

CPC forever

Ausgabe #2 - Dezember 2005





Zu viel Veränderung ist auch nix...

Wie schon in der #1 angedroht bleibt euch, Dank des durchweg positiven Feedbacks, eine Fortsetzung der CPC forever nicht erspart. Hier also, pünktlich vor Weihnachten, die Ausgabe #2. Diese ist nun so dick geworden, dass die durchaus realistische Gefahr besteht, dass das Weihnachts-übliche Familienleben darunter leiden muss. Immerhin ist das dann gut für die Figur... Zu verdanken ist dieser "dicke Brocken" einer kleinen Schar an fleissigen Schreibern ohne die das mal wieder alles nicht möglich gewesen wäre. Vielen Dank also von meiner Seite an all jene, die zu dieser Ausgabe beigetragen haben. Gleichzeitig dokumentiert diese, dass der CPC noch lange nicht tot ist. SymbOS, T-Rex und Justin sind nur einige Schlagwörter die eindrucksvoll das Gegenteil beweisen und den CPC auch ausserhalb der eigenen Szene erfolgreich zum Gesprächsthema machen. Insofern sollte auch einer dritten Ausgabe der CPC forever nichts im Wege stehen. Der ca. halbjährliche Rhythmus scheint sich zu bewähren, weswegen so ca. Mai/Juni mit der nächsten Ausgabe gerechnet werden kann. Artikel hierfür wie gehabt an:

cpc-forever@gmx.de.

Die üblichen Verdächtigen werde ich übrigens automatisch kontaktieren wenn's wieder Ernst wird, evtl. Neulinge (die immer gerne gesehen sind) würde ich bitten einfach mal bis Ende April Kontakt mit mir aufzunehmen und die Beiträge vorab anzukündigen.

Im Gegensatz zum Erstling gehe ich dieses Mal davon aus, dass ich es nach Verfassen des Editorials gebacken kriege euch das Ganze in einem .pdf-File zu präsentieren. Da ich euch aber nicht mit Veränderungen überfordern möchte habe ich mich wiederum dazu entschlossen, das Inhaltsverzeichnis einzusparen. Das erkläre ich hiermit direkt zu einer guten alten Tradition (passt wunderbar in die Weihnachtszeit) der CPC forever. Werde also eure wertvolle Zeit auch in Zukunft nicht für solche Nebensächlichkeiten in Anspruch nehmen (jaja, ich weiss schon, welche Nase mir vorschlagen wird ich solle aus diesem Grund das Editorial doch auch gleich weglassen... ;-)). Gerade rechtzeitig bin ich übrigens noch über ein furchtbares Foto gestolpert welches jetzt das Cover ziert. Auch so ein Stück Tradition...

Tja, dann bleibt mir nur noch euch "Viel Spass!" mit dieser zweiten Ausgabe zu wünschen und bis die Tage,

MARKUS aka VILLAIN OF HJT/BENEDICTION

MegaPhoenix

Wie ein Phoenix steigt er aus der Asche! Nach alten Mythen soll dieses Tier, welches der Schöpfung des Sonnengottes Ra entspringt, alle 500 Jahre nach Helipolis fliegen. Dort wird er sich auf dem Tempel des Ra ein Nest aus Myrrhen bauen.

In der Morgenröte bei Sonnenaufgang wird er in Flammen stehen und lebend verbrennen. Übrigbleiben wird nur ein Rest von Asche. Am Abend des dritten Tages wird er neugeboren aus der Asche emporsteigen und als Auge und Zorn des Sonnengottes erneut über die Menschen kommen. Und siehe da, sind wir doch verschont geblieben, unsere Nachfahren bekommen es richtig dicke ab! In ferner Zukunft kommt er wieder und wird versuchen uns zu vernichten. Und nicht nur die Menschen sind fortschrittlicher, nein auch der Tempel ist mit großen Düsen direkt ins All gewandert.



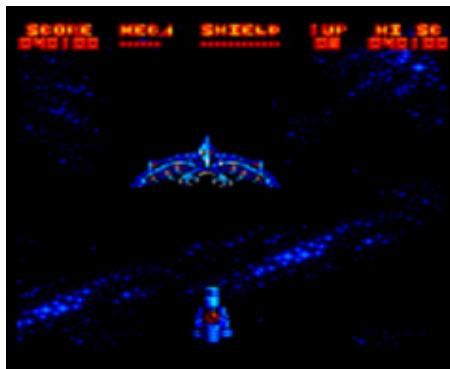
Das Titelbild weckt berechnete Hoffnungen auf eine feine Grafik

Und an diesem Ort kommen wir ins Spiel, bzw. an den Joystick. Natürlich, als der beste und einzige Pilot, der die Gefahr abwenden kann! Sollten wir schnell genug sein, könnte es uns sogar gelingen. Unser Raumschiff welches wir von der Erde bekommen haben hat eine waffenstarrende Artillerie, einen Schutzschild, aber auch einen Haken. Wir können nur links und rechts fliegen! Die Ingenieure haben sich da mal wieder mit Rum bekleckert, Vorwärtsgang wurde vergessen, und Rückwärts ebenso. Habt Ihr so schon mal versucht zu parken ohne diese Richtungen? Den Hangar möchte ich nicht sehen, und auch nicht die Lackschäden an den

Gleitern!

MegaPhoenix ist ein Remake des klassischen Phoenix von Taito anno 1980. Ein reinrassiges Shoot em Up das mit mehr Waffenpower und der Taktischen Nutzung eines Schutzschildes aufpoliert wurde.

Freunde des gepflegten Triggerns kommen ganz auf Ihre Kosten. Es gibt genug Level zum Austoben. Jeder Level besteht aus 6 Sequenzen, vier davon sind mit Flugformationen kleinerer Gegner voll gestopft. Anschließend versucht uns der Phönix, das Blech unter dem Hintern wegzufeuern und als wäre das nicht genug, bekommen wir es noch mit dem Geiste Ra's zu tun der uns mit seinem ganzen waffenstarrenden "Raumschiff" angreift.



Zum Glück können wir immer nachdem wir den Phönix erlegt haben ein Power up einsammeln, welches unser Piloten Herz erwärmt, rüstet es doch unser Raumschiff nacheinander mit mehr Feuerkraft und Satelliten auf. Damit macht's dann noch mehr Spaß! Aber größer bedeutet auch gefährdeter und so hilft uns ein Schutzschild als taktisches Mittel. Sollten wir einmal einen Gegner nicht treffen, können wir durch aktivieren unseres Schildes den Gegner einfach an uns

zerschellen lassen. Der Schild hält eine kurze Zeit an, macht uns aber währenddessen bewegungsunfähig. Nach einer kurzen Aufladephase steht er wieder zur Verfügung und wir sind wieder einigermaßen sicher. Für Langsamdrücker gibt es auch eine Dauerfeuerfunktion um deren zarte Fingerchen zu schonen, aber wer braucht das schon!

Der CPC verkraftet die ganze Aktion ganz ordentlich. Es kommt so gut wie nie zu stärkeren Rucklern, oder „Slowdowns“ der Grafik. Das ganze Spielgeschehen bleibt immer flott und hält den Schwierigkeitsgrad



Detailreiche Gegner

immer fair. Die Gegner sind hübsch gezeichnet und animiert. Unser eigenes Raumschiff sieht super aus und wirkt mit stärkerer Bewaffnung richtig beeindruckend. Schade nur, dass sich die Level immer wieder bis auf die Anhebung der Feuerfrequenz und der Geschwindigkeit der Gegner kaum verändern.

Nichtsdestotrotz kann die Grafik sich sehen lassen und hat man mal die Taktik mit dem Schutzschild kapiert macht das Ballern richtig Spaß. Die Motivation den nächsten Level und damit das nächste PowerUp für das Raumschiff zu bekommen ist hoch und verleitet einen gerne noch einmal in den Kampf zu ziehen. Super auch der Soundtrack der dem Geschehen angepasst schnell und auch auf die Dauer nicht störend ist. Trotzdem kann er während des Spiels abgeschaltet werden. Alles in allem ein feines Spiel, das bei Normalen- sowie den Hardcore-Joystickrüttlern, für Kurzweil sorgt.

JS/Tolkin

Justin

Neue Spiele braucht das Land, und wenn es so tolle sind, dann immer her damit!

Justin ist ein brandneues Spiel für den CPC, programmiert von CNGSoft. D.h. es steht seit dem 12.12.2005 zum Download auf der Seite <http://cpce.emuunlim.com> bereit. Kursierte vor wenigen Wochen noch eine Demo zu diesem Spiel, ist das nun die fertige Version. Gespannt heruntergeladen und anschließend per CPC-Booster auf eine 3-Zoll-Floppy übertragen startete ich das Spiel. Nach wirklich angenehm kurzer Ladezeit wird der Titelschirm eingeblendet, welcher von „Circus“ gezeichnet ist. Ein paar kurze Sekunden später und nach Druck auf die Leertaste springt mir das Titelmü entgegen. Es gibt einiges zum Einstellen, wie: Joystick oder Keyboard, einem Info-Bildschirm, Demo und Start. Wirklich klasse ist die Sprachauswahl, die eigentlich nichts offen lässt, kann man hier doch zwischen Spanisch, Englisch und Französisch unterscheiden.



Das Spiel ist inspiriert von dem Spieleklassiker Dustin, welcher 1986 von Dinamic veröffentlicht wurde. In „Justin-The Game“ geht um Justin, „einen Einbrecher, der in eine isometrische Villa mit einer ganzen Menge von Räumen einbrechen will, um dort vier versteckte Safes zu knacken und wieder lebend heraus zu kommen. Dazu benötigt er viele Schlüssel, eine gehörige Portion Reaktionsvermögen, viel Glück und viel Geduld.



Grafisch gehört das Spiel zur Oberklasse was am CPC so auf Disketten herumliegt. Es passt super in die Riege der Spiele die von dem Klassiker „Head over Heels“ angeführt wird. Anfangs noch skeptisch, war ich doch von Mode 0 Iso-Spielen wie Jak&Daxter mehr als abgeschreckt, fand ich die Grafik übersichtlich und wirklich schön. Die detailreichen Wände und Gegenstände sind farbenfroh und passen gut zum Thema. Die Sprites sind ausreichend animiert sehr detailreich. Nur Justin sieht irgendwie wie ein Pantomime auf Speed aus, wenn er sich bewegt. Ein paar mehr Animationsphasen, so eine pro Bewegung, ständen ihm wirklich gut. Da ich grundsätzlich gerne die Übersicht bzw. die Koordination bei isometrischen Spielen verliere, gerade wenn man auch noch zielgenau springen will, habe ich mich anfangs noch sehr dumm angestellt dorthin zu springen, wohin ich wollte. Mit etwas Übung ist das aber dann schnell gelernt und man kommt recht gut



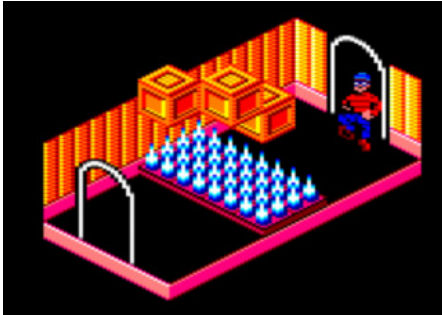
voran.

Mein Versuch mit dem Joystick hier lebend durchzukommen, scheiterte kläglich, da ich irgendwie immer die Richtungen beim Springen nicht halten konnte. Umgestiegen auf die Tastatur und nun wurde das hüpfen zu Freude. Lasst also euren Joystick im Schrank und nehmt die Tastatur, dann habt Ihr wirklich eine Chance. Gebt euch etwas Eingewöhnungszeit und lasst das Spiel nicht gefrustet in der Ecke liegen. Es lohnt sich wirklich. Die Kollisionsabfrage ist genau und es ist problemlos möglich durch Labyrinth von Stacheln oder Pflanzen zu manövrieren. Die Sprungeinlagen sind knifflig aber fair sobald man kapiert hat dass viele Blöcke nach dem ersten Berühren kaputt gehen. Sollte man hier mal einen Block berührt haben und ihn zum weiterkommen brauchen, hilft ein kurzes Verlassen des Raumes, danach ist er wieder da. Richtig ärgerlich wird die Sache aber, sobald Blöcke die gleich aussehen einmal begehbar sind und andere die genauso aussehen, gleich beim erste auftreten sich in Wohlgefallen auflösen. Hier muss man leider viel mit „Try und Error“ vorgehen, was einen etwas gefrustet im Regen stehen lässt. Auch ist komisch, dass wir ein Leben aushauchen, sobald wir eine der Pflanzen berühren die so im Haus herumstehen. Es ist aber wohl erklärbar, das unser Held eine starke Allergie gegen Farne hat, und der Farn auch ein ganz extrem giftiger seiner Art ist. Eben bei Berührung tödlich!

Die Gegner-Sprites sind wie schon erwähnt gut animiert, und sehr detailreich. Viele sind berechenbar wie die umherstreunenden Polizisten oder die putzigen Mäuse, doch treiben den Spieler die Riesenspinnen, welche wohl auf Pillen total verplant umhersteuern, zur Weissglut.

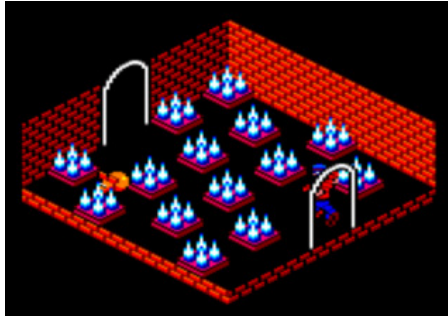
Im Gegensatz zur Grafik ist die musikalische Untermalung etwas zu kurz gekommen. Hier kommen nur Töne aus dem Lautsprecher, die der CPC schon 1985 gespielt hat. Das Rumgestampfe mit Basic-Tönen zu untermalen ist wirklich grausig, und macht es einem leicht den Lautsprecher, bzw. alles was einen Ton am CPC abgibt, auszuschalten. Wirklich schade darum.

Überhaupt ist der Frustrfaktor und



der Schwierigkeitsgrad sehr hoch angesetzt. Was es den wenigsten Spielern ermöglicht hier eine lang anhaltende Motivation zu finden, alle Schätze in den Safes zu holen

und sich dann zurück ins Freie zu schleichen. Hier wäre wohl noch etwas Spieletesten vor dem Release nötig gewesen.



Als großes Plus verzeichnet Justin für sich, nach so langer Zeit wieder ein komplettes Spiel zu sein. Es kann sich problemlos mit Kaufspielen messen, ist es doch um Längen besser als die meisten Spiele die zur Zeit von „RetroSoftwarefirmen“ vertrieben werden und um Längen besser als alte Iso-Spiele. Alles in allem ist Justin spielerisch und grafisch ein Highlight, was aber von der Soundkulisse und dem hohen

Schwierigkeitsgrad zurück auf den Boden geholt wird. Es hätte das Zeug zum absoluten Hit.



Vielen Dank für ein neues Spiel, und vielen Dank für die Zeit und die Arbeit die darin investiert wurde. Es freut alle, wieder ein neues Spiel zu sehen, was nicht auf dem „Tetris Niveau“ liegt und das zeigt, was man aus dem CPC so alles Grafisch herausholen kann.

JS/Tolkien

News

So, und falls hier jemand meinen sollte, das wäre schon alles was am CPC noch so geht dann kann ich nur sagen: weit gefehlt!

Fatality hat sich wieder mit einem neuen Crack zu Wort gemeldet. Dieses mal musste Eyespy dran glauben. Der Crack hat ein hübsches Intro mit einer ebenfalls hübschen Musik (vom Spectrum konvertiert). Man spricht davon, dass Fatality eine finnische Gruppe wären. Sachen gibt's! Download übrigens unter www.cpcscene.com

Climax-G ist eine Slideshow speziell für Grünmonitore. Dahinter steckt Super Sylvestre, der etwas abgedrehte Autor der "experimentellen" Slideshow Climax. Sylvestre will diverse Tricks benutzen um am CPC photorealistische Bilder darzustellen. Release ist für Januar 2006 geplant. Da sind wir doch mal schwer gespannt!

Sylvestre ist auch genau jener

welcher, der kürzlich in einem Kino in Westfrankreich CPC-Demos vorgeführt hat. Da das wohl durchaus ein Erfolg war plant er eine Wiederholung der Aktion.

Auf dem kürzlich in Frankreich abgehaltenen CPC at Work! Meeting hat Two Mag wohl eine neue Version von Overflow's Backtro mit Grafiken von Beb gebacken. Eigentlich hätten diese schon in der ersten Version enthalten sein sollen aber sie wurden wohl nicht rechtzeitig fertig. Wann nun die neue Version des Backtros released wird ist unklar.

Auf dem CPC at Work! wurde auch eine Coding-Competition abgehalten. Die drei entstanden Werke werden demnächst von Plissken/Revival unter's Volk gebracht.

