

Commodore Free

Issue 18 April 2008

www.commodorefree.com

Free to download magazine
Dedicated to Commodore Computers

Available as Text, Html, PDF, SEQ and Commodore 64 D64 disk image



What time is it? PROUDLY CANADIAN

BILL BUCKELS
ROBERT BOIVIN
DENIS COULDMBE



CLASSIC C64 Software

(C) Copyright Bill Buckels 2008

Editor

Well I may have overdone the "What time is it" and references to Aztec C and Bill Buckles. Read all about Bill in this issue and his Aztec C museum.

Lord Ronin continues his tutorials for new Commodore 64 and 128 users, I was hoping for the same sort of series about the Amiga range of machine, so if you feel you have enough knowledge about the machines history feel free to drop me an email.

I also managed an interview with Musician Rob Hubbard so you can read up on what Rob is up to now, and my review of the Commodore book is included with this issue. I did hope to have a review of Dirmaster V2 but ran out of time this month so look for that next month.

I included some stats about Magazine download issue 16 showing the number of unique downloads of the magazine for each format, its encouraging to see people still interested so thanks for reading and support with a total 1358 readers still with issue 16 I suppose the pressure is on to continue

What's that you say shut up babbling and let me read the magazine, yep sounds good to me ok then here we go.

Magazine download issue 16

Html	160
PDF	930
seq	68
text	119
d64	81

total	1358

Contents

Comments	
Editor	Page 2
Readers Comments	Page 3
NEWS	
News	Page 4
News	Page 6-7
Cartridge cases available	Page 5
PROJECT UPDATE	
DC2N project update	Page 8-9
"what Time is it" (educational software)	Page 18-21
Aztec C For Commodore	Page 22
Clipshop	Page 23-24
INTERVIEWS	
Rob Hubbard (Sid Musician)	Page 10-11
Bill BBuckels	Page 25-28
TUTORIALS	
In the Beginning Part 3 (lord Ronin)	Page 12-15
REVIEWS	
The Commodore 64 book	Page 16-17

HOW CAN I HELP COMMODORE FREE

Ok the best way to help would be "write something about Commodore" (yes for the observant I spelled the company correctly this time) _grin seriously though articles are always welcome,

WHAT ARTICLES DO YOU NEED

Well they vary contact me if you have an idea but I am looking for
 Tutorials – (beginners and Expert)
 Experiences with Commodore
 Why I love Commodore machines
 Interviews – maybe you have access to a power user

Readers Comments

Hi Nigel,

I thought it was interesting to forward you the answer of Ahmet. I was very interested in his project, but if he does not plan to follow it, it could be problematic to get this device and have no further assistance nor development.
FYI,

Cheers,
Vincent

----- Forwarded message -----

From: Ahmet Zeki Eymür
Date: 4 avr. 2008 03:26
Subject: Re: C64TPC
To: Vincent Col

Hi,
I planned releasing MacOS X binary but I may not continue developing C64TPC. I found a job again. I'm sorry but I have not enough spare time. I did not test on network but I think you can not access networkshared folder via C64TPC.

Thank you for your interest.
Ahmet

COMMODORE FREE

Oh this is sad news, maybe after Ahmet has settled in his new job he may find time to continue the project.
I did receive this news from Ahmet though

Hi Nigel,

I read my interview in this issue of Commodore Free Mag. Thank you very much. Attached e-mail is true. I can't continue developing C64TPC project.

Best regards,
Ahmet

COMMODORE FREE

Well Ahmet offered to refund me for the hardware but I asked him to send out the device, its somewhere in the postal system now so I should expect to see the device soon, hopefully by next month I will have a full review (postal system permitting) and I hope Ahmet finds time to continue supporting this very interesting hardware and software combination

Hello Nigel
I would like more information on Sid programming and creating my own editor in machine code please consider covering this as a topic in you magazine

Thanks Paul

COMMODORE FREE

Ok Sid programming and creating a sid editor there are thousands literally of Sid editors available and people like Richard Bayliss have created a how to guide for the DMC 4 and 5 editors I am currently asking if I can reprint the site or have a tutorial from Richard and maybe some more information. A number of other user have asked about how to create a SID tune and some even want to enter various competitions

