

Turbo-Pascal auf dem C 128

Turbo-Pascal hat sich seit seinem ersten Erscheinen vor einigen Jahren zu einer Art Neben-Standard für Pascal-Compiler entwickelt. Der Turbo-Compiler gilt als einer der schnellsten und leistungsfähigsten Pascal-Compiler überhaupt.

Unter CP/M ist Turbo-Pascal nun auch für den C 128 verfügbar.

Wer einen C 128 besitzt, der wird sich sicherlich auch schon mit dem CP/M-Modus seines Rechners beschäftigt haben. Denn mit CP/M eröffnet sich eine der größten Softwaresammlungen überhaupt. So wird auch das vielgerühmte Turbo-Pascal endlich für den Commodore-Anwender verfügbar.

Was bietet die moderne und überall verbreitete Sprache Pascal dem C 128-Anwender überhaupt? Hier ein kurzer Einblick:

Mit Pascal lernt man eine sehr strukturierte Sprache, bei der man den mitunter recht chaotischen Programmierstil von Basic sehr schnell vergißt. Diese nach dem französischen Mathematiker Blaise Pascal benannte Sprache fordert eine genaue Erfassung des Problems, das programmiert werden soll. Mit einfachem »Drauflosprogrammieren« wird man daher nicht sehr weit kommen.

Was bringt Turbo-Pascal?

Turbo-Pascal ist stark dem Standard-Pascal von Nikolaus Wirth, dem Schöpfer von Pascal, angepaßt, hat aber noch einige Zusätze und Erweiterungen parat, die das Herz eines jeden Pascal-Fans höher schlagen lassen: Mit ABSOLUTE kann der Programmierer die normalerweise vom Compiler vergebene absolute Speicheradresse von Variablen festlegen. EXTERNAL dient zum Aufruf von in Assembler geschriebenen und getrennt übersetzten Prozeduren und Funktionen. Mit INLINE kann Maschinensprache direkt in ein Pascal-Programm integriert werden, was bei zeitkritischen Anwendungen von Bedeutung sein kann. Die sehr maschinen-nahen Operationen SHL und SHR entsprechen den Schiebeoperationen in Assembler.

Das Attribut PACKED wird vom Turbo-Pascal-Compiler zwar ohne Murren akzeptiert, hat aber keine Wirkung, da Turbo-Pascal ohnehin schon einen sehr komprimierten Code erzeugt.

Doch damit ist das »Sonderzubehör« noch nicht erschöpft. Turbo-Pascal stellt noch eine Reihe von Standard-Prozeduren, -Funktionen und -Konstanten bereit, die die Programmierung erleichtern und die Möglichkeiten der Problemlösung erweitern.

Der wichtige und oft gebrauchte Datentyp STRING, den man sich in anderen Pascal-Versionen erst selbst definieren muß, ist in Turbo-Pascal bereits standardmäßig vorhanden. So können Strings ganz zwanglos deklariert werden:
 VAR EINGABE: STRING[80];
 NAME: STRING[16];

Derart verschieden dimensionierte Strings sind untereinander dennoch kompatibel, das heißt sie können untereinander zugewiesen werden, wobei natürlich der eventuell längere Teil des zugewiesenen Strings abgeschnitten wird.

Wenn ein größeres Pascal-Projekt nicht ganz in den begrenzten Arbeitsspeicher des Computers paßt, dann mag das für andere Programmiersprachen ein ernstes Problem darstellen, nicht so jedoch für Turbo-Pascal. Durch das Attribut OVERLAY können Pascal-Prozeduren oder -Funktionen als Overlay-Files kompiliert werden. Im Speicher wird dann lediglich Platz für die längste aller Overlay-Prozeduren reserviert. Beim Aufruf einer solchen Prozedur oder Funktion wird das entsprechende Overlay-File nachgeladen und ausgeführt. Insbesondere bei Prozeduren, die nur wenige Male im Verlauf des Programms aufgerufen werden, ist diese Overlay-Technik sehr effektiv.