Targhan hat wohl einige (streng geheime...) Previews seine Game-Projektes gezeigt. Bis zur Fertigstellung gibt's anscheinend noch einiges zu tun, aber Targhan ist als flotter Coder-Finger bekannt.

Ram 7, Macher der bekannten RAMcard, codet an einem neuen Hack-It-ROM.

Another World, fanzösisches Fanz das eigentlich im Zwei-Monats-Rhythmus sollte und von Tom's und Hick's gemacht wird soll bald wieder auf die CPC-Bühne zurückkehren. Schnell mal französisch pauken...

Aus dem Hause Cronosoft gibt's demächst ein neues Puzzle-Game für den CPC. Screenshot ist schon im Forum von www.cpczone.net zu sehen.

CRTC arbeitet immer noch an der Palatine-Demo... ;-)

Eliot & Villain/Benediction

Odiesoft/HJT

Einer der produktivsten und innovativsten deutschen CPC-Szener lässt zumindest auf den folgenden Seiten die gute alte Zeit noch mal aufleben...

CPC forever: Hallo Odie, vielen Lesern dürftest Du ja zumindest namentlich bekannt sein. Vielleicht erzählst Du allen anderen nochmal kurz, was Du am CPC alles so getrieben hast.

Odie: Hallo. Danke für das Interesse an meinen CPC Projekten. Ich denke, dass ich so alles in allem an die 40 Demos, 3 fertige Spiele und zahllose angefangene, aber nie beendete Spiele geschrieben habe sowie ein paar kleinere Anwendungen, wie ein Sprite-Designer, ein Malprogramm, eine Videokassetten- und Computerspiele-Datenbank (alle unveröffentlicht). Darüber hinaus habe ich auch ein paar Spiele geknackt und mit Trainern versehen, aber so wirklich veröffentlicht habe ich diese Cracks meistens nicht. Mir ging es eher darum, die Cassettenspiele auf Diskette zu übertragen, damit ich die schneller laden konnte.



Odiesoft... (damals Anfang der 90er)

CPC forever: Wie und wann begann eigentlich Deine persönliche Computergeschichte und warum bist Du damals am CPC (und nicht wie viele andere am C64) gelandet?

Odie: Angefangen hat alles im Herbst 1984. Meine Eltern haben meinen Geschwistern (ein Bruder und zwei Schwestern - alle älter als ich) einen CPC464 mit Farbmonitor und einem Diskettenlaufwerk geschenkt, um uns bei den

Hausaufgaben zu helfen (was für eine blöde Idee, oder?) und damit wir in der Zukunft mal einen Vorteil gegenüber anderen haben (wie wahr das sein würde, habe ich dann aber erst Jahre später begriffen...). Meine Eltern hatten den CPC als Computer ausgesucht, da der einen hochauflösenden Textmodus hatte und sie wirklich gedacht haben, dass wir damit unsere Hausaufgaben machen könnten. Wir haben anfangs natürlich alle versprochen, dass wir den zum Arbeiten und nicht zum Spielen verwenden werden - ich muss aber gestehen, dass ich das Anfangs auch noch geglaubt habe. Dieser Glaube hat aber nicht lange angehalten...

Natürlich haben wir erst einmal mit Basic rumgespielt, Farben verändern, Linien ziehen lassen und Töne ausgegeben - was man eben so macht, wenn man seinen ersten Computer hat. Ach ja, und DR Logo (wir haben immer "Doktor Logo" dazu gesagt, dabei heißt "DR" Digital Research...) haben wir auch gelernt, das war vielleicht ein Unsinn, aus heutiger Zeit gesehen. Die Schildkröte als Symbol für das Programm war schon sehr treffend, finde ich.

Bald entdeckten meine Geschwister und ich Computerspiele und wurden von diesen recht bald in Ihren Bann geschlagen. Ich weiß noch was für ein Gefühl das war, das erste Mal Manic Miner (mit Trainer) durchzuspielen - der letzte Level war echt die Hölle! Oder Spiele wie Sorcery, Jet Set Willy, die ganzen Roland Spiele, etc. das war schon lustig. Bei all diesen Spielen gab es aber immer wieder Dinge die mich geärgert haben, Manic Miner/ Jet

Set Willy fand ich zu schwer, Sorcery zu kurz, die Roland Spiele oft zu repetitiv oder auch zu schwer, Textadventures wie Marco Polo zu schlecht gemacht, The Way of the Exploding Fist hatte zu wenig Abwechslung, The Way of the Tiger war mir zu schwer, irgendwann hatte ich immer an jedem Spiel irgendetwas auszusetzen. Für mich gab es nie das perfekte Spiel und ich wollte gerne in der Lage sein, diese Spiele so zu verändern, dass sie meinen Wünschen gerecht werden würden. Mit Trainern (die damals noch "Pokes" hießen) konnte man schwere Spiele leichter machen, aber schlechtes Leveldesign, schlechte Grafik oder fehlende Eigenschaften konnte man eben nicht in ein Spiel hinein-poken. Also wollte ich irgendwann selbst ein Spiel schreiben. Da ich aber nicht wußte, wie man bewegte Figuren auf den Bildschirm bringt, fing ich zuerst an ein Textadventure zu schreiben. Ich wollte ein "Der Graf von Monte Christo" Adventure schreiben. Dieses Unterfangen dauerte aber nicht lange. Ich fing damit an über einige Plot und Draw Kommandos das Gefängniszimmer zu malen, in welchem das Spiel beginnen würde. Ich hatte schon ein paar simple Frage und Antwort Routinen in Basic geschrieben, die auf fest vorgegebene Eingaben reagieren konnten, aber wie man einen richtigen Parser schreibt, wußte ich zu der Zeit natürlich noch nicht. Ich stieß auch schon auf große Probleme, als ich mir vornahm eine animierte Ratte über den Boden der Gefängniszelle laufen zu lassen. Dem Monte Christo Spiel wurde ein rüdes Ende bereitet, als mir die Diskette mit den Originaldateien bei einer Vorführung vor Freunden im Spielwarenladen geklaut worden ist. Im selben Jahr (es war glaube ich 1987) ließ ich mir von meinen Eltern zu Weihnachten den Sybex Assemblerkurs schenken und lernte damit Schritt für Schritt wie man Assembler programmiert

und mehr aus dem Computer heraus holt. Nun ja und von da an ging es dann los mit meiner Karriere als CPC Spiele und Democoder...

CPC forever: Ich kann mich noch gut an die nicht enden wollende Entwicklungszeit Deines Mammutprojektes Megablasters erinnern. Wenn man's im Nachhinein betrachtet, inwiefern hat sich das für Dich ausgezahlt?

Odie: Der Begriff "Mammutprojekt" trifft es gut. Eigentlich hatte ich Megablasters nie als so ein umfangreiches Projekt geplant. Wie so oft hatte ich auf einem anderen Computer das Spiel gesehen und mir gedacht, dass man da doch noch einiges verbessern könnte. Mein Bruder hatte sich auf dem Amiga das Spiel "Dynablasters" besorgt und wir haben das sehr gerne gespielt, weil das einfach ein tolles Spielprinzip ist. Eines der ersten wirklich guten Mehrspieler (nicht nur Zweispieler) Spiele (mal abgesehen von Midi-Maze, aber das kannte ich damals nur aus Zeitschriftenberichten), Was mich an Dynablasters auf dem Amiga jedoch immer geärgert hatte war, dass das Spielfeld im Vierspielermodus so klein war und man sich gar nicht richtig mit Extras aufladen konnte, bevor man sich gegenseitig an den Kragen ging und die mangelnde Möglichkeit die Gegner im Mehrspielermodus abzuschalten, d.h. wenn man mit zwei bis vier Spielern gespielt hat, dann liefen immer noch zwei bis vier Computergesteuerte Gegner herum, die einen ständig nur im Weg waren oder einen kaputt gemacht haben, bevor man sich mit seinen menschlichen Mitspielern aus der Weltgeschichte bomben konnte. Erst ab fünf Spielern (das war auf dem Amiga ein besonderes Feature, dass man zu fünft spielen konnte) verschwanden diese nervigen, zufallsgesteuerten Gegner. Ich wollte zuerst nur den Dynablasters

Vierspieler Modus auf den CPC umsetzen und dabei das Spielfeld vergrößern, so dass man sich dann hochgerüstet mit vielen Extras erst gegenüber stehen würde.

Die ursprüngliche Version von Megablasters war auch im vierfarben Mode 1 programmiert und bot ein riesiges Spielfeld (verglichen mit der endgültigen Mode 0 Version). Zum einen stellte ich dabei fest, dass das Spielfeld zu groß war und es zu lange dauerte, bis man sich traf und zum anderen lernte ich Kangaroo auf dem GOS IV Meeting in Bad Hersfeld kennen als ich das Spiel dort vorführte und der fragte mich, ob ich das Spiel nicht verkaufen wolle. Da ich eigentlich schon immer mal reich und berühmt mit einem Computerspiel werden wollte (David "Elite" Braben war mein großes Vorbild...) sagte ich ohne lange nachzudenken zu und begann damit das Spiel auf Mode 0 umzuprogrammieren und mir Gedanken über einen Story-Modus zu machen. Und der Rest ist mehr oder weniger Geschichte (bzw. kann auch auf meiner Webseite www.odiesoft.de nachgelesen werden - das würde den Rahmen dieses Interviews sicher sprengen).

Nun ja, reich bin ich durch Megablasters nicht geworden. Wenn ich alle Einnahmen zusammen

zähle, habe ich vielleicht an die 1500 DM damit verdient. Bei guten zwei Jahren (mit Unterbrechungen) Entwicklungszeit ist das ein ziemlich bitterer Stundenlohn, aber mir ging es irgendwann gar nicht mehr um's Geld. Ich wollte das Spiel einfach nur fertig bekommen, einfach um den CPC'ern die schon darauf warteten und mir selbst zu beweisen, dass ich es kann. Wenn ich jetzt im Nachhinein darauf zurück blicke, denke ich mir immer wieder, "Wie hast Du das nur geschafft?". Als ich auf der Xzentrix 2001 mal wieder in die Source Codes von Megablasters reingeschaut habe (ich dachte mir, vielleicht könnte ich ja eine Xzentrix Sonderedition vom Vierspieler Modus mit ein paar neuen Features machen) muß ich gestehen, habe ich nicht mehr zusammen bekommen, wie das alles funktioniert hat. Die Sources waren einfach zu sehr verstreut und zum Teil habe ich nur noch Binärdateien von manchen Unterroutinen gefunden. Ausgezahlt hat sich dieses Spiel aber schon. Zum einen bin ich immer noch ein guter Bomberman Spieler (derzeit habe ich Bomberman Generation auf dem Gamecube) und ich kann mit Fug und Recht behaupten, dass ich mal ein halbwegs kommerzielles Spiel geschrieben habe - das war eigentlich immer mein Traum



Megablasters im Vier-Spieler-Modus. Party-Spass vom Feinsten!

gewesen. Er hat sich dann zwar anders erfüllt als, ich mir das ursprünglich vorgestellt habe (wäre meine Vorstellung wahr geworden, dann würden wir dieses Interview jetzt in meinem Penthouse in einem New Yorker Nobelviertel führen ;-), aber er ist wahr geworden. Das Schlimme mit Wunschträumen ist nur, dass jeder Traum, den man sich erfüllt, ist ein Traum weniger, den man träumen kann. Nach der Veröffentlichung von Megablaster bin ich irgendwie in ein Motivationsloch gefallen, ich hatte endlich mal ein Spiel geschrieben und veröffentlicht und das erreicht, wozu ich Programmieren gelernt hatte und dann war meine Motivation weg. Mit Rex hatte ich dann zwar noch angefangen das Spiel "Rigor Mortis" (so ein Zombie-Prügelspiel) zu schreiben, aber mit dem Beginn meines Studiums war nicht nur die Motivation sondern auch die Freizeit weg. Eine Demo und ein paar Grafik-Studien kann man sich von meiner Webseite runter laden.

CPC forever: Offiziell ist Deine aktive CPC-Zeit ja beendet. Juckt's denn manchmal noch in den Fingern?

Odie: Wie ich schon bei der letzten Frage geschrieben habe, hatte ich bei der Xzentrix 2001 daran gedacht mal wieder was zu machen, aber ich hatte nicht mehr die Zeit und Geduld mich wieder darin einzuarbeiten. So bleibt der aufgemotzte Vierspielermodus (in welchem man Bomben hätte kicken oder werfen können und die Spieler sich beamen können und in welchem es einen Felder-Einfärben Spielmodus gegeben hätte, bei welchem man versuchen muss innerhalb einer vorgegebenen Zeit möglichst viele Felder durch das Explodieren der eigenen Bomben mit der eigenen Farbe einzufärben) nur ein nettes Gedanken Experiment. Nun ja, und ich hatte auch mal geplant eine

China-World Sonderedition rauszubringen, in welchem man volle 8 Level (fünf normale, ein Bonus- und einen Secretlevel sowie den Endgegner, einen Sumo Ringer) spielen kann. Aber das ist bisher nur beim Gedanken geblieben und wird sicher auch nicht mehr in die Tat umgesetzt. Dass ich diesen Chessboard Zoomer nie fertig programmiert habe, wurmt mich auch immer noch. Neben dem regulären Chessboard hatte ich geplant so eine Art von Wellenfunktion (siehe Erklärung auf meiner Webseite) einzubauen. Ich war und bin auch immer noch der Überzeugung, dass das auf dem CPC möglich sein müsste.



Das oben angesprochene Chessboard

Und unter uns gesagt, hatte ich auch eine Weile lang darüber nachgedacht, ob eine Wolfenstein 3D Umsetzung auf dem CPC möglich sein könnte. Aber das war zu einer Zeit, in welcher das Studium für mich schon wichtiger geworden war als die CPC Programmierung. Zur Zeit verspüre ich kein großes Bedürfnis mehr, am CPC noch was zu programmieren. Ab und an sehe ich mir Demos auf einem Emulator an, aber einen echten CPC habe ich nicht mehr zu Hause stehen. Den letzten habe ich im Sommer diesen Jahres verkauft - d.h. so ganz richtig ist das nicht, bei meinen Eltern zuhause steht noch der original CPC 464 von 1984 mit Farbmonitor und einem Diskettenlaufwerk. Aber da wird der auch noch eine Weile stehen werden... Meine Prioritäten haben sich geändert, ich programmiere

zwar immer noch gelegentlich am Computer, aber es gibt mir nicht mehr dieses Gefühl, das ich früher dabei empfunden habe.

CPC forever: Du warst ja für einige recht unkonventionelle Sachen (Wishing Well Construction Kit zum Beispiel) bekannt, hast quasi immer einen eigenen Stil gepflegt. Was hättest Du denn gerne mal noch am CPC umgesetzt? Was spukt Dir vielleicht immer noch im Kopf rum?

Odie: In vorangegangenen Fragen habe ich schon einiges über unvollendete oder nicht angefangene Projekte geschrieben. Wie gesagt, ein Wolfenstein 3D wäre noch ein schöner Knaller für meine CPC Karriere gewesen, war mir dann aber im Endeffekt ein zu hoher Aufwand. Spielideen hatte ich sehr viele, einiges davon machbar, andere Sachen utopisch. Ich hatte lange Zeit davon geträumt ein echtes 3D Spiel zu machen, in welchem man fast grenzenlose Bewegungsfreiheit hat und praktisch einen ganzen Planeten zum erforschen hat und dass natürlich in fotorealistic Grafiken. Der CPC hätte niemals ausgereicht, um derartige Phantasien in die Realität umzusetzen, ich habe mir zwar immer Gedanken gemacht, wie man eine echte 3D Welt mit entsprechenden Abstrichen auf dem CPC machen könnte, aber mehr als ein paar Versuche mit Vergrößerungs-/ Verkleinerungsroutinen und einem angefangenen Rollenspiel (-> Webseite) sind dabei nicht rausgesprungen. Mein zweiter großer Wunschtraum neben dem Veröffentlichen eines kommerziellen Computerspiels war eine eigene Megademo zu schreiben. Ich hatte schon recht früh angefangen, erste Teile für eine Megademo zu programmieren, aber mit zunehmenden Fähigkeiten waren mit die alten Parts nicht mehr gut genug und viele Demoparts für die geplante Megademo habe ich dann

einzel veröffentlicht oder zu anderen Megademos beigesteuert. Der erste Anlauf zu einer Megademo hieß lediglich "Odie'soft Megademo", der zweite Versuch sollte "Dangerous" heißen, aber mehr als ein paar später ebenfalls einzeln veröffentlichte Teile sowie ein angefangener Abschlußpart für die Demo sind dabei nicht rausgekommen. Mit dem Spiel "Rigor Mortis", welches Rex und ich 1995/96 noch angefangen hatten, wollte ich eigentlich ausloten wie viel Blood and Gore auf dem CPC machbar sind. Meine Intention war es ein ultra-brutales Prügelspiel zu schreiben, in welchem der eigene Spielcharakter je nach Art und Umfang der erhaltenen Verletzungen mehr und mehr auseinander fallen würde und mitunter nur noch als Torso ohne Beine über den Boden kriechen würde. Neue Körperteile sollte man dazu gewinnen können, in dem man sie den Gegnern oder unbeteiligten Passanten abreißen würde. Dabei sollte dann immer viel Pixelblut spritzen und auf der Straße liegen bleiben - damals war ich etwas komisch drauf und wollte offenbar die Grenzen des Machbaren und des guten Geschmacks ausloten. Aber dazu kam es dann doch nicht... Was mir auch heute immer noch ein wenig im Kopf als Spielidee rumspukt wäre eine richtig gute und moderne Elite Version mit Multispielermodus und der Möglichkeit auf Planeten zu landen und dort neue Kulturen kennen zu lernen und Abenteuer zu bestehen. Quasi ein Spiel ohne Ende, bei dem man so lange durch den Weltraum fliegt, Handel treibt und mit neuen Kulturen zusammen trifft, bis man sich zur Ruhe setzt - eben so eine Art Spiel des (Weltraum)lebens. Zum einen gibt es inzwischen Online Rollenspiele, die dieser Idee sehr nahe kommen und zum anderen bräuchte ich wahrscheinlich ein Team mit 100 Leuten, um ein derartiges Vorhaben

in der heutigen Zeit umzusetzen - mal ganz zu schweigen davon, dass das natürlich nicht auf dem CPC rauskommen würde...

