt. Jedenfalls war die in der DOS veröffentlichte Version mit der heißen Nadel gestrickt und strotzte vor Fehlern. Da ich damals recht gut mit Turbo-Pascal vertraut war und mich schon für Forth interessierte, störte mich der unbrauchbare Zustand des Programms so sehr, dass ich mich dazu in Form von Leserbriefen äusserte. Daraufhin forderte mich Herr Everts auf, es doch besser zu machen, so ich könnte. Dies führte zu der im Sonderheft 3 der DOS veröffentlichten weitgehend stabilen Version. Da die DOS damals nur für Artikel, nicht für Listings, zahlte, wurde der Artikel zum Interrupthandling zum Alibi für die Kompensation meiner Mühen am Programmkern.*

*Sehr viel Resonanz von Lesenseite hat es sonst nicht zu diesem Projekt gegeben. Ich selbst habe die Möglichkeit, mit SForth TSR (Terminate and Stay Resident) Programme für MS-DOS erstellen zu können, mehrere Male kommerziell genutzt. Unter dem Gesichtspunkt des stark begrenzten Speicherplatzes unter MS-DOS konnten mit keiner damaligen (sonstigen) Compilersprachen hinreichend kompakte Spezialtreiber erzeugt werden, Entwicklungs- und Testzyklen unter Assembler hätten für die fraglichen Projekte viel zu lange gedauert.*

## Download

- SFORTH (V 2.1, 2.3, 3.0, 3.1)[sforth.zip](#)

## NAXOS

**Alexandros Zissis** schreibt 2008:

*Prof. Dr. rer. nat. Peper war Professor für Physiologie in Homburg, von Hause aus allerdings Physiker und kein Mediziner, von ihm kam der Anstoß für das Projekt. Das Problem war, daß zur Prozesssteuerung und Meßwerterfassung physiologischer Experimente dort eine Nicolet Med80 (PDP7 kompatibler 19bit-Rechner mit diskreter CPU und faszinierendem Ferritkernspeicher und zwei sehr schnellen Hardwaremultiplizierern) außer Betrieb genommen werden sollte, die Datenerfassung in*

Echtzeit auf PC's (derzeit 286'er und ein '386DX33) nicht so wirklich klappen wollte. Also wurde eine Arbeitsgruppe gebildet und zwei Studenten als wissenschaftliche Mitarbeiter eingestellt. Herr Tossounidis war hauptsächlich für die Benutzeroberfläche verantwortlich, ich im Wesentlichen für den Compiler und mit Prof. Peper zusammen für die Laufzeitbibliotheken. Mit der Zeit kamen dann mehr oder weniger aus Spaß an der Freude weitere Features wie Objekte, Vererbung (allerdings nur Einfachvererbung), dynamische Speicherverwaltung und eben der Zugriff auf 4GB linearem Speicher in Version 3.0 hinzu. Ein ganz nettes Beispiel für die Objekte in Naxos ist WINDOWS.FTH (in Naxos 3 wurde die Endung für Quelldateien in .NX geändert .FTH war sozusagen noch ein Souvenir von SFORTH). Ansonsten kann man durchaus zugeben, daß wir versuchten, möglichst alle Möglichkeiten, die uns in Turbo Pascal 5.5 gefielen, in Naxos zu übertragen. NAXOS selbst war von uns als Denkanstoß gedacht und public domain, lediglich der Copyright Text sollte nicht verändert werden.

Die DOS-Toolbox-Version produzierte - soweit ich weiß - oft keinen lauffähigen Code, da die Redaktion entgegen der Absprache damals den Copyright-Text veränderte und somit einige Systemvariablen verändert hat.

Zwei nicht ganz unwichtige Dinge noch: \* Nach dem Entpacken muß die Umgebungsvariable NAXOS auf das Verzeichnis mit den Naxos-Dateien zeigen, also z.B. SET NAXOS=C:\NAXOS, sonst startet die IDE nicht. \* Falls das Kompilieren ohne aussagekräftige Fehlermeldung direkt abbricht, sind die Dateiattribute vmtl. nicht korrekt. → Einfach im Naxosverzeichnis ein ATTRIB -r -s - h \*.\* ausführen. (Das Problem ergibt sich daraus, daß die Dateien unter Unix archiviert wurden und die DOS-Attribute dabei zuweilen unter die Räder kommen)

Ich wünsche viel Spaß mit NAXOS

## Downloads

- NAXOS Version 1.01 [naxos-dtx9203.zip](#)
- NAXOS Version 2.14 (ohne PROF) [naxos-2.14.zip](#)

BIN\ sind der Compiler, die IDE, usw.

LIB\ sind die Laufzeitbibliotheken

SRC\ die (Pascal-)Quelltexte

divers\ wie der Name sagt: diverse NAXOS-Programmbeispiele

- NAXOS Version 3.02 [naxos-3.02.zip](#)

Version 3.0 ist mit 32-bit-Support (aber nur unter Nativem DOS OHNE EMM386.EXE oder sonstigen Programmen, welche im protected mode arbeiten, lediglich der XMS Treiber muß geladen sein.)

- aktuelle Arbeitsversion

die Snapshots der Arbeitsverzeichnisse der Naxos Entwicklung, darin befinden sich auch diverse Dateien, welche in den „offiziellen“ Archiven fehlten, wie z.B. die Sprites (\*.spr) für das ping-pong Spiel oder im nx32 Zweig der Quelltext für den NXX039 Naxos-Compiler in Naxos.

From:

<https://hc-ddr.hucki.net/wiki/> - **Homecomputer DDR**

Permanent link:

<https://hc-ddr.hucki.net/wiki/doku.php/forth/naxos?rev=1372605583>

Last update: **2013/06/30 15:19**