But the programming your own editor sorry I wouldn't know where to start would anyone like to take up the challenge of a Commodore Free Sid editor

Nigel

I would like to make a disk magazine and wanted to ask permission if I could use your excellent disk magazine engine would that be ok

Steve

COMMODORE FREE

Steve hi;

The disk magazine is all thanks to Loadstar magazine if you go to the tools menu of the Commodore free website you will see a link to download and use the disk magazine tools, everything is in there for creating a menu editing text and converting logos for your own magazine feel free and maybe drop Loadstar a line to tell them how good you think the tools they supply are

Strangely a number of people contacted me to tell me how poor the disk mag was No music no real menu system etc, of course creating the magazine and disk (the disk is created by Al Jackson) and all the other formats HTML text and SEQ files takes time, If someone has a better disk magazine system that is guaranteed to work with Pal /ntsc and has all the tools needed to convert images text etc feel free to tell me about it. Of course it would need to be easy to use as well

Thanks for all your comments feel free to contact me an let me know of any projects or software you think deserves a mention in the magazine, or even just a "hello" I think your magazine is (good /bad /other)

Regards

Nigel

www.commodorefree.com

NEWS

MossyCon4 video

From: Andrew Wiskow
Sent: 08 April 2008 19:40
Subject: {Anything Commodore Users Group}
MossyCon4 video UPDATED (again!)

Okay... I've done some experimenting, and I think I've found a compression setting with more acceptable video image quality, and the audio is still in sync with the video. So now the MossyCon4 video is easy on the eyes AND the ears. ;)

Check it out at
<http://cottonwood.serverbbs.com/acug0447/mc4>

As a bonus, the thumbnail image that Google Video chooses arbitrarily is a freeze-frame of Lord Ronin, so it's much more fitting, I think! ;)

-Andrew

Commodore 25 Years Party

From: Robert Bernardo
Date: Apr 4, 7:42 pm

Here is another video rip of the after show of the Impact of the C64 with Bill Herd, Adam Chowanec and Dale Luck.

<http://blip.tv/file/799758>

AND

Here is a rip of a cool dvd that has some of the preshow meet and greet at the Computer History Museum. The Woz and Jack Tramiel from a spy cam view.

This is the correct link for the Impact of the C64 video:

<http://blip.tv/file/792025>

Truly,
Robert Bernardo
Fresno Commodore User Group
<http://videocam.net.au/fcug>

Commodore Free

Quite an old interview but here is Jeri Ellsworth on a television news report talking about the C64 DTV
<http://blip.tv/file/749800>

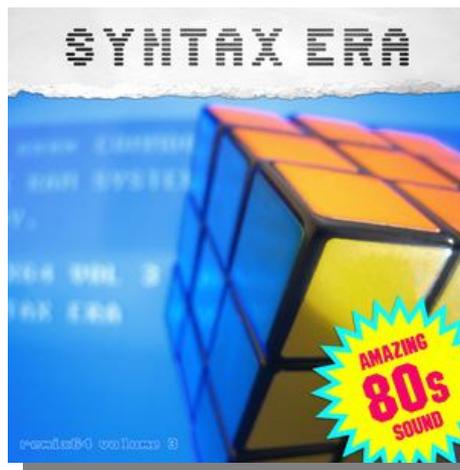
Cevi-Aktuell

In this German PDF magazine the following articles: News, Frauen und Computer, Streetrod, DoReCon 2008, Alpha Mod, Interview with Jörg Winterstein and more. You can download the magazine from the web page.

<http://www.cevi-aktuell.de.vu/>

Syntax Era

Pure Geek-cheek! Take famous 80s tracks such as 19, Two Tribes, Too Shy, Billy Jean, Broken Wings and I Feel Love, painstakingly recreate the sound,



and inject that into classic C64 music such as BMX Kidz, Comic Bakery, Sanxion, Arkanoid and Ghouls and Ghosts.

Remixes come from amazing talent: Romeo Knight, LMan, Makke, Tom Detert, Tron, Moog, Binster, Ziphoid, Boz, Sonic Wanderer, Tonka, Skitz, Trace, Slaygon, iFado and even (gasp) a guest appearances here and there from Chris Abbott.