Eine Idee, die ich neulich mal hatte, aber die auch nichts für den CPC wäre (oder wenn, dann nur in einer sehr abgespeckten Version) wäre ein "Was wäre wenn" Spiel. Eine Art Spiel des Lebens, bei welchem man Anfangs so einige Grundbedingungen seines Charakters festlegt (Geschlecht, Name, Familie, Wohnhaus und Ort, wichtige Lebensereignisse) und dann sein Leben spielt. Am Ende des Spiels kann man dann zu bestimmten Ereignissen im Leben zurück gehen und sich anders entscheiden, als man es im Spiel hat, und dann weiter spielen und sehen, was sich am Lebensweg ändert. Quasi so eine Art Sims (ich muss dazu sagen, dass ich die Sims noch nie gespielt habe und nicht weiß, ob sowas da schon geht oder nicht), bei dem man zurück spulen kann und falsche oder richtige Entscheidungen ändern kann und sehen kann, was dann aus einem wird. Jeder fragt sich ja mal, was wäre aus mir geworden, wenn dieses oder jenes nicht passiert wäre. Ich finde das hat einen gewissen Reiz wenn man das mal spielerisch ausprobieren könnte.

CPC forever: Auch wenn Du selbst eigentlich nicht mehr aktiv bist, verfolgst Du ja immerhin noch, was sich rund um den CPC so abspielt. Was sagst Du zu aktuellen Projekten wie zum Beispiel SymbOS und SYMBiFACE?

Odie: Ich lese mehr oder weniger regelmässig die eMails die in der CPC-Mailingliste (CPC-Telegramm) rumgehen und muss sagen, dass ich es absolut krass finde, wie viel Zeit und Aufwand Leute wie Prodatron, Dr. Zed, Tobiflex, etc. noch alle in den CPC stecken. Wäre so etwas wie SymbOS oder das

SYMBiFACE um 1990 herum rausgekommen, dann wäre das sicher ein großer Verkaufsmotor für den CPC und den CPC+ geworden. Es ist echt irre, was Prodatron da mal "so nebenbei" (so wirkt es immer) auf die Beine stellt. Ich glaube nicht, dass ich noch mal so viel Begeisterung für den CPC aufbringen könnte und habe alle Achtung vor Prodatrons Aufwand und Energie, die er da reinsteckt. Ab und an schaue ich auch auf Pouet nach neuen CPC Sachen oder auf diversen Französischen Seiten - wie gesagt, schaue ich mir die Sachen dann aber immer nur auf dem Emulator an, auf dem nicht immer alles läuft. Auf Parties gehe ich auch ab und zu noch, wenn ich die Zeit habe. Aber im Januar auf das Treffen werde ich nicht kommen, da bin ich zu einer Taufe eingeladen...

CPC forever: Wenn man die gesamte Geschichte des CPCs so betrachtet, was sind für Dich persönlich die absoluten Highlights?

Odie: Der CPC war und wird immer ein fester und bedeutender Bestandteil meiner Jugend sein. Es gibt viele Momente an die ich mich noch genau erinnere, auch wenn ich diese nicht wirklich als Highlights bezeichnen würde, so kann ich mich daran erinnern, wie ich endlich mal einen Poke für "The fifth Axis" gefunden hatte und das Spiel durchgespielt habe während "Hart aber Herzlich" im Fernsehen lief - frag mich nicht, wieso ich mich ausgerechnet daran erinnere, aber irgendwie ist das haften geblieben. Ich erinnere mich an den Computerunterricht in der Schule am CPC 6128, wo wir den Lehrer verarscht haben mit einem Programm, dass die Einschaltmeldung des CPC nachmacht und dann aber bei jedem eingegebenen Befehl entweder ein "Syntax Error" oder einfach nur "Ready" ausgegeben hat und welches den drei-Finger-

Resetgriff abgefangen und sich dann neu gestartet hat. Der Spaß war vorbei, als der Lehrer dann den Computer ausgeschaltet und neu gestartet hat, aber das war auch lustig. Die Zeit bei Marabu in Schloß Sommershausen wird mir nicht nur wegen der CPC Parties dort immer als etwas ganz besonderes in Erinnerung bleiben. Es sind also nicht unbedingt Spiele oder Demos auf dem CPC, die mir wichtig sind, sondern Ereignisse rund um den CPC, die mir viel bedeuten. Ich könnte natürlich viele Spiele aufzählen, die mir gefallen haben, wie zum Beispiel mein ewiges Lieblingsspiel "Head over Heels" oder "Sorcery+", "The Castle of Dr. Destructo", "Green Beret", "The way of the Exploding Fist" und andere, die ich bis zum Abwinken gespielt habe. Oder die süsse Ironie des Schicksals als ich nach einem Poke für das Spiel "Rambo First Blood Part 2" gesucht habe und es, bevor ich einen Cheat gefunden habe beim Testen eines eingegebenen Pokes das Spiel doch noch ohne jegliche Betrügereien durchgespielt habe. Dasselbe ist mir auch bei Kung-Fu Master passiert. Alles in allem war der CPC ein prägender Faktor in meiner Kindheit und Jugend und das werde ich nie vergessen.

CPC forever: Was macht Odiesoft eigentlich heute? Immer noch "irgendwas" mit Computern?

Odie: Im Moment versuche ich meine Dissertation im Bereich Sozialpsychologie und Motivation an der Uni Konstanz endlich zu einem guten Abschluß zu bringen, was hoffentlich gegen Mitte 2006 der Fall sein wird. Danach bin ich noch recht unschlüssig, aber ich werde versuchen einen Job in Hamburg zu finden, wo meine Lebensgefährtin seit gut einem Jahr lebt. Ich wollte versuchen etwas in einem Bereich zu finden, in dem ich meine Psychologische Ausbildung mit

meinen Informatik-Fähigkeiten verbinden kann. Das könnte so in Richtung Informationsverarbeitung und -darbietung, im Bereich Lernsoftware oder vielleicht auch bei einer Firma, die speziell Software und Methoden für die Psychologische Forschung entwickelt sein. Aber genaueres kann ich dazu nicht sagen. Meine Gesichtsfeld hört derzeit beim Ende meiner Dissertation aus, weiter blicke ich gerade noch nicht.

CPC forever: Gibt es etwas, was Du dem CPC verdankst? Etwas das in dieser Form nie geschehen wäre, wenn Du keinen gehabt hättest und woran Du Dich auch heute noch gerne zurückerinnerst?

Computer aufbringe, was nicht viele haben. Darüber hinaus habe ich über den CPC viele Freunde und meine erste Freundin kennen gelernt. Das waren wichtige und interessante Erfahrungen. Zu manchen Freunden von damals habe ich noch regelmässig Kontakt, andere sehe ich nur auf CPC Parties - aber ich weiß diese Freund- und Bekanntschaften durchaus noch zu schätzen.

CPC forever: Möchtest Du den Lesern zum Abschluss noch irgendwas mitteilen?

Odie: Ich warte schon auf die Einladung zur 30 Jahre CPC Feier ;-)



*Odiesoft (ganz links) auf der Castle Party 1 im Kreise seiner Lieben... ;-)
Darunter u. a. Prodatron, Elmsoft, Alien, DSC...*

Odie: Dem CPC verdanke ich eindeutig, dass ich mich die ganze Zeit während des Studiums selbst finanzieren konnte, da ich immer als Programmierer irgendwo Arbeit hatte. Andererseits denke ich auch, dass ich dieser Programmierarbeit mitunter zu verdanken habe, dass ich relativ lange studiert habe, da ich oft mehr gearbeitet als studiert habe. Nun ja, und ich habe meinen Altersgenossen tatsächlich etwas voraus, da ich ein Verständnis für

Markus Buntru

www

www.odiesoft.de (noch mehr über Odiesoft, alle Downloads, einfach alles!)

SymbOS

Benutzeroberflächen, sogenannte "GUIs", gibt es auf dem CPC Dutzende. Die meisten davon sind BASIC-Erweiterung, die Befehle für das Darstellen von Fenstern und die Steuerung der Maus zur Verfügung stellen. Es gab gute und schlechte Lösungen, aber durchgesetzt hat sich keines der Systeme. Denn kein System war wirklich konsequent genug gemacht, und die oft schlechte Performance und der verschwenderische Speicherverbrauch taten sein übriges. Die meisten Anwendungen auf dem CPC, die ebenfalls "GUI"-Elemente enthielten, griffen so auch lieber auf ein eigenes, spezialisiertes System zu. Und ohnehin herrschte irgendwie immer die Meinung, soetwas sei sowieso nur auf einem 16Bit System wie z.B. dem Amiga sinnvoll.

Ganz anders war dies auf dem C64. Berkeley Softworks (später "GeoWorks") brachte bereits 1987, nur wenige Jahre nach dem Release des C64, GEOS heraus. Es war eine an die damaligen MAC-Oberflächen angelehnte GUI, für die in der folgenden Zeit Dutzende von Anwendungen geschrieben wurden, darunter GeoWrite (wie Word), GeoCalc (wie Excel), GeoFile (wie Access), GeoPaint (wie Paint) usw. GEOS war langsam, aber es funktionierte, und das sogar auf einer 64K-Maschine! Durch die guten Anwendungen, die damals normalerweise nur auf 16Bit-Systemen zu finden waren, war es sofort DAS Betriebssystem für den C64 - zumindest für die Leute, die nicht nur spielen wollten.

Auf dem CPC hat man soetwas nie geschafft. Dabei bietet er viel optimalere Voraussetzungen als der C64. Er ist aufgrund seiner wesentlich moderneren Technik definitiv besser für Anwendungen geeignet, er ist schneller, hat meistens mehr Speicher und bietet eine flexiblere Grafik, auch wenn diese nicht unbedingt so bunt wie beim C64 aussieht. Als ich mir im Jahre 2000 verschiedene GEOS Internet-Seiten ansah, begann ich mich tierisch über diese Tatsache zu ärgern. Es war im Grunde eine Schande für den CPC, daß er hier immer noch von einem System überflügelt wurde, dessen Grundkonzept noch aus den 70ern stammt. Daher kam für mich die

Überlegung, nochmal (genauer gesagt zum vierten Mal nach MesoDOS, CPCBench und CEUS 2.0) einen GUI-Versuch auf dem CPC zu starten.

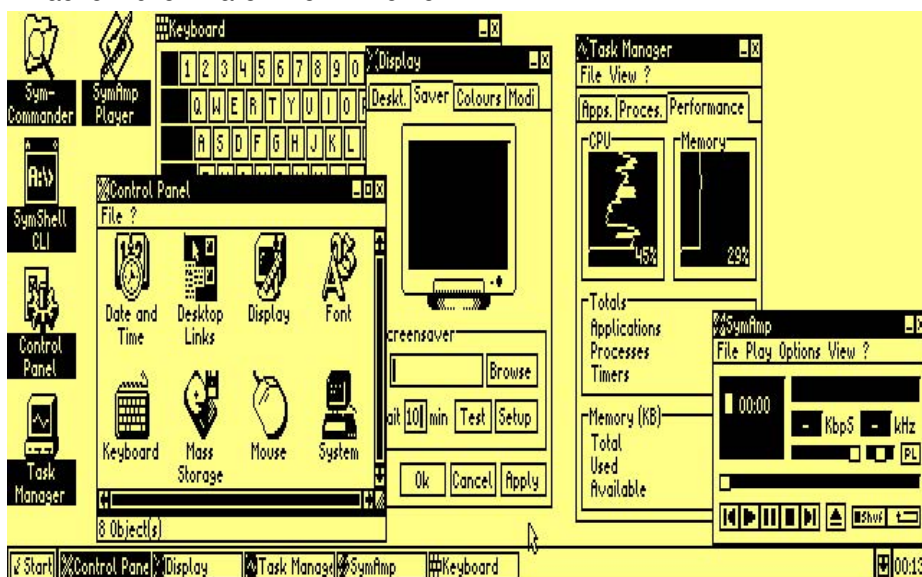
Der Screen-Manager

Die erste Frage war: Was brems eine grafiklastige GUI eigentlich am meisten? Richtig, ausgerechnet die Textausgabe! Sprites plotten, gerade Linien zeichnen und viereckige Flächen füllen waren nie wirklich ein

Beweis erbracht, daß der CPC doch in der Lage ist, eine ziemlich fixe GUI hinzubekommen, und daß es sich lohnen würde, damit weiterzumachen.

Der Screen-Manager war also der erste Schritt. Er stellt die Low-Level-Routinen für die Bildschirmausgabe zur Verfügung. Diese bestehen hauptsächlich aus der Ausgabe von Proportionaltext und Sprites, dem Ausfüllen von Flächen bzw. Linien sowie dem Scrollen und Kopieren von Bildschirm-Bereichen.

Hochoptimierte Bildschirm-Routinen gibt es bereits viele auf dem CPC. Viel seltener sind solche, die Mode-unabhängig sind, pixel- statt byte-genau arbeiten und zusätzlich mit Clipping umgehen können. Für die Mode-Unabhängigkeit wurden jeweils eigene Versionen für jede Funktion geschrieben. Clipping bedeutet, daß die Ausgabe auf einen beliebigen, pixelgenau-



Betrieb in Mode 2: Dank der Lowlevel-Routinen des Screenmanagers läuft SymbOS in jeder Auflösung

Problem für den CPC. Clippingfähige Proportionalchrift dagegen könnte das schon eher sein. Nach dem Schreiben und Testen einer Mode-unabhängigen Proportionaltext-Routine, die vollgestopft mit selbstmodifiziertem Code und aufgelösten Schleifen ist, hatte ich dann quasi für mich den

definierbaren Bereich eingeschränkt werden kann und dann Teile des Sprites bzw. des Textes nicht zu sehen sind. Um das alles trotzdem in einer hohen Geschwindigkeit zu erhalten, quillt der Screen-Manager über vor selbstmodifizierendem Code und Schleifenaufösungen.

Das Ergebnis ist, daß die SymbOS-GUI die erste ihrer Art ist, die eine hochdynamische auf Pixel- statt Byte-Basis laufende Bildschirmausgabe in einer angenehmen Geschwindigkeit bereitstellt.

Der Desktop-Manager

Das Problem bisheriger CPC-GUIs war der Speicherverbrauch und die Unflexibilität. Wurde ein Fenster geöffnet, mußte der komplette Bildschirmbereich dahinter zwischengespeichert werden. Öffnet man also vier Fullsize-Fenster, so werden bereits 64KB nur für das Zwischenspeicher des Hintergrundes verschwendet. Völlig unbekannt war meistens die Möglichkeit, ein Fenster zu verkleinern und den Inhalt zu scrollen. In diesem Falle hätte man den nichtgezeigten Bereich ebenfalls zwischenspeichern müssen.

Der Desktop-Manager von SymbOS ist auf dem CPC der erste seiner Art, der die Oberflächen-Darstellung vollständig Objekt- und nicht Byte-orientiert realisiert. Der Inhalt eines Fensters besteht also aus einer beliebigen Ansammlung von Objekten (auch "Controls" genannt). Dies können z.B. Texte, Symbole oder Bilder sein, aber auch Eingabefelder, Buttons, Check-boxen, Listen oder Regler. Um den Inhalt eines Fensters zu speichern, müssen also nicht die Bytes aus dem Bildschirmspeicher kopiert werden, sondern lediglich die Beschreibung der Objekte. Theoretisch ist SymbOS dadurch in der Lage, selbst auf einem 64K-System dutzende von beliebig großen Fenstern auf einmal zu öffnen.