Overlay-Technik löst Speicherplatzprobleme

Eine bemerkenswerte Anzahl von Prozeduren ist zur direkten Manipulation des Speichers vorgesehen. Es lassen sich zum Beispiel ohne Schwierigkeiten Maschinenprogramme oder andere Programmdateien vom Pascalprogramm aus laden und starten. Mit einigen Proze-

duren können sogar Routinen aus dem BIOS (Basic Input/Output System) und dem BDOS (Basic Disc Operating System) von CP/M angesprungen und somit das CP/M-Betriebssystem voll ausgenutzt werden.

Mit den gleichnamigen Prozeduren BIOS, BIOSHL, BDOS und BDOSHL lassen sich gezielt Routinen des CP/M-Systems aufrufen und für Pascal-Programme nutzbar machen; für CP/M-Kenner geradezu ein Leckerbissen.

Turbo-Pascal und der C 128

Da Turbo-Pascal unter CP/M läuft, ist es notwendig, zunächst CP/M von der Systemdiskette in Ihren C 128 zu laden. Wie Sie vielleicht bereits bemerkt haben, tut dies Ihr C128 von selbst, wenn Sie ihn anschalten und die Systemdiskette im Diskettenlaufwerk steckt. Er »bootet« (lädt und startet) CP/M automatisch. Das ist sowohl mit der 1570/1571-Floppy als auch mit der guten alten 1541 möglich. Von der Verwendung des 1541-Laufwerks kann allerdings nur abgeraten werden. Allein das Hochfahren des CP/M-Systems benötigt damit fast zwei volle Minuten. Meldet sich nun endlich CP/M auf dem Bildschirm, darf die Diskette mit Turbo-Pascal in das Floppy-Laufwerk geschoben werden. Wenn Sie noch keine Kopie der Original-Diskette gemacht haben, dann sollten Sie das spätestens an dieser Stelle tun. Wie alle CP/M-Software wird auch Turbo-Pascal ohne unnütze Kopierschutz-Pfuschereien geliefert. Dafür hat jede einzelne Diskette eine Seriennummer. Wenn man sich nach dem Kauf mit einer beigelegten Postkarte unter Angabe der Seriennummer als Benutzer registrieren läßt, hat man die beruhigende Gewißheit, bei Problemen mit dem Produkt nicht allein gelassen zu werden. Bei anderen bekannten Software-Produkten erschöpft sich die »Unterstützung« des Benutzers ja oftmals mit dem Hinweis »Telefonische An-

fragen können leider nicht beantwortet werden».

Nachdem man also jetzt die Originaldiskette sicherheitshalber kopiert hat, kann's endgültig losgehen. Mit DIR listen Sie das Inhaltsverzeichnis der Diskette und können mehrere Files darauf finden.

TURBOCOM ist der eigentliche Compiler, der die gesamte Arbeit leistet. Er benötigt zirka 32 KByte Speicher. TURBOOV ist die Overlay-Datei zum Turbo-System. TURBOMSG enthält alle Compiler-Meldungen im Klartext und kann auf Wunsch von Turbo-Pascal geladen werden.

TLISTCOM kann, wenn ein Drucker vorhanden ist, den Quelltext Ihrer Pascal-Programme auf dem Drucker ausgeben. Ein weiteres File mit dem Namen README gibt nach dem Auflisten mittels TYPE-Kommando Informationen über die Implementation von Turbo-Pascal.

Daneben gibt es noch einige Beispielsprogramme, mit denen man herumexperimentieren kann. Dazu gehört auch ein hervorragendes Tabellenkalkulationsprogramm namens MC, das vollkommen in Turbo-Pascal geschrieben ist.

Es dauert nun nicht mehr lange, bis Turbo-Pascal endgültig sein Hauptmenü zeigt und dem Programmieren nichts mehr im Wege steht.