You'll wonder how we could get away with such a cheeky tribute to the 80s: well, no samples from the original tracks were used, and the musical content has been replaced by the C64 tune covered.

The CD also features some really great songs in their own right, which result from a perfect match between C64 melodies and hooks, the sound of SID, and classic 80s instrumentation, such as Ace II, Jethro Walrus, Aspar Grand Prix, and Dominator.

The CD was the baby of Markus "LMan" Klein, who acted as Co-ordinator, remixer, artist and layout, and even made his own tea, ably assisted by Tom Detert who did much of the mastering.

CD is in now shipping! £9.99 + postage
<http://www.c64audio.com/productInfo.php?cat=R64SE>

SCACOM: New issue and New homepage

We launched our new website www.scacom-online.de.vu There you can reach the new websites www.english-scacom.de.vu and www.scacom-aktuell.de.vu too. The new English-SCACOM issue 2 is available. Goto www.english-scacom.de.vu. You can download the PDF or read it online! Also the new German SCACOM-Aktuell is available! Look at our homepage and the first virtual Commodore Museum of the world! Please Give feedback and support!

Cartridge cases available!

I have been contacted by this Ebay seller, who informed me he has a number of these cartridge cases for sale, I have purchased a couple of the clear ones because we all like to see the electronic bits especially if there are any of those nice flashy lights, just a moment 2 men have a rived in white jackets they have injected me in the arm sleep.....

Commodore 64/128 Empty Plastic Cartridge Case



Time to make your own Commodore 64 or 128 cartridges!

Cartridge cases are hard to find in bulk, this is an auction for an empty cartridge case, ready for you to put your PCB & eprom. This case is known as an Atarisoft case, it has two plastic pins/pegs and will fit any suitable PCB. This case is used, the label has been removed and will have some minor label residue.

To assemble your cartridge, insert your tested PCB, use a plastic glue and glue both shells and leave for 5 minutes, then clamp the cartridge together and leave for 12 hours (or as per the glues instructions). To finish just over label the case, you can use any front 3.5" floppy disk label (widely available & low cost) it will fit perfectly, one on the front & one on the back. Finished!

Ever wanted to make your own Commodore games/utility cartridges? This is the solution, burn your EPROM, install and plug & go! That's it!

We have listed 5x 27c64 eproms & 5x Selectable Blank Cartridge PCBs to complete your cartridges.

If you have any questions just ask, we will combine postage.

CLEAR	£2.49
Postage U.k.	£3.85



Discount for bulk

<http://stores.ebay.co.uk/The-Commodore-Store>

The Commodore Store have a varied range of Commodore items for sale on eBay well worth looking at, hardware and software most of the items are listed as brand new and indeed the Commodore 128 memory upgrade I ordered was just that New never opened with full instructions and a test disk! I always wonder where people still find these disks and hardware from.

--- **The-Commodore-Store** wrote ---

Also a friend lists these to purchase he is based in America, he just lists he doesn't have an eBay store, people can contact him via eBay or - ray.lejuez@gmail.com

NEWS

1541 Ultimate Project Available now to pre-order

Introduction to the 1541 Ultimate Project

What is it?

The '1541 Ultimate' is a storage solution for your Commodore home computer, such as the C64, C128, C plus4, C16, etc. This piece of hardware implements a 'real' 1541 diskdrive for Commodore computers, and is capable of storing the floppy disks on a modern SD-card, or MMC-card. It contains a replica of all the necessary hardware to be fully compatible with a real drive.

The '1541 Ultimate' can be used as a cartridge in the C64/C128, or as a stand-alone unit for use with other Commodore models with a serial bus. In cartridge mode, the user can view the SD-card content with just a simple push on one of the buttons, and 'mount' .d64 images into the floppy drive. In stand-alone mode, the user can use the buttons to navigate through the directories, or send IEC commands to a secondary device on the bus. This secondary device also allows saving and loading files to the FAT file system directly, but is not as compatible as the 1541 drive.