Der Desktop-Manager ist der komplexeste Bestandteil von SymbOS. Neben Fenstern und Objekten stellt er die Maussteuerung, verschachtelbare

Pulldown-Menüs und die Task- und Iconleiste mit Uhrzeit und dem bekannten "Start"-Button zur Verfügung. Er verfügt über einen eigenen Prozess, der mit der zweithöchsten Priorität direkt hinter dem Kernel-Prozess liegt. Auf diese Weise wird sichergestellt, daß auch auf einem unter Last stehenden System immer eine schnelle und bedienbare Oberfläche zur Verfügung steht. Das Arbeiten mit SymbOS wird dadurch niemals zäh und man bekommt das Gefühl, daß der CPC niemals zu langsam ist.

Der Kernel

Eine GUI macht noch kein Betriebssystem aus. Das wird immer gerne verwechselt. Ein eigenständiges Betriebssystem zeichnet sich unter anderem dadurch aus, daß es selbst den Speicher und die Prozesse, die darin laufen, verwaltet.

Der Kernel und das Multitasking

Die Prozessverwaltung ist vollständig Multitasking-basiert und arbeitet mit einer Kombination aus der preemptiven (Unterbrechung des arbeitenden Prozesses durch das System nach einer festgelegten Zeit) und der kooperativen (wartender Prozeß gibt freiwillig CPU-Zeit ab) Methode. Durch die Kombination beider Methoden ist eine Priorisierung der verschiedenen Prozesse möglich. Nur wenn hochprioritäre Prozesse allesamt freiwillig ihre CPU-Zeit abgeben, erhalten niedrigprioritäre ebenfalls CPU-Zeit. Der Leerlaufprozeß, der ganz am Ende der Kette steht, kann auf diese Weise mit einem Zähler exakt die noch freie CPU-Zeit berechnen.

In der Praxis klappt das Ganze auf



Prozessliste im Taskmanager: Der Kernel erlaubt echten Multitasking-Betrieb von bis zu 32 Prozessen

Der Microkernel ist das Herz von SymbOS. Er regelt alle grundlegenden Kernaufgaben des Betriebssystems. Dies betrifft die Prozess-, die Speicher-, die Banking- und die Message-Verwaltung. In jedem Bereich hält er Besonderheiten bereit, die es in der Form bisher nicht auf dem CPC gegeben hat.

dem CPC erstaunlich gut und man fragt sich, warum man den Z80 im CPC über 20 Jahre lang mit öden Single-Tasks gelangweilt hat. Das Verhalten unter SymbOS ist quasi identisch zu dem aktueller 32Bit-Systeme. Tatsächlich wird nicht viel mehr CPU-Zeit durch die Taskwechsel verbraten, als dies das normale CPC-OS mit der

Interruptverwaltung tut. Timer sind Sonderprozesse, die über keine Priorität verfügen und allesamt genau 50mal pro Sekunde aufgerufen werden. Übrigens verfügt jeder Prozeß über seinen eigenen Stack - anders ist dies in einer Multitasking-Umgebung gar nicht möglich.

Der Kernel und das Speicher-Management

Die Speicherverwaltung teilt das komplette RAM ab der zweiten Bank in einzelne, 256 Byte große Zellen auf und vergibt sie dynamisch. Blöcke bis zu 63KB an einem Stück können reserviert, wieder freigegeben oder nachträglich in ihrer Größe verändert werden.

Die Bankverwaltung regelt die wirklich kniffligen Aufgaben innerhalb des Speicher-managements. Ich war mir längere Zeit nicht sicher, ob das geplante Konzept aufgehen würde, und das war einer der Gründe, warum SymbOS zwischendurch solange auf Eis gelegt wurde. Im September 2004 ist dann schließlich der Durchbruch gelungen. Voraussetzung ist, daß eine Anwendung in einer komplett eingblendeten sekundären 64KB Bank ausgeführt wird und sich keinerlei Gedanken um Banking machen muß. Durch Sprungfunktionen, die das erste 1K jeder Bank belegen, ist ein Wechsel in die primäre Bank möglich, die komplett vom Betriebssystem und vom Bildschirmspeicher belegt wird. SymbOS ermöglicht auf diese Weise die Programmierung von Anwendungen, die bis zu 63KB am Stück lang sein dürfen und dabei selbst völlig ohne Banking arbeiten. Die Speichernutzung wird dadurch fast so einfach wie es unter MS-DOS auf 16Bit-Systemen möglich war, wo ein Segment auch nicht größer als 64K sein durfte! In der Praxis ist es erstaunlich, wie simpel man plötzlich mit einer 8Bit-CPU

einen mindestens halben MB großen Speicher nutzen und verwalten kann. Selbstverständlich darf eine Anwendung auch mehr als 63KB reservieren.

Der Kernel und die Prozess-Messages

Durch die Message-Queue des Kernels wird die Kommunikation der Prozesse untereinander geregelt. Dies ist eine der größten Besonderheiten, die man bei der Programmierung in einer Multitasking-Umgebung beachten muß. Denn anders als bei einem Single-Task-Betriebssystem rufen Applikationen meist nicht durch CALLs komplexere Funktionen des Betriebssystems auf. Statt dessen senden sie Nachrichten an das jeweilige Modul. So wartet z.B. der Desktop-Prozess auf Fenster-Update-Anweisungen der Prozesse, während er selbst z.B. User-Interaktions-Nachrichten an die Prozesse zurücksendet.

Der System-Manager

Alle Dienste des Betriebssystems, die nicht mit der Bildschirm-Verwaltung zu tun haben, werden vom System-Manager abgearbeitet. Er verfügt ebenfalls über einen



Programmstart-Dialog: Der Systemmanager stellt unter anderem Dienste zum Öffnen von Dateien zur Verfügung

eigenen Prozeß, der von der Priorität her unter dem des Desktop-Managers angesiedelt ist. Neben den Diensten übernimmt er das Starten und Beenden von kompletten Anwendungen. Beim Starten kümmert er sich um die

Reservierung von Speicher und dem Initialisieren des Programmes. Dank des Relokators kann eine SymbOS-Anwendung an jeder beliebigen Speicheradresse ausgeführt werden! Nur dadurch ist unbeschränktes Multiprogramming möglich. Wird ein Programm beendet, sorgt der System-Manager dafür, daß alle zugehörigen Prozesse beendet, Speicherbereiche freigegeben und Desktop-Fenster geschlossen werden, ohne daß sich die Anwendung selbst darum kümmern muß. Auf diese Weise ist sogar ein kontrolliertes "Abschießen" einer Anwendung möglich, ohne daß diese davon überhaupt Kenntnis bekommt.

Der Filemanager

Nach dem Desktop-Manager ist der File-Manager das umfangreichste Modul von SymbOS. Zwei Hauptziele wurden bei seiner Entwicklung verfolgt: Das erste war die Unterstützung von beinahe unbegrenzt großen Datenträgern mit einem entsprechend leistungsfähigen Filesystem. Das andere war die Bereitstellung eines möglichst komfortablen Filemanagements für die Applikationen, das handlerbasiert ist und sich stark an den Leistungsmerkmalen der letzten MS-DOS-Versionen orientiert.

Das Timing für den Filemanager hätte nicht besser sein können. Genau zu dem Zeitpunkt, als ich die Arbeit beginnen wollte, brachte Dr. Zed den Prototyp seines CPC-IDE-Interfaces ("SYMBiFACE") heraus. Das Betreiben von Festplatten mit gigantischen Kapazitäten von bis zu 128GB war erstmals direkt mit dem CPC möglich. Aus dem Grunde fiel die Entscheidung leicht, innerhalb SymbOS nicht nur AMSDOS, sondern auch FAT12, FAT16 und FAT32 zu unterstützen. Mit letzterem ist SymbOS theoretisch in der Lage, Datenträger mit einer Kapazität von bis zu 2 Terrabyte zu

verwalten. Erstaunlich dabei ist, dass ein 32Bit-Filesystem den CPC in keiner Weise in die Knie zwingt! Der Z80 handhabt die breitbittigen PC-Datenstrukturen in einer Geschwindigkeit, als wäre er von Anfang an für so etwas konzipiert worden. In der Praxis erreicht er Transferraten von knapp 140KB pro Sekunde.

Der Filemanager besteht aus drei Schichten, von denen zwei treiberbasiert sind. Die LowLevel-Hardware-Schicht beherrscht das Lesen und Schreiben von Sektoren. Momentan gibt es IDE- und FDC-

Lese- und genau einen Schreib-Handler beschränkt. Statt dessen können bis zu 8 Dateien gleichzeitig ohne zusätzlichen Speicherverbrauch geöffnet werden. In jede davon kann gleichzeitig geschrieben und gelesen werden, und das auch noch in beliebig vielen und großen Blöcken.

Die Applikationen

Noch vor Abschluß des File-Managers waren bereits die ersten Applikationen in Arbeit. Dabei handelt es sich allerdings eher noch um Betriebssystem-Bestandteile als

es völlig egal ist, ob es eine Amsdos- oder PC-Diskette ist. Minesweeper ist eine 1:1 Kopie der bekannten Windows-Version. Es hat mit zwei Abenden Entwicklungszeit gezeigt, daß es sehr schnell sein kann, Anwendungen für SymbOS zu schreiben, weil das Betriebssystem einem bis auf die Logik komplett alles abnimmt.

Abschluss

SymbOS bringt 32Bit-Feeling auf den CPC und katapultiert ihn ins neue Jahrtausend. Es scheint irgendwie geklappt zu haben, ein Windows-Betriebssystem mit einer überraschend akzeptablen Geschwindigkeit basierend auf einem Multitasking-Kernel und einem 32Bit-Filesystem auf dem CPC zu realisieren. Der CPC kommt einem wie verwandelt vor, da man einen voll dynamischen Windows-Desktop und ein solch selbstverständliches Multi-programming bisher nicht gekannt hatte. Mit SymbOS scheint der CPC vom Leistungsfeeling her sogar 16Bit-GUI-Betriebssysteme aus den 80ern zu überholen. Tatsächlich wirken die Workbench auf einem Ur-Amiga oder GEM auf einem Atari ST primitiver als SymbOS und sind von der Geschwindigkeit her auf dem selben Niveau. Es ist ein bisschen bedauerlich, daß dies schon vor 20 Jahren möglich gewesen wäre.

Das Ziel war es, ein Speicher- und Prozeß-Management zu haben, das den CPC bis an die Grenzen ausreizt, und gleichzeitig MS Windows so exakt wie möglich nachzubilden. Eine alternative Oberfläche zu entwickeln wäre vielleicht kreativer gewesen, würde aber bei langem nicht den Überraschungseffekt auslösen, den ein WinXX-Klon auf einem 8Bit-System hervorruft.

Der Traum ist es, ein neues lebendiges System zu schaffen, das



Directory-Ansicht in der Shell: Der Filemanager verwaltet neben Amsdos auch alle FAT-Formate bis 128GB

Treiber. Die MidLevel-Filesystem-Schicht sorgt für die generelle Verwaltung der Datenstruktur auf dem Massenspeicher. Aktuell existieren AMSDOS- und FAT-Treiber. Unterverzeichnisse und Timestamps stehen dabei sogar in beiden Filesystemen zur Verfügung, ohne AMSDOS-Datenträger inkompatibel zu machen.

Die HighLevel-Filehandling-Schicht letztendlich stellt die Funktionen zum Arbeiten mit Dateien und Verzeichnissen zur Verfügung. Die Möglichkeiten und der Umfang sind fast identisch mit denen von MS-DOS. Anders als bei AMSDOS ist man also nicht auf genau einen

um Anwendungen. Um den Überraschungseffekt zu steigern, sind der Task-Manager und die System-Steuerung soweit wie möglich Kopien ihrer MS Windows Varianten.

Die ersten "echten" Applikationen stellen der SymCommander, MineSweeper und der DiskDeDumper dar. Mit dem SymCommander erscheint zum ersten Mal ein Norton Commander Klon auf dem CPC. Es ist schon ein komisches Gefühl, damit wie auf dem PC durch die Verzeichnisse einer mehrere Gigabyte großen Platte zu surfen. Oder die Files auf einer Diskette zu managen, bei der

einem wirklich zum Benutzen verleitet. Anwendungen wie ein Internet Browser, ein Instant Messenger oder Äquivalente zu Word und Excel sind hierfür

notwendig. Sie sind allesamt realisierbar, man kann nur hoffen, daß auch die Zeit dafür da ist.

Prodatron / Symbiosis

www

www.symbos.de (SymbOS-Download, SymbOS-Doku... einfach alles was man auf der offiziellen Seite so erwartet)

Power Drift

Warum spiele ich eigentlich keine Autorennspiele am CPC? Ach ja, da seit Jahren Schumi ununterbrochen in seinem roten Flitzer gewinnt, interessiert mich das ganze drumherum irgendwie nicht mehr. Zum Glück war es dieses Jahr etwas anders. Also gibt es für mich keinen Grund mehr das zu boykottieren. Also CPC an, Tape reingeworfen und... warten!

Power Drift liegt in meinem mono Kassettenrekorder, der nun ganz gemächlich das Programm lädt. Zeit für einen Kaffee. Als ich wieder zurückkomme, strahlt mir schon das Titelbild entgegen, und siehe da, dreidimensional dreht sich der Vektorschritzug Power Drift auf meinem Monitor. Das Game eine Art Strandbuggy- bzw. Kart-Rennen ist im Original in Regie von Segas Spitzenmann Yu Suzuki Entstanden. Dieser hat z.B. auch Outrun oder G-Loc auf der leistungsstarken Model 2 Hardware entwickelt. Größter Clou von Segas Model 2 war das Zoomen von Objekten und die Berechnung im Raum, sie stand auch Pate bzw. Vater für den Sega Saturn. Also ist die Aufgabe nicht gerade einfach, die dem kleinen CPC mit diesem Spiel gestellt wurde. War doch Outrun eher eine Kaffefahrt auf den 8Bit Rechnern.

Joystick an den CPC gestöpselt und auch ausgewählt. „Null“ gedrückt und wir landen im Auswahl Menü des Fahrers und der Strecke. 12 knallharte Jungs und Mädels stehen zur Auswahl, und ich greif mir mal gleich den coolen mit der Sonnenbrille. Leider hat während des Rennens keiner seine Vor- und Nachteile, zumindest keine merklichen. Als Strecke, von denen es fünf zur Auswahl gibt, lacht mir das Icon mit der Palme entgegen. Die passt zu meinem Fahrer. Seite umdrehen und noch mal gute 3

Minuten warten.

Dann stehe ich vor dem Start. Die Ampel wird grün und ich quetsche meinen Joystick nach vorne. Kurz das Feuer gedrückt und der zweite Gang drückt mich weiter in den



Sessel. Überrascht von dem schnellen und dem ziemlich flüssigen Ablauf ergreift mich der Ehrgeiz. In der Kurve versuche ich zu überholen doch da trifft mich der Schlag. Rotierend fliege ich kreuz und quer über die Strecke und werde von drei anderen überholt. Das Ranking wird immer am oberen



Knapp hinter der Spitze...

Bildschirmrand eingeblendet, so manifestieren sich bald starke Abneigungen gegen bestimmte Fahrer. Schnell auf einen Zettel geschrieben, „den Irokesen von der Bahn werfen“. 2 Runden später

sehe ich auch schon die ersten drei Fahrer im Ziel erscheinen. Sie werden links eingeblendet und sorgen nicht wirklich für Frohsinn bei mir. Die hohe Detailvielfalt der Landschaft und der Renner macht Spass. Doch für mich ist das Rennen aus.



Anscheinend weiß mein CPC, dass ich zuviel Kaffee trinke, und fragt nett nach, ob ich dieses Rennen wiederholen will. Schön, denn auf Ladezeiten hab ich gerade keine Lust.

Noch mal angefangen und los. Dieses mal werde ich zweiter und komme in die nächste Runde. An die Steuerung kann man sich leicht gewöhnen. Die Kollisionsabfrage ist fair und die Objekte welche einem aus dem Horizont entgegen kommen haben fünf Größenunterschiede, welche ein wirklich gutes Raumgefühl entstehen lassen.

Weiter und weiter fahre ich Runde um Runde und vergesse, dass ich eigentlich noch weg wollte. Zufrieden schalte ich dann nach einer Stunde den CPC aus und habe endlich mal wieder ein gutes Gefühl, wenn ich ein Rennen sehe, in dem Schumi nicht gewinnt.

Im Fazit ein wirklich gutes Spiel, schnell, feine Grafik und fair.

JS/Tolkin

Das RGB Kabel

Dieser Artikel ist für Bastelanfänger gedacht, so wie ich auch einer bin. Vielleicht hast du deinen CPC an einem Grün Monitor angeschlossen und würdest auch lieber gern einen Farbmonitor oder Fernseher mit SCART-Eingang benutzen, den du noch rumstehen hast.

Dazu benötigt man dann ein SCART Kabel. Dieses besteht aus nur wenigen Teilen, die auch nicht sehr teuer sind.