Professioneller Texteditor

Wem der Name »WordStar« ein Begriff ist und wer eventuell schon damit gearbeitet hat, der darf sich freuen. Die Kommandos des Pascal-Editors sind mit den entsprechenden WordStar-Befehlen identisch und haben, wenn nicht anders vereinbart, auch die gleiche Tastenbelegung. Wer das WordStar-Textverarbeitungssystem noch nicht kennt, wird sich dann wohl erst einmal mit dem recht dicken Turbo-Pascal Handbuch, Stichwort Editor, vertraut machen müssen. Doch obwohl die Fülle der Kommandos auf den ersten Blick überwältigend ist (selbst Block-, Such- und Ersetz-Kommandos sind vorhanden), gestaltet sich das Arbeiten mit dem Turbo-Pascal-Editor doch sehr einfach und angenehm.

Der Cursor wird neben den WordStar-üblichen Control-Kommandos auch mit den vier Cursortasten bewegt, die sich gleich links neben den Funktionstasten des C128 befinden. Auch die INST/DEL-Taste hat ihre Funktion behalten, so daß man also auch ohne nähere Kennt-

nisse der einzelnen Editor-Kommandos gleich mit dem Programmieren beginnen kann.

Interessant dürfte sein, daß beim Druck der Return-Taste der Cursor nicht an den ganz linken Rand der neuen Zeile springt. Statt dessen plaziert er sich immer unter dem ersten Wort der zuletzt eingegebenen Zeile. Das Programmieren, mit den für Pascal so wichtigen Einrückungen von Zeilen, wird durch diese kleine Feinheit zum Kinderspiel.

Der Turbo-Compiler besitzt mehrere Funktionen zum Compilieren eines Programms. Mit einem Kommando des Hauptmenüs läßt sich ein Untermenü mit den möglichen Compiler-Optionen aufrufen, die das Übersetzen von Quelltexten auf verschiedene Art ermöglichen. Je nachdem, welche Option man wählt, wird der Compiler bei Aufruf entsprechend arbeiten.

Der Turbo-Compiler

Turbo-Pascal erlaubt es auf Wunsch, das im Moment im Speicher befindliche Quellprogramm zu compilieren und auch dort wieder abzulegen. Dieses wird dann vom Hauptmenü aus gestartet. So kann ein sofortiger Test des eben geschriebenen Programms erfolgen.

Der Compiler kann auch ein Quellfile von Diskette laden und compilieren, um das Compilat (das übersetzte Programm) dann als .COM-File wieder auf Diskette zu speichern. Dieses File ist später als Programmfile von CP/M aus, ohne Hilfe von Turbo-Pascal, lade- und startfähig, somit also ein vollkommen eigenständiges Programm.

Schließlich kann der Compiler auch sogenannte Chain-Dateien (Kennung .CHN) verarbeiten. Diese Übersetzungen haben keine eigenen Pascal-Bibliotheksroutinen und sind somit auch nicht einzeln für sich lauffähig. Sie sind dafür gedacht, von anderen .COM-Pascal-Programmen nachgeladen zu werden, um dort zu laufen.

Der letzte Untermenüpunkt mit dem Namen »Find run-time error« ist ein gutes Hilfsmittel zum Aufspüren von Laufzeitfehlern. Laufzeitfehler sind Fehler, die der Compiler nicht erkennen kann, da sie erst beim Ablauf des compilierten Programms auftreten. In diesem Falle unterbricht das Programm und gibt eine entsprechende Fehlermeldung und die Stelle aus, an der der Fehler aufgetreten ist. Nun besteht ein Compilat nicht mehr aus den Pascalbefehlen, die man im Quelltext eingetippt

hat, sondern aus Maschinencode. Die angezeigte Stelle eines Fehlers ist deshalb eine Hexadresse. Mit dieser Adresse kann man den Lauffehler im Quellprogramm jedoch bequem wiederfinden. Dazu muß das Quellprogramm noch einmal von Turbo-Pascal geladen werden. Ein Aufruf der oben genannten Find-Option und die Angabe der besagten Hexadresse genügen, und der Editor führt Sie automatisch an die Fehlerquelle. Eine praktische Sache.

Doch bevor ein Laufzeitfehler auftreten kann (man will es nicht hoffen), muß man sein Quellprogramm selbstverständlich erst compilieren lassen.