Context

The '1541 Ultimate' is a personal hobby project, that has grown into a mature device that can be very useful for current Commodore users. It enables the user to copy real 5.25" floppy disks onto an SD-card, and run them from there, without the C64 knowing that it is not a real drive it is talking to. Floppy disks are prone to aging, and many floppies are on the verge of getting unreadable. So in the context of the greater "Preservation Project" that is carried out by Commodore enthusiasts, the '1541 Ultimate' can do a lot of useful work.

Although I am not a company, I would like to make the '1541 Ultimate' hardware available for Commodore enthusiasts, especially because this piece of hardware is very powerful and can potentially do a lot more than 'just' implementing a replacement for the 1541 drive. I hope that people can understand that although this board has been made with a lot of care and enthusiasm, it is still a hobby project and not an official retail product.

Features

- Full emulation of a real Commodore 1541 diskdrive.
- Secondary 'IEC-drive' for direct access to FAT filesystem.
- FAT/FAT32 support, with sub-directories and long filenames
- C64 cartridge emulation; Final Cartridge III, Action Replay, Commodore 1750/1764 REU
- Flexible freezer menu to select, mount, create D64 images, browse disk images
- Configuration screen, with resident settings
- 'Stand-alone mode' - powered through Mini-USB connector, controlled through buttons and IEC.

Specifications

- Board size: 68 x 96 mm
- 250k-gate FPGA

- Full size 6-pin DIN connector (IEC)
- 512 kB SRAM
- 2 MB Flash ROM
- 32 MB SDRAM (the Plus version only)
- Full size MMC / SD card compatible (Mini/Micro SD possible with adapter)
- Stereo audio output
- 6-pin expansion pin-header for connection to tape port (for C2N emulation), or other future features.

Payment details

Currently, I am working on the online webshop. Until it is finished, you can sign up on the pre-order list for the 1541 Ultimate or the 1541 Ultimate Plus.

Basic or Plus?

There are currently two variants of the 1541 Ultimate, the basic (€99) and the plus (€119) version. The major difference is the amount of available memory on the card. The 'basic' has just enough for 1541 emulation and the necessary software to run; the 'plus' version, however, has 32MB of RAM, which can be used as REU (16 MB max) and in the future for a lot more interesting functions. If pure 1541 emulation is what you need, the basic version will suffice. But if you are a power-user and you are excited to see more and more features added to your Ultimate card, I would strongly recommend purchasing the plus version.

Shipping

The Ultimate card will be shipped to you as a personal postage package. If you are living outside of the Netherlands, your country may apply additional taxes upon receiving your package. Most likely, though, the value of the package is below the taxable limit, so you won't be having any problems. The following shipping rates apply:

- | | |
|-----------------------------|--------------------------------------|
| Within the Netherlands: | €3, in small box |
| Within the European Union: | € 6, as a postal package (priority) |
| Outside the European Union: | € 10, as a postal package (priority) |

For more information and details where to send funds visit the 1541 ultimate website
<http://www.1541ultimate.net/content/>

COMMODORE FREE

You can bet your life I have the Ultimate version on order and await its delivery, Expect a full review when the item arrives

NEWS

BACK IN TIME LIVE STOCKHOLM 2008

September 13, Kolingsborg, Stockholm, Sweden

It's all about the Commodore 64 and its music Live on stage: Jeroen Tel, Reyn Ouwehand, 6581, Disco Danceaway

It's time for another instalment of Back in Time Live, and yet again, it's all about celebrating the Commodore 64 and its music. The event will take place in Stockholm, Sweden, and this year it's hosted by Andreas Wallström from C64.COM and John Carehag from Ziphoid Entertainment. The guys have been around the demo scene since 1987, they have their own radio shows on SLAY Radio, they run C64 sites, Andreas played in SID'80s together with Ben Daglish – among other things. It's safe to say they breathe Commodore 64 air every day.

Oh, yeah... The event.

On stage, you will find headliner Jeroen Tel (Myth, Hawkeye, Turbo Outrun) doing his DJ act that definately will get the dance floor moving! His set

includes his classic SIDs, one great Rob Hubbard track, plus unreleased SID tracks.

A band that makes its debute this evening is the rock act 6581. The band consists of music extraordinaire Reyn Ouwehand (The Last Ninja 3, Flimbo's Quest, Deadlock), Eike Steffen, aka Romeo Knight in Amiga and C64 remix circles, plus Andreas Wallström.