Materialien

- 1x Scart Stecker (Conrad Artikel-Nr.: 741787 - LN)
- 1x 6 poliger DIN Stecker (Conrad Artikel-Nr.: 737453 - LN)
- 1x Kabel 6 Adern plus Abschirmung

Werkzeug

- LötKolben
- LötZinn
- Flussmittel
- Litze (für die Brücken)

Die beiden Stecker und das Kabel müssen nach folgendem Plan mit einander verlötet werden.

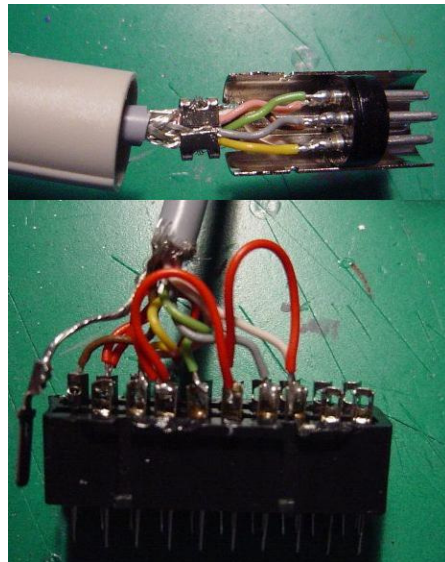
Der Pin 5 von dem Din Stecker muss mit 5 Pins am SCART Stecker verbunden werden. Auf dem nebenstehenden Foto sind diese Verbindungen als rote Brücken zu erkennen.

Hier noch ein paar Bilder während des Bastelns. Die Abschirmung des Kabels wird beim DIN Stecker hinten mit den beiden Metalllaschen eingequetscht und so mit der Abschirmung des Steckers verbunden. Beim Scart Stecker wird Pin 21 mit der Abschirmung verlötet (siehe Bild ganz links).

Ein wirklich einfaches Vorhaben, was jedem, der einen LötKolben in der Hand halten kann ohne sich oder anderes dabei zu verbrennen gelingen sollte ;)

Wenn man zusätzlich den Ton über

seinen Monitor / Fernseher laufen lassen will, so kann man den 3,5 Klinkenausgang am CPC nutzen (aus der Videobuchse des CPC kommt leider kein Audio raus). Dazu benötigt man dann noch zusätzlich ein 3,5 Klinkensteckerkabel, dessen



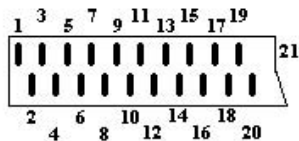
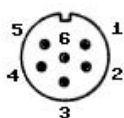
Ende man in den SCART Stecker einlöten muss und zwar an die Pins 2 und 6. Die Masse muss dann an Pin 4 des Scart Steckers gelötet werden.

Viel Spaß!

Erik Timendorfer

CPC - SCART Kabel

*** Stecker von hinten gesehen ***



1	Rot	15
2	Grün	11
3	Blau	7
4	Synchronisation	16
5	Masse	5/9/13/14/18
6	Luminanz	20

Sicherheits halber ein Hinweis: ein jeder bastelt auf eigenes Risiko. CPC forever übernimmt für eure Versuche keinerlei Haftung. Weder für beim Lötten gegrillte Haustiere noch für gebratene Hardware und schon gar nicht für Brandblasen an den frisch manikürten Pfoten. Wer sich also unsicher ist: Finger weg oder weitere Ratschläge von Experten (z. B. unter www.cpc-forever.com) einholen.



3,5" Laufwerk und Seitenschalter

Das 3"-Diskettenformat der CPCs hat sich ja nun nicht gerade zum Standard entwickelt. Was liegt also näher, als auch am CPC im gewohnten 3,5"-Format arbeiten zu wollen?

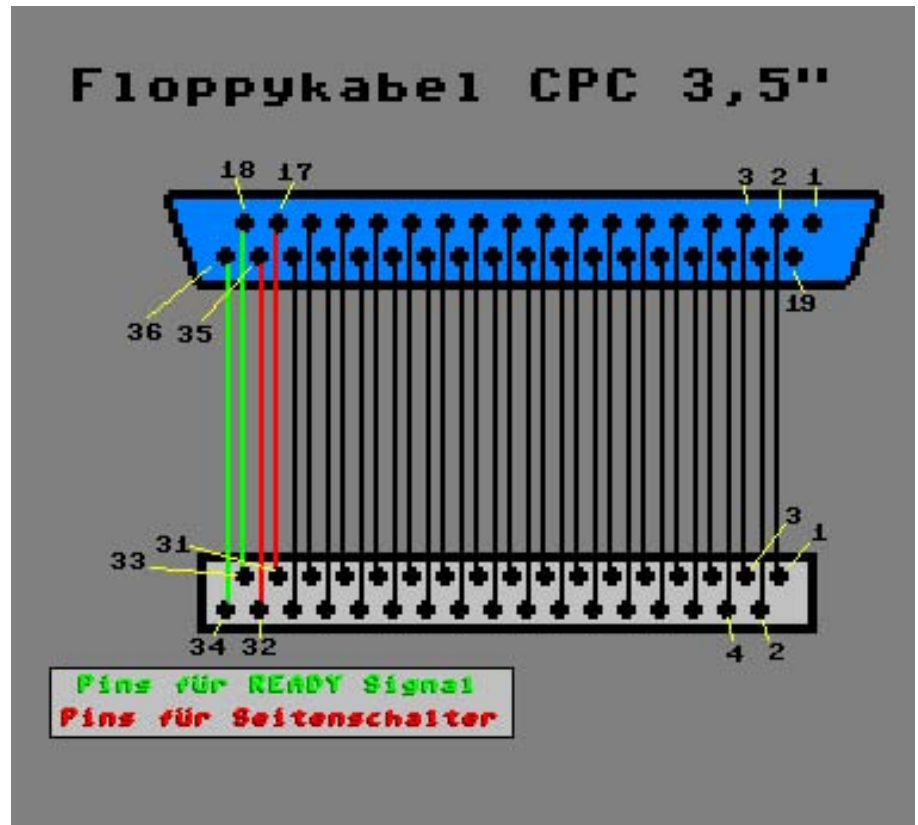
Um ein 3,5" Laufwerk als 2.Laufwerk auf einfache Weise an seinem CPC betreiben zu können benötigt man zuerst ein entsprechendes Kabel um es anschließen zu können. Unter AMSDOS können dann 3,5"-DD Disketten mit 180KB / 40 Spuren pro Seite beschrieben werden. Es passt also genau der Inhalt einer 3" Diskettenseite auch auf eine 3,5" Diskettenseite. Mit dem Seitenschalter wird zwischen der 1. und der 2. Seite der Diskette umgeschaltet, sodass dann 360KB pro Disk zur Verfügung stehen. Unter anderen Betriebssystemen (wie z. B. Parados) ist es möglich die 3,5" Disketten mit 720KB doppelseitig zu beschreiben (also ohne Seiten-Umschalter). Das wird hier nicht beschrieben.

Dazu benötigt man:

- 1x Centronics-Stecker 36 polig (Conrad Artikel-Nr.: 742031 – 62)
- 1x Flachbandkabel 34 polig mit Floppy Stecker (? ArtNr. - das hatte ich hier rumfliegen ;))

Nachdem ich zuvor das SCART Kabel gebastelt hatte, dachte ich, dass ich auch hier jedes Kabel mit einem Pin verlöten müsste, wovor mir bei 34 Stück schon ein wenig graute. Aber netterweise wird ein solcher Centronics-Stecker in der Quetsch/Schneid Technik an das Kabel gebracht, sodass man nur das Flachbandkabel in den Stecker legen und dann den Deckel des Steckers mit viel Kraft zudrücken muss. Dadurch wird das Kabel dann gegen die kleinen Schneiden gedrückt, welche sich durch die Isolierung schneiden und den Kontakt zu den Leitungen herstellen.

Hier nun der Schaltplan:



Die Stecker sind von hinten, also von der Lötseite gesehen

Bei dem Centronicsstecker muss man nur darauf achten, dass die beiden Pins 1 + 19 (also die ganz rechts) leer bleiben, dann müsste alles richtig verkabelt sein. Zuquetschen und fertig. Ich habe das mit einer Zange gemacht, und mir dabei leicht die Metallumhüllung verbogen. Es war aber nicht zu stark, so dass der Stecker trotzdem gut passt.

In dem Schaltplan zu erkennen sind die grün und die rot eingezeichneten Leitungen des Kabels. Die Grünen sollten einfach kurzgeschlossen werden, um so ein READY Signal zu erzeugen. Manchen Laufwerke erzeugen selbst so eines, andere anscheinend nicht, hat man diese beiden Leitungen kurzgeschlossen sollte es aber auf jeden Fall gehen. Die Roten sollten mit einem An/Aus-Schalter versehen werden, da man

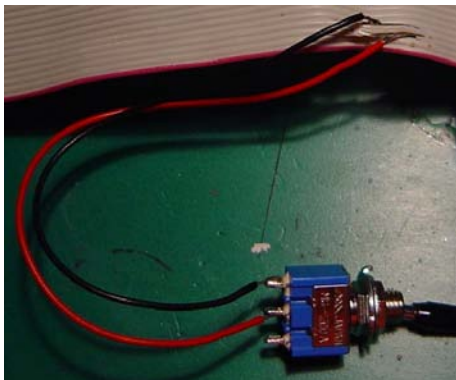
hier die Seiten der 3,5" Diskette umschalten kann.

Der CPC schreibt ja auf seine 3" Disketten nur mit einem Lese-Schreibkopf. Um die andere Seite

nutzen zu können muss die Diskette umgedreht werden. Schließt man nun ein 3,5" Laufwerk an, so behandelt der CPC dieses genauso, d.h. er schreibt nur auf Seite 1 der Diskette. Nun ist es leider nicht möglich die Diskette einfach umzudrehen, wie man es mit der 3" Diskette machen würde. Aber mit dem Schalter kann man den Schreibkopf des Laufwerks, der gerade aktiv ist wechseln, sodass man damit auch die Seite 2 der Diskette beschreiben kann.



Hier sind die Leitungen abisoliert.



Und hier der Schalter dazu

Damit das Ganze etwas netter aussieht wollte ich mir ein Gehäuse für den Schalter und den Stecker basteln. Da eine MiniDV Kassette auf dem Tisch lag (ich werkel auch gerade an dem letzten Urlaubsfilm) entschloss ich mich eine der Kassettenhüllen zu nehmen.



Hier etwas zurechtgestutzt mit eingeschraubten Schalter. Das ganze wird nun noch verklebt und angesprüht.

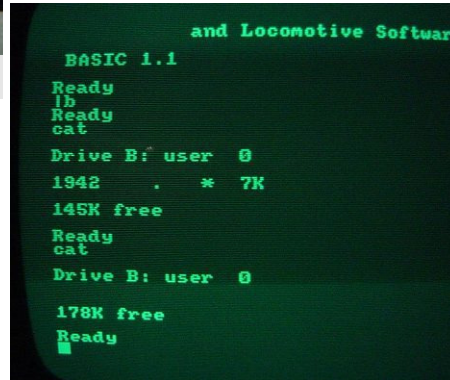


Im Nachhinein wäre es sicher einfacher gewesen, einfach etwas Flachbandkabel aus dem Centronicstecker herausragen zu lassen, um an diesem Stück sämtliche Kurzschlüsse und Schalter anzubringen. Daran hatte ich aber nicht gedacht, als ich den Stecker angebracht habe, und da das Kabel ja nun schon funktionierte, wollte ich es nicht

wieder auseinandernehmen. Wer weiß, ob die Schneiden noch mal durch die Isolierung kämen... :)

Quellen:

- <http://www.cpc-forever.com>
- <http://www.c64forum.de/>
- <http://www.schneider-cpc.de/>
- <http://www.zock.com/8-Bit/>



Hier wurde nur einmal zwischen den CATs der Schalter umgelegt, die Diskette wurde nicht gewechselt.

Erik Timendorfer

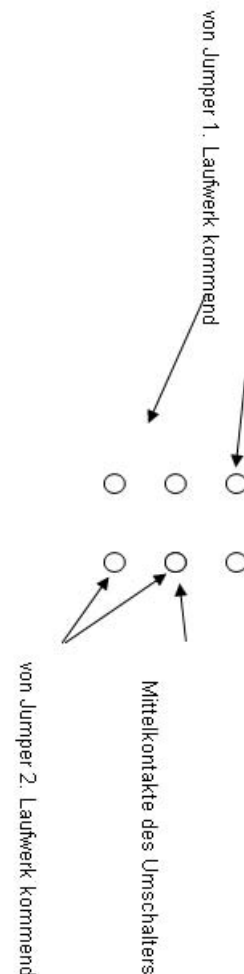
Hardware/Bastelecke

Laufwerksumschaltung einmal anders

Wer am CPC mehrere B-Laufwerke gleichzeitig angeschossen haben möchte hat auch ein Problem. Die passende Lösung folgt aber auf dem Fusse.

Als B-Laufwerke am CPC 6128 betreibe ich ein 3 Zoll Hitachi Laufwerk (180K) und ein 3,5 Zoll Teac Laufwerk (720K). Nervig ist, dass ich immer den Laufwerkstecker am Laufwerk umstecken muß, je nach dem, welches Laufwerk ich betreibe. Das 3 Zoll Laufwerk wird dabei natürlich nur unter AMSDOS und das 3,5 Zoll Laufwerk unter AMSDOS, X-DDOS, X-VDOS oder PARADOS betrieben. Meine Laufwerk Umschaltung funktioniert wie folgt: Beide Laufwerke hängen am gleichen Kabel vom CPC kommend. Die Laufwerkjumper bei beiden sind aber alle offen. Je nach

dem welches Laufwerk nun betrieben werden soll, wird ein Kippschalter umgelegt, der wechselseitig die DRIVE 1 (B) Jumper auf der Platine kurzschließt.



So können beide Laufwerke immer angeschossen bleiben und bei Bedarf wird umgeschaltet.

Xesrjb

Sicherheithalber ein Hinweis: ein jeder bastelt auf eigenes Risiko. CPC forever übernimmt für eure Versuche keinerlei Haftung. Weder für beim Löten gegrillte Haustiere noch für gebratene Hardware und schon gar nicht für Brandblasen an den frisch manikürten Pfoten. Wer sich also unsicher ist: Finger weg oder weitere Ratschläge von Experten (z. B. unter www.cpc-forever.com) einholen.

Der CPCnG ist tot, es lebe der CPCnG

Der Turbo Dino kommt!

**576KB Ram; 256KB Rom; 12MHz Z80; 6 GB IDE-Device; VGA und PS2
...und trotzdem 100% CPC kompatibel! Hier ein erster Erfahrungsbericht
vom neuen CPC TREX made by Tobiflex.**

Viele Jahre werkelte ein internationales Team an einem neuen CPC Modell, welches bei voller CPC Kompatibilität aktuelle Technologien beinhaltet und den CPC zukunftsfähig machen soll. Trotz eines guten Konzeptes gab dann irgendwann im Jahre 2004 der Leiter des Teams endgültig zu, daß das Projekt gescheitert war, und vorerst nicht am CPCnG weiterentwickelt wird, bis sich neue Leute finden, die dies fortsetzen. Daß dies nie passieren würde, darin waren sich die meisten leider sicher.

Doch völlig unabhängig davon entwickelte seit einiger Zeit Tobiflex, ein Meister der Hardware-"Programmierung", einen Core für den C-ONE. Der C-ONE war als Nachfolger des C-64 geplant und sollte diesen mit der sogenannten FPGA-Technologie vollständig nachbilden und ergänzen. Dabei handelt es sich um Chips mit frei konfigurierbarer Logik-Gatter-Verknüpfung. Einfacher gesagt kann man sich mittels FPGA seinen eigenen Microprozessor und die Hardware drumherum zusammenbauen, ohne auf Software-basierte Emulation zurückgreifen zu müssen, und ohne selber eigene Chips herstellen zu müssen.

Der C-ONE ist zwar von der Hardware her auf den C-64 spezialisiert: z.B. enthält er ein oder zwei SIDs und einen MOS 65816 ("SuperCPU"). Jedoch ist es trotzdem möglich, fast jeden anderen 8Bit basierten Computer nachzubilden. Die Konstrukteurin des C-ONE arbeitete an einem C-64 Core (Core ist die Bezeichnung für die FPGA-Konfiguration, die einen bestimmten Computer nachbildet,

wobei unter "Konfiguration" ein verdammt komplexes Konstrukt zu verstehen ist), und Tobiflex entschied sich, einen alternative CPC-Core zu kreieren.

Tatsächlich schaffte es Tobiflex, den ersten überhaupt vollständig lauffähigen Core für den C-ONE zu entwickeln - den CPC464 Core - und ihn auf der XzentiX 2004 zu präsentieren. Eine immer noch riesengroße C64-Szene ärgert sich bis heute, daß es keinen finalen C-64 Core für "ihren" C-ONE gibt, während dessen schon das C-ONE Hauptmenü auf dem Amstrad CPC basiert, denn auch das hat Tobiflex übernommen. Mittlerweise gibt es gleich mehrere CPC-Cores, wobei der "TurboCPC" der perfektionierteste ist.