Mit der Wahl »C« im Hauptmenü startet der Compiler und compiliert je nach eingestellter Compiler-Option aus eben besprochenem Untermenü. Die Fehlerbehandlung während der Übersetzung geschieht bei Turbo-Pascal auf sehr interessante und effektive Weise. Hat der Compiler einen Fehler gefunden, den er nicht akzeptieren kann, läßt er eine entsprechende Fehlermeldung auf dem Bildschirm erscheinen. Falls man die Fehlermeldungen beim Start von Turbo-Pascal nicht geladen hat, wird nur die Fehlernummer ausgegeben, deren Bedeutung im Handbuch nachzulesen ist. Der Übersetzungsvorgang wird gestoppt und ein Druck auf die Escape-Taste (ganz links oben auf der Tastatur) wird erwartet.

Folgt man dieser Aufforderung, startet der Editor automatisch und postiert den Cursor ähnlich wie bei der Laufzeitfehlerbehandlung an die fehlerhafte Programmstelle. Eine überaus effektive Fehlersuche ist damit gewährleistet, die man bald nicht mehr missen möchte.

Schnell, schneller, turbo ...

Eine Auflistung des Programmes während der Compilation ist nicht möglich. Sie würde die Übersetzung nur unnötig verlangsamen.

Denn der Name »TURBO« hält, was er beim Lesen verspricht. Turbo-Pascal erledigt seine Aufgabe in Windeseile und erzeugt ein Compilat mit ebensolchen Eigenschaften.

Wenn ein C 64-Besitzer diese Geschwindigkeiten bestaunt, sollte er aber selbstverständlich daran denken, daß die Taktfrequenz des Z80-Prozessors im C 128 doppelt so hoch wie die des 6510 ist und der Compu-

ter somit auch um einiges schneller arbeiten kann. Leider ist sie aber (effektiv) wiederum nur wenig mehr als halb so hoch wie bei anderen Z80-Computern, was bedeutet, daß Turbo-Pascal auf anderen CP/M-Systemen noch schneller ist.

Trotzdem muß man sagen, daß Turbo-Pascal der schnellste derzeit für den C 128 verfügbare Pascal-Compiler ist.

Ein Benchmark-Programm, in dem eine Integer-Variable von 0 bis 2000 hochgezählt wurde, ergab bei der Ausführung einen Zeitbedarf von nur 0,4 Sekunden (zum Vergleich: Profi-Pascal 1,0 Sekunden; Oxford-Pascal 2,3 Sekunden).

Betrachtet man diese Zeiten, so wird es klar, daß Turbo-Pascal eindeutig der schnellste Compiler ist. Die Arbeitsgeschwindigkeit von Turbo-Pascal zeigt, was ein Z80-Prozessor alles vermag. Ein kleiner Anstoß vielleicht für manchen Interessierten, den Z80 näher kennenlernen zu wollen. Ganz sicher aber ein gutes Argument für die Qualität von Turbo-Pascal. (M.Thomas/ev)

Info: Turbo-Pascal ist ein eingetragenes Warenzeichen von Borland International. Vertrieb in Deutschland: Heimsoeth Software, Fraunhoferstr. 13, 8000 München 5, Preis 225 Mark.

Was uns gefiel

- Sicherheitskopie kann vom Besitzer angefertigt werden
- Editor mit den Qualitäten eines Textverarbeitungssystems
- Editor-Kommandos WordStar-kompatibel
- Extrem schneller Compilations-Vorgang
- Hohe Geschwindigkeit des übersetzten Programms
- Sehr komfortable Fehlerkorrektur
- Einbindung von Maschinensprache möglich
- Automatische Overlay-Technik
- Hoher Bedienungskomfort und hohe Bediensicherheit
- Wirth-Standard wird unterstützt
- Viele sinnvolle, zusätzliche Standardprozeduren und Funktionen
- Standardtyp String mit entsprechenden Funktionen implementiert
- Direkter Zugriff auf CP/M-System-Ebene und CPU-Register