Der T-Rex

Die Zukunft des C-One ist ungewiß, denn er ist eigentlich zu teuer und zu unhandlich, um für einen 8Bit-Computer-Ersatz in Frage zu kommen. Mitte 2005 fand Tobiflex dann eine vielversprechende Alternative. Den "T-Rex" von der Firma "TerAsic", dessen Herz ein "Cyclone"-FPGA von Altera ist.

Der T-Rex ist ursprünglich als Entwicklungs-Board für Spezialisten gedacht, die neue Chips und Hardware entwickeln möchten. Tatsächlich erfüllt er aber alle Ansprüche an einen vollständigen CPC-Clone. Er ersetzt den C-One, verzichtet auf die C64-spezifischen Komponenten und stellt genau die vom CPC benötigte Hardware zur Verfügung und sogar einiges mehr. Es gibt grundsätzlich nichts, was der

CPC braucht, und was der T-Rex nicht bietet.

Mittlerweile hat Tobiflex den T-Rex zu einem fast ausgereiften CPC-Clone gemacht, der zudem noch wesentlich mehr als das Original zu bieten hat. Doch dazu später. Der T-Rex ist ein professionelles Produkt, was sich wunderbar einfach und schnell über PayPal ordern und bezahlen läßt. Der Versand dauert etwa eine Woche.



Das T-Rex Board mit nur 15x15cm läßt fast keine Wünsche bei den Anschlüssen offen.

Zunächst einmal darf man nicht vergessen, daß der T-Rex wie schon gesagt ein Entwickler-Board ist. Der T-Rex ist eine kleine nackte Platine ohne Gehäuse, die über fast alle nötigen Anschlüsse und Kabel verfügt und fast, aber nicht ganz, dem Prinzip "Anmachen und Loslegen" unterliegt. Wer sich den T-Rex aus den USA (da kommt er her) bestellt, sollte direkt zusätzlich aus heimischen Läden folgendes besorgen oder schon haben:

- 7,5 oder 9Volt Schalt-Netzteil; oder ein Stromstecker Adapter USA-> Europa
- PS/2 Y-Kabel, damit Maus und Tastatur gleichzeitig an die PS/2 Buchse angeschlossen werden können. Achtung, manchmal muß man Maus und Tastatur andersherum an das Kabel anschließen, als es die Beschreibung vorgibt. Wenn es also nicht klappt einfach umgekehrt ausprobieren.
- PS/2-Keybaord, Maus, CRT bzw. TFT (muß 50Hz können!) und

Audio-Anlage

- CF-Speicherkarte; am besten eine "kleine" (128MB oder 256MB) Standard-Karte von SanDisc.

Achtung: mit Highspeed-Karten scheint der T-REX nicht klar zu kommen!

- Windows PC, um die Konfigurations-Software zu betreiben

Installation und Handhabung

Packt man den T-Rex aus, so kommt eine kleine gut 15x15cm große Platine zum Vorschein. Alle Anschlüsse befinden sich praktischerweise an einer Seite, der CF-Card-Slot an der angrenzenden Seite. Außerdem findet man noch den Ein-/Ausknopf, den "Run/Prog"-Schalter, 8 Tastschalter, 4 Dip-Schalter, 8 Leds, ein 4x7 Segment Hexadezimal Display und weitere Status-Leds.

Wie gesagt ist das Board hauptsächlich für Entwickler vorgesehen. Zwar sind Löcher und Schrauben für den Einbau in ein Gehäuse vorhanden, jedoch ist noch nicht ganz klar, was das denn für ein Gehäuse sein soll. Das Erreichen bzw. Sichten von bestimmten Schaltern und Leds wird dann auch nicht mehr so ohne weiteres möglich sein. Für Bastler stellt das allerdings sicherlich eine willkommene Herausforderung dar. Es kursieren bereits mehrere mehr oder minder verrückte Ideen, wo man den T-Rex überall einbauen kann.

Einzig richtiges Manko am T-REX ist vielleicht der CF-Card-Slot. Zum einen besitzt er keinen Auswurf-Hebel, zum anderen sitzt er sehr weit auf dem Board drauf, was das Herausnehmen der CF-Karte erschwert. Das dürfte allerdings nur stören, wenn man häufig Daten mit dem PC tauschen will. Kopiert man einfach die gesamte CPC-Software-

Sammlung auf die Karte, wird man diese nicht oft tauschen müssen. Immerhin passen bereits auf eine 128MB Karte knapp 700 (!) DSK-Files!



Komfortabler Disketten-Wechsel ohne Mechanik. Alle DSK werden auf der CF-Card abgelegt.

Für die Installation und Konfiguration des T-Rex werden zwei Software-Pakete benötigt. Das erste stammt vom FPGA-Hersteller Altera und wird hauptsächlich für das Konfigurieren des FPGAs benötigt, also das Laden des Cores (z.B. des CPC Cores) in den FPGA. Das zweite stammt vom Board-Hersteller TerAsic und dient vor allen Dingen zum Beschreiben des Flash-Speichers.

Ich möchte jetzt nicht weiter auf Details zur Installation eingehen. Nach kleinen anfänglichen Schwierigkeiten, die eher auf meine Ungeduld zurückzuführen waren, entpuppten sich die Installations-Prozeduren als recht einfach und schnell zu handhaben. Ohnehin lädt Tobiflex CPC Core mittlerweile die CPC spezifischen Konfigurationen wie ROMs und DSKs von der CF-Karte, so daß man nur sehr selten das Board selber mit der oben genannten Software anfassen muß.

Der CPC im König der Saurier

Nun kommen wir endlich zum eigentlich Interessanten. Wie läuft er denn jetzt, der neue CPC? Wer den C-One und Tobiflex Core darauf kennt, wird gar nicht so viele

Überraschungen erleben. Der CPC ist nahezu perfekt nachgebaut worden, von der Grafik, dem Sound, dem Timing usw. Fast alle Demos laufen, Spiele und Anwendungen sowieso. Das Timing und der im T-Rex vorhandene "CRTIC" sind perfekt genug, um auch so gut wie alle Hardware-Tricks zu ermöglichen. Der T-Rex kann sich also problemlos mit den Top-Emulatoren wie WinApe, Caprice oder WinCPC messen.

Gibt es trotzdem noch Probleme, so kann man sicher sein, daß auch diese bald gelöst sein werden. Tobiflex arbeitet derzeit mit absolut rekordverdächtigem Tempo am T-Rex und bringt gleich mehrmals pro Woche neue Updates heraus. Und das, obwohl der T-Rex bereits jetzt fast ein 100%ig einsatzfähiger CPC ist.

Besonders schön ist die Tatsache, daß man zwar quasi einen echten Hardware-CPC vor sich stehen hat, man sich die Peripherie jedoch aus aktuellen Standard-Komponenten zusammenstellen kann. Ein 19" TFT Display, eine moderne PS/2 Tastatur und eine schnurlose optische Maus - so sieht z.B. mein neuer CPC-Arbeitsplatz aus! :-)) Einzig ein Joystick fehlt derzeit noch, hier steht aber zumindest eine Simulation über die Tastatur bereits auf Tobiflex Todo-Liste.

Turbo Mode und SYMBiFACE II Integration

So manchen werden aber vielleicht erst die erweiterten Fähigkeiten des T-Rex reizen. Neben der Tatsache, daß der T-Rex einen CPC mit voll ausgebautem Ram (576KB) und 256KB großem Rom enthält, bietet er außerdem eine fast vollständige SYMBiFACE II Implementation und einen Turbo Modus, in dem der Z80 mit dreifacher Geschwindigkeit läuft, nämlich mit vollen 12MHz.

Das Verblüffende am Turbo Mode ist seine hohe verbleibende Kompatibilität zu bestehender Software. Da Interrupts und Framefly Impulse weiterhin genau wie im Normal-Modus getimet sind, laufen tatsächlich sogar Demos mit Hardware-Tricks im Turbo Mode. Der "Constellation" Part von der "Digital Orgasm" Demo kommt mit dem Screen Splitting im Turbo Modus nicht durcheinander, plottet die Vektor-Bälle aber mit 3facher Geschwindigkeit! Überhaupt arbeitet es sich im Turbo Mode oft sehr angenehm auf dem CPC.



Sogar Demos laufen im 12MHz (!) Turbomode: Digital Orgasm mit dreimal soviel 3D-Bällen.

Die SYMBiFACE II Implementation enthält die IDE- und die PS2-Maus-Ansteuerung. Eine Echtzeit-Uhr ist auf dem T-Rex nicht vorhanden, deshalb wäre hier höchstens eine Dummy-RTC möglich. Wer die Wartezeit für das SYMBiFACE II überbrücken möchte, ist mit dem T-Rex aber bestens bedient.

Die neue SymbOS Workstation

Derzeit noch unterstützt SymbOS (Beta 1) als einzige Software die neuen SYMBiFACE II Möglichkeiten des T-Rex. Aber hier können sie bereits vollständig ausgereizt werden. Die Handhabung der Maus läßt sich nicht mehr von der auf einem PC unterscheiden. Es ist erstaunlich, wie perfekt flüssig sich eine echte Proportional-Maus am CPC verhält!

SymbOS erkennt bereits beim

Booten, ob sich der T-Rex im 12MHz Turbo-Mode befindet. In beiden Modi zeigt der Task-Manager daher jeweils die korrekte CPU-Nutzung an, und tatsächlich ist diese im Turbo Mode genau auf dem Drittel des Normal Mode. Trotzdem kann man natürlich auch während des Betriebes den Modus wechseln, z.B. um die Auswirkungen direkt vergleichen zu können. Im Turbo Mode ist der Bildschirmaufbau in SymbOS so schnell, daß man in den meisten Fällen keine Verzögerungen mehr feststellt.

Mit Hilfe des SymCommanders, der SymShell oder einfach der Fileselect-Box kann man mühelos die Verzeichnisse der CF-Karte durchstöbern, um dann in einer bisher nicht gekannten Geschwindigkeit Programme und Dateien zu laden. Alles in allem ist der T-Rex die ideale Plattform, um mit SymbOS bereits jetzt schon echtes Workstation-Feeling zu bekommen. Bleibt zu hoffen, daß dies mittels SYMBiFACE II auch bald auf dem echten CPC Wirklichkeit wird.

Fazit

Tobiflex hat mit dem CPC-Core für den T-Rex eine wahre Meisterleistung vollbracht. Der CPC ist nicht nur perfekt nachgebaut worden, auch verwandelt er sich mittels Turbo-Mode und Hardware-Erweiterungen wie der SYMBiFACE II Implementation zu einer interessanten Power-Alternative zum Original, ohne diesem auf irgendeine Weise untreu zu werden. Die Nutzbarkeit aktueller Peripherie und der Einsatz der für CPC-Verhältnisse fast unendlich großen CF-Karten erhöhen den Komfort beträchtlich, wodurch das Arbeiten mit dem CPC TREX verdammt viel Spaß macht. Man hat nicht wie bei einem Emulator das Gefühl, zu pfuschen, noch muß man Angst haben, das gute alte und echte

Original zu verschleifen.

Der CPC TREX ist jedem zu empfehlen, der sich nicht mit Emulatoren zufrieden gibt, sich gerne noch einmal einen CPC mit völlig neuen Möglichkeiten auf den Schreibtisch stellen möchte, dabei den Komfort moderner Peripherie nutzen will und trotzdem nicht auf 100%ige Kompatibilität verzichten möchte.

Zum Abschluß nochmal eine Zusammenfassung der Leistungsdaten:

- komplette CPC 6128 Standard-Hardware
- Z80 CPU im 4MHz (normal) oder 12MHz (Turbo) Modus
- nahezu 100%ige Kompatibilität
- 576KB RAM (dk'tronics kompatibel; kann mittels MMU noch auf 8MB [!] erweitert werden)
- 256KB ROM (16x16KB, frei belegbar)
- VGA-Anschluß (CRT- oder TFT-Monitor muß 50Hz verarbeiten können!)
- PS/2 Tastatur Anschluß
- PS/2 Maus Anschluß (SYMBiFACE II kompatibel; muß über Y-Kabel zusammen mit der Tastatur angeschlossen werden)
- CF-Karten-Slot als SYMBiFACE IDE-Laufwerk ansprechbar (Standard 128MB oder 256MB OHNE Highspeed vom Hersteller "SanDisc" wird empfohlen; nicht alle CF-Karten funktionieren im T-Rex)
- Audio-Ausgang
- 2 x FDC-Laufwerks-Emulation, DSK werden von der CF-Card geladen
- Serieller RS232-Anschluß
- USB-Anschluß (für FPGA- und Flash-Updates vom PC)

Prodatron / SymbiosiS

www

http://www.terasic.com/english/fpga_01.htm (die offizielle Seite zum Board)

Reisende im Wind

160x200 Bildpunkte sind bekanntlich ja nicht viel. 16 Farben sicher auch nicht. Nimmt man nun ein Grafikadventure und bastelt um den gezeigten Spielverlauf ein paar Leisten und Fenster so bleibt fast nix mehr für Grafik über! Denkste!

Jeder der sich mehr oder minder mit Comics beschäftigt, nicht Micky Maus oder diverse Marvelblättchen, sondern anspruchsvolle Geschichten wie z.B. „Der Sandmann“ kennt die Kunst die hinter dem Gezeichneten steht. Die Linienführung die soviel mehr ausdrückt, als in ein paar Sprechblasen geschrieben stehen kann. Jedes einzelne Bild für sich ein kleines Kunstwerk. Comicleser wissen, das einige der besten Comics nicht aus den USA kommen sondern aus Frankreich. Da der CPC in Frankreich ähnlich stark vertreten war wie bei uns der C64, ist es nicht verwunderlich, dass sich hier auch Infogrames mit dem Medium Comic beschäftigt hat. Und zwar zur damaligen Zeit mit diesem Spiel zuerst auf dem CPC später dann auch auf anderen Systemen.



Man nehme einen mit vielen Preisen ausgezeichneten Zeichner und seinen Comic, ein bisschen liebe zum Detail und eine gehörige Portion Mut für etwas andere Spielthemas und man hat das Spiel „Les Passengers du Vent“. Dieses Abenteuer ist vorneweg nicht jedermanns Sache, entweder man liebt es oder man kann damit überhaupt nichts anfangen. Richtet sich die melancholische Geschichte doch eher an Erwachsene.

Grundlage der Geschichte ist das 18. Jahrhundert während der Zeit des Sklavenhandels. Isa eine Französische Abenteurerin, in Wirklichkeit eine Komtesse, will

nach Südafrika und lernt auf der Reise den Matrosen Hoel kennen, der ungerechterweise wegen Mordes gesucht wird. Viele Stationen werden bis Afrika angelaufen, und viele Menschen treten in das Leben von Isa und Hoel. Saint-Quentin will Isa helfen Ihren Anspruch auf deren Adelstitel geltend zu machen, dafür müssen sie aber erst einmal heil aus diesem Abenteuer herauskommen. Kranke Gestalten sowie fantastische Schauplätze warten darauf erkundet zu werden.



Das ganze Spiel läuft nicht nach dem typischen Schema eines Adventures ab. In der Originalpackung wurde der erste Band der fünfteiligen Comicreihe beigelegt, und stellt damit die Vorgeschichte dar. Dort wo der Comic aufhört, beginnt das Spiel. Es ist nichtlinear, und man kann einige Wege zum Ziel verfolgen. Deswegen gibt es auch keinen „echten“ Schluss. Im Grunde steuert man nur die Unterhaltungen und Taten der Personen und Akteure. Es kommt darauf an, wann ihr mit wem spricht oder wie lange ihr für Aktionen wartet. Alles wird in den Spielverlauf eingebunden und entscheidet über die Zukunft der Geschichte. Ein gutes Beispiel ist in der ersten Episode ein auf dem Wasser treibender Sarg. Wird dieser nicht schnell gerettet, geht er unter und es fehlt im Verlauf eine Person. Damit ändern sich dann auch alle Handlungsstränge. Das Ziel zu erreichen kann trotz unterschiedlicher Entscheidungen erreicht werden oder eben

unerreicht bleiben.

Die Story ist spannend und das Interesse beim nächsten mal anders zu handeln, um zu sehen wie sich die Geschichte verhält, ist hoch. Auch die spitzen Grafiken, welche in hohem Maße das Flair der Comics wiedergeben sind einmalig auf dem CPC. Alle Grafiken sind in Mode 0 und halten sich strikt an die Vorlage. Die gesamte Story zeigt sich wie ein Comic überlappend im Hauptfenster und gibt so das Gefühl einen Comic zu lesen gut wieder. Der Sound ist anfangs etwas irritierend, ist aber ein wunderbarer Ohrwurm und schmiegt sich bestens in die Atmosphäre des ganzen Spiels ein.



An eines der größten Versäumnisse bei Spielen am CPC wurde hier zum Glück gedacht, Es kann zwischengespeichert werden. Denn zügig spielt sich der ganze Plot in ca. 2 Stunden durch. Beim ersten Mal können es schon 5 bis 6 Stunden werden. Für Grafikfreunde und Spieler mit Geduld wirklich zu empfehlen. Für eine schnelle Runde oder nur ein kurzes Spielchen denkbar ungeeignet. Wer aber in die Story eintauchen kann und will, wird sich bestimmt auch bald nach den Comics umsehen. Erschienen bei „Comic Art“. Gezeichnet von Francois Bourgeon.

JS/Tolkin



Shaolins Road

Böse Buben und ein guter Held. Stoff der in den Siebzigern zu einer, heute wohl etwas belächelten, Gattung von Filmen geführt hatte...

Kung-Fu wurde von unzähligen Bruce Lee- und den „Knochenbrecher“ Filmen auf die Deutschen losgelassen. Jetzt sind viele Spielfilme davon in die Gattung „Trash“ abgerutscht. Was soviel heisst wie (O-Ton) „Der Film ist so schlecht, den musst du unbedingt ansehen, der ist so klasse!“ Ein Widerspruch ins sich also?!

So erging es wohl auch den Beat em Ups. „The Way of the exploding Fist“ machte den Anfang auf den Computern, und ging über in die „International Karate“ Reihe. Heute ist wohl der ungekrönte Abschluss auf dem NeoGeo das King of Fighters Franchise, welches mit der Version 2003 einige Neuerungen und „Extreme Schlagkombos“ einführte.

„Etwas“ kleiner, dafür aber für Kurzzeitspieler wesentlich angenehmer, ist wohl ein kleines Spiel welches zu seiner Zeit nicht für große Aufmerksamkeit gesorgt hatte. Shaolins Road ist für die Mitte der achtziger typisch. Erstens ist es eine Arcadeumsetzung, zweitens kommt es von einem damals noch unbekanntem japanischen Haus namens „Konami“, drittens gab man

nicht viel Herzblut in die Umsetzung. Das Scrolling ruckelt ein bisschen kann aber verkraftet werden. Schöne große Sprites und davon nicht zu wenige machen uns das Leben schwer, bzw. versuchen uns selbiges auszulöschen.

Nachdem unser Held nämlich den Grad „Chin Shaolin“ gemeistert



hatte fand er sich in einem Bandenkrieg zwischen den Triaden wieder. Nun kämpft er sich den Weg frei. Dieser führt ihn über den Shaolin-Tempel zu der Strasse der Shaolin, in die Freiheit.

Endgegner beschließen jeden Level von dem es genug gibt, um uns Joystickrüttler am Monitor zu fesseln. Die gesamte Grafik ist ziemlich nah am Arcadeoriginal und für die Zeit wirklich in Ordnung.

Schlägt man einige Gegner, kann man Zusatzitems sammeln, die uns zu tollen Waffen verhelfen, wie dem Morgenstern und den Feuerpfeilen. Sollten wir aber einmal in die Fänge der Gegner geraten, sind wir nicht sofort dahin, sondern verlieren etwas Energie. Beim dritten „Schubser“ lassen wir aber dann doch ein Leben, von denen es drei zu verlieren gilt, bevor uns das „Game Over“ entgegenstrahlt.

Der Sound beschränkt sich auf ein unaufdringliches Gedudel und einfache Effekte. Mit etwas Übung kommt man bald gut voran und reagiert entsprechend auf bestimmte Gegner. Der Schwierigkeitsgrad ist im oberen Bereich. Man muß also Geduld haben um voran zu kommen. Die Steuerung ist zudem etwas träge aber präzise. Zusätzliche Credits oder Levelcodes gibt es nicht wollte man doch Geld in den Arcaden verdienen. So kommt doch Anfangs öfters Frust auf, sah man doch gerade einen neuen Level, und muss nach verlorenen Leben, dann wieder von vorne begonnen werden.

Shaolins Road ist in seinem Genre ein gutes Spiel, und bringt einigen Zeitvertreib, doch bringt es nicht den „Hin- und Wieder- Spieler“ dazu sich damit über längere Zeit zu befassen.

JS/Tolkin



www.genesis8bit.com

Super Cyle

Es geschah irgendwann im Jahre 1986. Ein damaliger Klassenkamerad hatte sich einen C128D zugelegt und präsentierte mir voller Stolz diverse Spiele. Kaum eines hat mich dabei so fasziniert wie das Motorrad-Rennspiel Super Cycle. Die Wirkung war wohl in zweierlei Hinsicht nachhaltig. Zum einen, weil meine Eltern dank immer neuer Argumente warum der Bub jetzt auch einen Computer braucht, anschliessend kaum eine ruhige Nacht verbracht haben dürften. Zum anderen weil ich mir sicher bin, dass dieses Spiel durchaus entscheidend für meine Infizierung mit dem Motorradbazillus gewesen ist. Die jeweiligen Nachwirkungen halten bis heute vor.

Meine Eltern erhörten meinen Wunsch damals rasch und bald zierte ein CPC 6128 mein Kinderzimmer. Trotzdem war es zu meiner Enttäuschung erst Mal Essig mit Super Cycle. Nachdem das Spiel 1986 für den C64 erschienen war mussten wir CPCler noch bis 1987 auf die Konvertierung warten. Doch das Warten hat sich gelohnt, DJL Software lieferte mit der Konvertierung des Epyx-Titels saubere Arbeit ab.

Durch sich ändernde Hintergründe und verschiedenen in die Strecken eingebaute Hindernisse - die Level für Level hinzukommen - fällt gar nicht weiter auf, dass sich die Streckenführung selbst nicht gerade massiv verändert. Wie auch die Sprites sind die Hintergründe übrigens für damalige Verhältnisse recht nett gezeichnet. Von der breiten Masse der Rennspiele hebt sich dieser Titel aber eindeutig

Schaltung bereitet keinerlei Probleme. So beruht der Schaltvorgang (wenn wir eventuelle Abflüge mal aussen vor lassen) pro Level nur auf einem einmaligen Durchschalten der Gänge von eins nach drei.

Während das Titelbild noch durch einen hübschen Tune untermalt wird, konzentriert sich die Geräuschkulisse im Spiel vor allem auf den "Klang" des Motors. Die Melodie des Tunes ist übrigens sehr eingängig. So eingängig, dass man mich wahrscheinlich nachts um Drei wecken könnte und ich könnte ihn auf Befehl pfeifen.

Eine speicherbare Highscoreliste (so was wurde bei vielen CPC-Spielen ja "vergessen"), die Platz für sage und schreibe 100 Einträge lässt, sorgt zusätzlich für Langzeitmotivation. Wem der standardmässige Schwierigkeitsgrad dann irgendwann doch zu lau wird, der kann diesen noch um bis zu zwei Punkte nach oben schrauben.

Ach, etwas haben die Jungs von DJL bei ihrer Konvertierung dann doch übersehen. In der C64-Version lassen sich Farben und Design von Motorrad und Lederkombi variieren und dem Geschmack des Spielers anpassen. Aber das ist sicher leicht zu verschmerzen.

Die Kombination all dieser Faktoren macht diesen Oldie für mich zu einem der besten Motorrad-Rennspiele aller Zeiten und zu einem der besten CPC-Spiele, das mich immer wieder zum Einschalten des CPCs animiert, sowieso. Ich bin mir sicher: auch Valentino Rossi hätte an diesem Klassiker seine helle Freude.

Markus Buntru



Hanging off vom Feinsten

Das Spielprinzip ist denkbar einfach. Man tritt mittels eines Superbikes mit Drei-Gang-Schaltung gegen die Uhr auf diversen leicht variierenden Strecken an. Die erfolgreiche Arbeit erschweren einem hierbei diverse Computer-Gegner. Im Gegensatz zu vielen anderen Rennspielen sind diese in ihrem Verhalten meist recht gut kalkulierbar und rauben einem nicht durch übermässige Agressivität schnell die Motivation.

durch äusserst schnelles und flüssiges Scrolling ab. Die - heute dann doch nicht mehr so ganz zeitgemässen - 140 Mph, die euer Feuerstuhl in der Spitze erreicht, kommen überzeugend rüber. Was ist übler als ein optisch gut gemachtes Rennspiel bei dem sich nichts "bewegt"?

Die Steuerung ist unkompliziert und exakt und auch die Drei-Gang-

YAZE-AG und Z3Plus

Mit YAZE-AG bereitet uns Andreas Gerlich einen exzellenten CP/M-Emulator. Und Z3Plus macht die Sache gemütlich.

Die Installation von YAZE-AG unter Linux gestaltet sich denkbar einfach: Die Datei `yaze-ag-2.10-pre.tar.bz2` in das Zielverzeichnis kopieren und mit `tar xfvj yaze-ag-2.10-pre.tar.bz2` entpacken. So entsteht das Unterverzeichnis `yaze-ag-2.10-pre`, in dieses wechseln, `cp Makefile_linux Makefile`, dann `make` und schon wird YAZE-AG installiert. Es erscheinen viele Warnings, von denen man sich besser nicht stören läßt. Unter Mac OS X funktioniert bei installiertem Xcode die Installation genau so - nur wird hier statt `Makefile_linux` `Makefile_solaris` verwendet. Mittels `./yaze` wird YAZE-AG nun gestartet – und der begrüßt uns erst einmal mit einer Menge Text, bevor das vertraute `A>` erscheint.

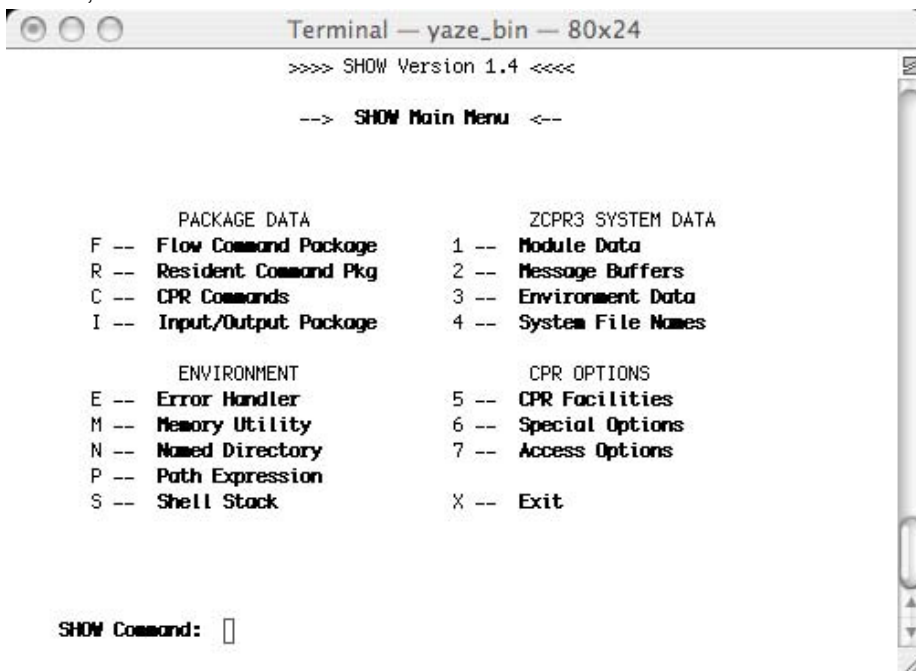
Wenn wir uns nun auf den Laufwerken A: bis P: umsehen werden wir feststellen, daß Andreas es gut mit uns meint – und eine Menge Zeug gleich vorinstalliert hat. Es ist viel interessantes dabei, aber unübersichtlich ist es auch. Deshalb habe ich beschlossen, mir erst einmal eine Bootpartition mit dem nötigsten einzurichten. Da hilft uns `SYS.COM` das uns mit einem `$>` begrüßt. Mit `?` kommt eine Übersicht der Befehle und `? befehl` zeigt eine weitere Erklärung. Also erstelle ich mit `create systemdisk 8388608` ein 8 MB großes Diskimage und binde es als Laufwerk N: mit `mount n systemdisk` ein. `go` setzt die Emulation fort. Nun noch `CPM3.COM` und `CCP.COM` auf N: kopieren, mit `E.COM` den Emulator beenden und in der Datei `.yazerc` `BOOT_UTILS` anstelle `testdisk` als Laufwerk N eintragen und die erzeugte `systemdisk` auf Laufwerk A. So bleibt alles im Zugriff und man kann nach und nach auf Laufwerk A: umkopieren, was man so braucht

und ich weiß wenigstens, was ich auf meinem System so alles drauf habe.

So, jetzt will ich aber noch etwas mehr Komfort, wie ihn z.B. Z3Plus bietet, das im ZIP-Archiv von neben

passende Modell VT100 oder IBM PC Color. Mit `lput z3plus default.z3t` wird diese in `Z3PLUS.LBR` abgelegt. Wenn alles geklappt hat können mit `z3plus off` das Z-System wieder beenden.

Beim nächsten Start wird sich Z3Plus scheinbar nicht anders verhalten, aber `zshow` bringt es an den Tag, wie in Abbildung 1 zu sehen.



dem Handbuch heruntergeladen werden kann.

Das entpacke ich nun in ein Unterverzeichnis in meinem Home-Verzeichnis, starte wieder `SYS` und binde dieses mit `sys mount e /home/almasys/z3plus` als Laufwerk E: ein. Ein solches als Laufwerk gemountetes Verzeichnis funktioniert leider nur `read only`.

Zunächst kopiere ich nun die Dateien `Z3PLUS.COM`, `Z3PLUS.LBR`, `TCSELECT.COM` UND `Z3TCAP.TCP`. Mit `z3plus` wird das noch nicht konfigurierte Z3Plus gestartet, welches auch gleich `STARTZ3P.COM` vermißt, aber es funktioniert und somit kann mit `tcselect default.z3t` die Terminaldefinition ausgewählt werden. Da die Linux-Konsole `VT102` entspricht wäre das

So sieht die Bildschirmdarstellung richtig aus. Wo wir grade schon mal da sind, können wir uns auch gleich ein wenig umsehen.

Den Befehl dir gibt es im Moment übrigens so (noch) nicht, dafür ist `SDZ.COM` da.

Mit zu den schönen Dingen bei Z3Plus zählt die Möglichkeiten, den Usern auf den Laufwerken Namen zu geben und so die Übersicht über seine Festplatte zu erhalten.

Das geht auch ganz einfach – mit `EDITNDR`. So läßt sich beispielsweise Laufwerk A: User 15 mit `a15:base` der Name `BASE` zuweisen. Mit `X` wird `EDITNDR` verlassen. Tatsächlich, `base:` bringt uns in User 15 auf Laufwerk A, und egal wo wir sind, `sdz base:` zeigt uns uns genau dieses Verzeichnis. Nur

ist die Änderung mit dem Ende der aktuellen Z3Plus-Sitzung auch schon wieder verschwunden. Erst wenn die Einstellung mit savendr default.ndr gespeichert und die so erzeugte DEFAULT.NDR in der Z3PLUS.LBR abgelegt wird, wird diese Änderung auch beim nächsten Start von Z3Plus wiederhergestellt.

Dann wollen wir uns mal der fehlenden STARTZ3P.COM widmen, welche für Z3Plus sowas ist wie eine PROFILE.SUB für CP/M Plus oder die AUTOEXEC.BAT für MS-DOS; dazu geben wir *salias startz3p* ein. Jedoch fällt mir im Moment auch nicht mehr ein, als da *lsh base:lsh.lsh* einzutragen. Die Log Shell wird nun in der Datei LSH.LSH alle Befehle mitschreiben, die wir eingegeben haben – um sie jederzeit wieder zurückholen zu

ZDENST16 dient zur Anpassung an ZDE16.COM, hier ist Generic CRT voreingestellt, ANSI Standard dagegen ist die für die Linux-Konsole passende Terminal-configuration.

Eine weitere Stärke von Z3Plus sind die Definition von Aliassen. Das könnte natürlich mit SALIAS geschehen, aber dann wird jedes Alias eine eigene .COM-Datei. Es gibt aber auch die Möglichkeit, die Alias-Definitionen in einer Textdatei zusammenzufassen. Als Beispiel ist die ALIAS.COM bei Z3Plus dabei. Diese wird in ein Verzeichnis im Suchpfad kopiert sowie die ARUNZ.COM, allerdings muß die dafür in CMDRUN.COM umbenannt werden. So, nun geben wir mal *dir* ein und... ein Verzeichnis erscheint, wie in Abbildung 2.

\$* bedeutet nichts anderes, als an *sdz* alle Parameter zu übergeben die für dir eingegeben wurden.

Nun fühle ich mich bei meiner Arbeit mit YAZE-AG immer noch so unter Linux und gebe daher öfter mal *ls* ein, deshalb sieht der Eintrag bei mir so aus:

```
DIR=ls      sdz $*
```

Damit macht es keinen Unterschied, ob *SDZ* direk oder als *dir* oder *ls* aufgerufen wird.