- Hohe Genauigkeit bei reellen Zahlen (11 Stellen)
- Hoher Programmierkomfort durch viele Compiler-Optionen
- Automatisches Finden von Laufzeit-Fehlern
- Systemmeldungen editierbar
- Ausführliches, gut verständliches Handbuch
- Sinnvolles, größeres Beispielprogramm im Lieferumfang
- Unterstützung der Turbo-Pascal-Käufer durch den Hersteller
- Gemessen an der Leistung, sehr preiswert

Was uns nicht gefiel

- Handbuch im Taschenbuchformat wird intensive Benutzung nicht überstehen
- Kein Compilerprotokoll auf Drucker möglich

Die Testergebnisse:

	Compilation	Ausführung	
		m. Ausg.	o. Ausg.
Oxford-Pascal	3,7 sec	34,7 sec	2,3 sec
Profi-Pascal	24,8 sec (Disk)	25,1 sec	1,0 sec
Turbo-Pascal	0,8 sec	30,6 sec	0,4 sec

Doppelte Grafikauflösung für C 128

Ein kleines Maschinenprogramm macht dem 80-Zeichen-Videocontroller des C 128 Beine und bringt eine Grafikauflösung von 640 x 200 Punkten.

Wie Sie als stolzer C 128-Besitzer vielleicht wissen, besitzt Ihr Computer zwei Arten der Zeichendarstellung. Die üblichen 40 Zeichen pro Zeile und den 80-Zeichen-Bildschirm.

Schalten Sie auf 80 Zeichen um, so übernimmt ein anderer Video-Chip die Arbeit des uns vom C 64 her bekannten VIC II und zaubert ein 80-Zeichen-Feature auf den Bildschirm. Sehen können Sie dabei allerdings nur etwas, wenn Sie auch einen Monitor an der RGB-Buchse des C 128 angeschlossen haben, denn nur dort sind 80 Zeichen pro Zeile möglich. Der 8563-Videocontroller sorgt in diesem Modus für ein anständiges Bild.

Doch er kann noch mehr. Neben Buchstaben und Zahlen ausgeben ist er fähig, Punktgrafik zu erzeugen, und das in doppelter Auflösung, also statt mit den bekannten 320 mal 200 Punkten nun 640 mal 200 Punkte. Sie

haben richtig gelesen. Das sind insgesamt 128000 Bildpunkte, die einzeln ansprechbar sind. Prima, werden Sie sagen, der C 128 hat ja die vielen tollen Grafikbefehle ... Doch halt! Die Freude ist ein wenig verfrüht. Die doppelte Auflösung des 8563 wird nämlich unverständlicherweise von diesen Basic-Befehlen **nicht** ausgenutzt. Die ganzen fantastischen Grafikbefehle des C 128 sprechen nur die vom C 64 bekannte 320 x 200 Punkte-Grafik an. Warum das so ist, das weiß nur Commodore allein.

Auch der Versuch, die Punkte einfach in den Grafikspeicher des 8563 zu POKEN wird fehlschlagen, denn dieser Grafikspeicher ist vom Prozessor aus nicht ansprechbar. Der Videocontroller 8563 steht nämlich im Genuß eines eigenen Zeichenspeichers von 16 KByte, der nicht im normalen Adreßbereich liegt und nur ihm selbst zugänglich ist.

Doch ganz so eigenständig ist der VDC 8563 nun auch nicht. Es muß selbstverständlich ein Informationsaustausch zwischen Videoprozessor und dem übrigen Computer stattfinden können. Den gibt es natürlich auch.

Die Verbindung besteht allerdings nur aus 2 Byte im Input/Output-Bereich mit den Adressen \$D600 und \$D601. Durch sie hindurch drängt sich der gesamte Informationsverkehr von VDC 8563 und Computer.

Denn der VDC 8563 muß viel wissen, wenn er ein ordentliches Bild erzeugen will. Da er nur seinen »Privatspeicher« von 16 KByte kennt, kann er auf den Zeichengenerator nicht direkt zugreifen. Damit er trotzdem die Zeichen erzeugen kann, wird er beim Anschalten des Computers mit den nötigen Bytes aus dem Zeichengenerator über den Engpaß \$D600/\$D601 gefüttert.