Selbstverständlich läßt sich Z3Plus auch unter CP/M Plus am CPC einsetzen. Hier ist es sinnvoll, auf einem Laufwerk (z.B. einer akkugepufferten RAM-Disk) die wichtigsten Dateien parat zu haben.

Damit hätten wir, denke ich, das wichtigste über YAZE-AG und Z3Plus beisammen.

Mr. AMS

```

Terminal - yaze_bin - 80x24
Vers. 1.02 (c) 1988 Bridger Mitchell
[ TPA: 0100 - DB05 54.75k ] Morrow MD-5/11/16

18:53 A0:COMMANDS>>dir

Drive A0 [COMMANDS] Files: 59/558k Free: 724k
ALIAS .BAK 4k : ECHOTST .SUB 2k : PROFILE .BAK 2k : UNCRUNCH.COM 8k
ALIAS .CMD 4k : EDITNDR .COM 8k : PROFILE .SUB 2k : VLU .COM 18k
ARUNZ .COM 8k : EXIT .COM 2k : PUT .COM 8k : XTYPE .COM 2k
BBCBASIC.COM 16k : GET .COM 8k : RENAME .COM 4k : Z3PLUS .COM 16k
CCP .COM 4k : HELP .COM 8k : SALIAS .COM 8k : Z3PLUS .LBR 28k
CMDRUN .COM 8k : HELP .HLP 68k : SAVENDR .COM 2k : Z3TCAP .TCP 12k
COPY .COM 8k : LBREXT .COM 8k : SDZ .COM 6k : ZCNFG23 .COM 8k
CPM3 .COM 46k : LDIR .COM 2k : SETDEF .COM 6k : ZDE .COM 18k
DEFAULT .BAK 4k : LINST11 .COM 12k : SHOW .COM 10k : ZDENST16.COM 12k
DEFAULT .NDR 2k : LPUT .COM 6k : STARTZ3P.COM 2k : ZDESSCR .COM 18k
DEFAULT .Z3P 4k : LSH .COM 10k : SUBMIT .COM 6k : ZEXALL .COM 10k
DEFAULT .Z3T 2k : MOUNT .COM 2k : SYS .COM 2k : ZEXDOC .COM 10k
DEVICE .COM 8k : NSWEEP .COM 12k : TCMAKE10.COM 6k : ZF .COM 16k
DUMP .COM 2k : NULU .COM 16k : TCSELECT.COM 4k : ZSHOW .COM 12k
ECHO3 .COM 2k : PIP .COM 10k : UNARC .COM 6k :

```

können. Der Cursor wird übrigens mit den für Wordstar üblichen Tastenkombinationen gesteuert.

Ein Editor muß auch noch her, hier bietet sich ZDE 1.6 an. Zuerst wollte ich die Dateien unter Linux editieren, dann in die Emulation importieren und mit einem kleinen BASIC-Programm die Linefeeds einfügen, aber das ist dann wohl ein wenig umständlich.

Ich denke, es ist Zeit, die ALIAS.COM mal genauer anzusehen. In dieser Datei steht folgender Eintrag:

```
DIR      sdz $*
```

Wenn wir nun *dir* eingeben, im Suchpfad aber *DIR.COM* nicht gefunden wird wird *ALIAS.COM* nach einem Alias namens *DIR* durchsucht und dieses verwendet.

Entwickeln mit Z88dk

Neue Programme für den CPC werden mittlerweile hauptsächlich mit sogenannten Cross-Compilern auf dem PC entwickelt. Dass es an dieser Stelle auch ohne Assembler geht soll dieser Artikel zeigen.

Wenn man „mal eben“ ein Programm erstellen will ohne in die Tiefen von Assembler einsteigen zu wollen so greift man normalerweise auf das mitgelieferte BASIC im CPC zurück. Soll es dann doch mal etwas schneller sein, so bleibt meist nur noch der Assembler übrig und wenn man dann merkt, dass die eigenen Assembler Kenntnisse mit den Jahren dann doch etwas eingerostet sind wünscht man sich eine andere Programmiersprache, eventuell sogar eine, mit der man sowieso schon tagtäglich arbeiten muss. Warum also nicht den CPC in C programmieren, schliesslich soll C doch eine relativ hardwarenahe Sprache sein. Jetzt am besten noch eine einfache Entwicklungsplattform für den PC und schon könnte man einfach und schnell einsteigen.

Bei der Suche nach einem C-Compiler für Z80 Systeme stösst man relativ schnell auf diverse kommerzielle Compiler und zwei Compiler aus dem OpenSource Bereich: SDCC und Z88dk. An dieser Stelle soll es nur um letzteren gehen.

Was ist Z88dk?

Z88dk ist ein Cross-Compiler für die Sprache Small-C, der neben dem eigentlichen C-Compiler auch eine C Standard-Bibliothek für viele Z80 Rechner mitbringt. Z88dk erweitert Small-C um viele Funktionen von ANSI-C, wobei die Funktionen weggelassen wurden, die sich auf einem Z80 System nur ineffizient realisieren lassen.

Der Name Z88dk deutet auf den Ursprung des Compilers hin. Er wurde für den Cambridge Z88 entwickelt und unterstützt

mittlerweile mehr als 20 unterschiedliche Z80 Systeme.

Installation

Für die Installation unter Windows sollte am besten der Installer verwendet werden. Er kopiert die notwendigen Dateien an die richtige Stelle und setzt ein paar Umgebungsvariablen unter Windows. Leider wird nicht automatisch das Programmverzeichnis von Z88dk in die Umgebungsvariablen eingetragen, sodass man hier selbst Hand anlegen muss. Man muss also unter „Start-Einstellungen-Systemsteuerung-System“ im Reiter „Erweitert“ den Button „Umgebungsvariablen“ anklicken und bei den Benutzervariablen die Variable PATH um den Pfad zu dem Programmverzeichnis von Z88dk erweitern. Bei einer Standard-Installation ist das „C:\Programme\z88dk\bin“. An dieser Stelle sollte man daran denken, dass die neu gesetzten Umgebungsvariablen erst dann wirksam werden, wenn man eine neue Eingabeaufforderung öffnet.

Alle Anwender, die Z88dk unter Linux verwenden wollen, sollten sich das tar.gz Paket laden und den Compiler entsprechend selbst kompilieren. Solange man nicht die aktuelle CVS Version verwendet, sollte es dabei keinerlei Probleme geben.

Alle weiteren Programmierbeispiele werden sich allerdings auf Windows und Flynn's WinCPC beziehen, da so die Binärdateien einfacher zu handhaben sind. Eine kleine Anmerkung am Rand: Unter Linux kann man WinCPC mit WINE

verwenden.

Hallo Welt!

Eine Einführung in die C-Programmierung kann natürlich an dieser Stelle nicht gegeben werden, dennoch soll anhand einiger Beispiele die Arbeit mit Z88dk beschrieben werden und wie fängt man üblicherweise bei einer neuen Programmiersprache an? Genau, mit „Hello World!“. Listing 1 zeigt

```
hello.c:
#include <stdio.h>

void main(){
    printf(„Hello World!“);
}
```

Listing 1: Hello World!

```
zcc +cpc -lpcfs -O3 -create-app
hello.c
```

Listing 2: Kompilieren von hello.c

den Quellcode des Programms. Das Programm erklärt sich eigentlich von selbst. Es wird die Standard- Ein-/Ausgabebibliothek **stdio** geladen und danach mit dem Befehl **printf** der Text ausgegeben.

Als nächstes möchte man sein Programm natürlich kompilieren und auf dem CPC ausprobieren. Dafür verwendet man den in Listing 2 angegebenen Befehl, wobei die folgenden Werte übergeben werden müssen:

- +cpc – Als Zielplattform soll der CPC verwendet werden
- -lpcfs – Standardbibliothek für Ein- und Ausgabe mit einbinden
- -O3 – Maximale Optimierung einschalten
- -create-app – Es wird zusätzlich zur eigentlichen Binärdatei eine Datei mit AMSDOS Header generiert

Nachdem das Kompilieren

erfolgreich durchgeführt worden ist, finden sich in dem Verzeichnis die Dateien **a.bin** und **a.cpc**. Die Datei mit der Dateieindung „.cpc“ ist die Datei, die den AMSDOS Header beinhaltet. Sie könnte z.B. mit einem Programm wie CPCFS direkt in ein DSK-Image geschrieben werden oder auf den CPC übertragen werden. Wir werden an dieser Stelle aber den Emulator WinCPC von Flynn verwenden, da er die Möglichkeit bietet, Binärdateien direkt in den Speicher zu laden.

Z88dk erzeugt immer Programme, die an der Adresse &6000 beginnen. Um das Programm also in WinCPC auszuführen muss man mit „File – load binary file“ die Datei **a.bin** auswählen. Nach dem Bestätigen wird dann noch die Ladeadresse eingegeben, also &6000. Die Datei liegt jetzt im Speicher von WinCPC und kann mit **CALL &6000** gestartet werden.

Wenn nichts falsch gemacht worden ist, sollte der CPC „Hello World!“ ausgeben.

5kb? Was für ein Biest!

Wie man leicht erkennen kann, hat Z88dk eine 5kb grosse Binärdatei erzeugt. Warum ist das so? Auch wenn **printf** relativ mächtig ist und dementsprechend viel Platz verwendet, ist der Compiler in der Lage zu erkennen, ob die volle Funktionalität von **printf** überhaupt verwendet wird. Wird nicht die volle Funktionalität verwendet, ersetzt der Compiler automatisch **printf** durch eine einfachere Ausgabefunktion. Daran kann es also nicht liegen, da wir eine sehr einfache Ausgabe verwendet haben.

Schaut man sich einmal die CPCFS Bibliothek von Z88dk an, so fällt sofort auf, dass hier ein ganzer Block mit Null-Bytes belegt ist. Ein Blick in den Quellcode verrät, dass die Bibliothek sowohl für die

Dateieingabe als auch für die Dateiausgabe 2kb reserviert (AMSDOS). Abhilfe schafft an dieser Stelle die Bibliothek **ndos**, die nur eine Dummy Datei Ein-/Ausgabe unterstützt. Also ersetzt man in dem Befehl aus Listing 2 **-lcpdfs** durch **-lndos** und kompiliert nochmal die Datei hello.c und siehe da: die Datei ist nun nur noch 331 Byte gross.

Darf's noch etwas schneller sein?

Wie bei den meisten C-Compilern ist es auch bei Z88dk möglich Assembler im Quelltext einzugeben. Dadurch ist es möglich, wichtige Funktionen, die häufig angesprungen werden oder zeitkritisch sind, zu optimieren, während man die unkritischen Routinen weiterhin in C schreiben kann.

Typ	Grösse	Min	Max
char	1 Byte	-128	127
unsigned char	1 Byte	0	255
int	2 Bytes	-32767	32767
unsigned int	2 Byte	0	65535
long	4 Bytes	-2147483647	2147483647
unsigned long	4 Bytes	0	4294967296

Tabelle 1: Datentypen in Z88dk

An dieser Stelle muss man wissen, wie in C Variablen an Funktionen übergeben werden und wie die einzelnen Datentypen definiert sind. Tabelle 1 enthält die verschiedenen Datentypen und deren Grösse in Bytes. Natürlich werden auch Pointer unterstützt, welche jeweils 2 Bytes Speicher belegen.

Die Assembleranweisungen werden bei Z88dk in einem Block abgelegt, der durch die Anweisung **#asm** und **#endasm** geklammert ist.

Adresse	Parameter
SP	Stackpointer für Rücksprung
SP+2	2 Byte int Wert (codeint2)
SP+4	2 Byte int Wert (codeint)
SP+6	1 Byte char Wert (code)

Tab. 2: Aufbau des Stacks in Listing 3

Typ	Register
long	dehl
int	hl
char	h=0,l

Tab. 3: Rückgabewerte n. Datentyp

Listing 3 ist ein relativ unnützes, kleines Programm, das die Übergabe von Parametern an C Funktionen demonstrieren soll. Es werden nacheinander die Buchstaben 'A'-'E' auf dem Bildschirm ausgegeben und die Zahl 42 an den aufrufenden Programmteil zurückgegeben.

Wie man an dem Programm in Listing 3 sieht, werden die Parameter über den Stack an die Funktion weitergegeben. Tabelle 2 zeigt den Aufbau des Stacks.

Um die Werte zu laden wird HL als

Pointer auf den Speicherbereich mit den übergebenen Parametern verwendet. Als Startwert bekommt HL den höchsten Parameter, in diesem Fall 'code' zugewiesen, was laut Tabelle 2 einen Wert von SP+6 entspricht. HL wird deshalb mit dem Wert 6 geladen und der Stackpointer dazuaddiert. Der Wert wird jetzt ausgelesen und in A gespeichert, damit die ROM Routine BB5A diesen Wert auf dem Bildschirm ausgeben kann.

Als nächstes wird der Wert von HL dekrementiert, sodass HL jetzt auf das Highbyte von 'codeint' zeigt. Auch hier wird der Wert wieder nach A geladen und ausgegeben. Die weiteren Befehle lesen die Bytes der entsprechenden Übergabeparameter analog aus.

```

void printchar( char code, int codeint,
long codeint2 ){
    #asm

    ld hl,6    //code (SP+6)
    add hl,sp  //'A'
    ld a,(hl)
    call 0xbb5a

    dec hl    //highbyte codeint (SP+
5)
    ld a,(hl) //'B'
    call 0xbb5a

    dec hl    //lowbyte codeint (SP+
4)
    ld a,(hl) //'C'
    call 0xbb5a

    dec hl    //SP+3
    ld a,(hl) //'D'
    call 0xbb5a

    dec hl    //SP+2
    ld a,(hl) //'E'
    call 0xbb5a

    ld hl,42  //Rueckgabewert

    ret

    #endasm
}

void main(){
    int a = printchar('A', 'B'*256+'C', 'D'*
256+'E');
}

```

Listing 3: Beispiel für Variablenüber-
gaben an Funktionen

Kompiliert man das Programm, so erhält man eine gerade einmal 107 Byte grosse Binärdatei.

Wie geht's weiter?

Wer sich gerne weiter mit Z88dk befassen möchte, sollte auf jeden Fall die Dokumentation lesen und sich eingehender mit den mitgelieferten Beispielprogrammen beschäftigen. Aus Platz- und Zeitgründen konnte hier auch nicht auf die Erstellung von Softwarebibliotheken eingegangen werden.

Möchte man mehr Informationen zu den mitgelieferten Systembibliotheken, sollte man in das Include Verzeichnis von Z88dk schauen. An dieser Stelle liegen die Header-Dateien der Bibliotheken.

Ausblick

Mit Z88dk ist es möglich komplexe Programme relativ einfach zu erstellen und das ohne allzugrossen Geschwindigkeitsverlust wie bei BASIC. Die Integration von Z80 Assembler in den Quellcode bietet ausserdem die Möglichkeit spezifische Optimierungen vorzunehmen.

Schaut man sich ein wenig bei den Ports für die anderen Z80 Systeme um, so stellt man fest, dass es dort einige Bibliotheken gibt, die sicherlich für die Programmierung des CPCs auch recht interessant wären, z.B. die sehr umfangreiche Sprite-Bibliothek für den ZX Spectrum.

Spezielle Bibliotheken für Hardware könnten zudem die Programmierung von Anwendungsprogrammen erleichtern, bei denen es nicht nur auf die reine Geschwindigkeit ankommt. Auch eine Portierung des Z88 TCP/IP Stacks könnte durchaus interessant sein und mittels Far-Pointern kann man die 64kb Barriere knacken.

Durch die einfache Portierbarkeit von Z88dk wäre es ausserdem ohne hohen Aufwand möglich den Compiler an andere Betriebssysteme wie FutureOS oder SymbOS anzupassen.

Schlussendlich wird Z88dk natürlich keinen Assembler ersetzen können, aber im Vergleich Code, Programmieraufwand und Wartbarkeit schneidet die C Lösung sicherlich nicht schlecht ab.

Tim Riemann

www

www.z88dk.sf.net (Z88dk Cross-Compiler)

www.rst38.org.uk/zsock (TCP/IP Stack für Z88dk)

www.geocities.com/aralbrec (ZX Spectrum Sprite Pack)

www.easypoint.ch/wincpc (CPC Emulator)

News

Tatsächlich sind hier doch noch ein paar weitere News aufgetaucht:

Beb (Design) und Candy (Code) arbeiten an einem neuen Intro namens "Tomate".

Iron programmiert anscheinend an einer sauschnellen Rotozoomer-Routine für den CPC+. Wo das dann zum Einsatz kommt ist noch unklar.

Ein Mensch namens Dada hat mit der Programmierung eines horizontalen Shoot-em-up für den CPC+ begonnen. Steht wohl noch ziemlich am Anfang, soll aber recht vielversprechend aussehen.

Kevin Thacker hat die Arbeit an seinem Repton-Clone eingestellt nachdem der Rechteinhaber was dagegen hatte. Er grübelt nun, wofür er den Code stattdessen einsetzen kann.