Reyn will also do his one-man-band set where he starts building up a song on the keyboard, then moves over to the bass, then the guitar, the drums, and finally returns to the keyboard for an improv solo. Take a look at this movie: for what to expect

www.youtube.com/watch?v=v8UBY8Yj9c0

Disco Danceaway, who's remix of Bangkok Knights got rave reviews, makes a beloved return to the Back in Time Live stage. The wig! The glasses! This is the coolest thing since powdered milk.

The DJ of the evening is Jucke of Genesis Project and The experiMENTAL SID Show fame.

Line up is provisional and may be subject to change.

Two shops will be available at the event: The creator of Back in Time Live, Chris Abbott, will bring CD's and DVD's full of C64 goodies. More information here: www.c64audio.com.

Individual Computers will bring their newly produced C64 products, like the MMC Replay and the RR-Net. More information here: www.jschoenfeld.com.

We will also see the release of The 8Bit Philosophy from Shining Movie Vision. This is a documentary about the C64, the C64 remix scene and the people

in it. See and hear the likes of Chris Hülsbeck, Reyn Ouwehand, and Press Play on Tape. More information here: www.shining-movie-vision.de.

Will there be any C64 celebrities like Martin Galway and Jeff Minter there? Who knows. But there's a rumour that Darth Vader and a few stormtroopers will show up to arrest naughty ZX Spectrum owners. As

of now, this should really be considered a rumour though.

[Download the radio show](#) aired on SLAY Radio where the event was announced. Join the Back in Time Live Stockholm 2008 [Facebook-group](#).

Discuss the event at the [Remix 64 forum](#).

More information coming up, but for now: **Mark you calendar for Back in Time Live Stockholm 2008 on September 13!**

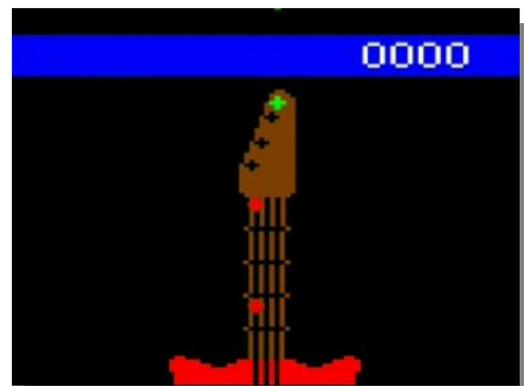
<http://www.backintimelive.com/>

Gabriel Noel, the c64 filmmaker

All: Gabriel Noel, the c64 filmmaker, here.

After over 300 hours of work spanning nine months, my latest video is done, and utilizes not only Martin Galway's title track from 'Mikie', but incorporates the music and audio effects from the Stickybear Educational Software series, which made its debut on the c64. Given the nature of what it complements (i.e. a demonstration of the 1983 equivalent of 'Guitar Hero'), I imagine it will generate a tremendous amount of interest. Anyway, here is the link:

<http://www.youtube.com/watch?v=w-cF-MN7tQ4>



enjoy, Gabriel Noelitstheit productionsgn

Eric Schwartz Amiga Tribute

Eric Schwartz made a tribute to the Amiga. It's an animation with the song "Still Alive" it tells the history about Amiga. You can

<http://youtube.com/watch?v=9mg6wrYCT9Q>

DC2N project update

<http://www.luigidifraia.com/c64/dc2n/index.html>

Last update	13 Apr 2008
Completed	99.9%
Tested	99.9% (the three main features: PLAY, RECORD, and <i>Dump</i>)
FAT-16 support	<ul style="list-style-type: none"> • read: 100% • write: 99% <p>I developed a hard real-time implementation of the FAT-16 filesystem driver.</p>
PLAYback	<ul style="list-style-type: none"> • legacy TAP v0 files • support for DC2N 16-bit TAP files may be added
RECORD and <i>Dump</i>	<ul style="list-style-type: none"> • DC2N 16-bit TAP files (counter rate is 2 MHz) • TAP version 0 files (100% MTap compliant)

01 Feb 2008

First mount
 Evening: I almost finished the first mount after Farnell sent me the sockets I needed. It came out nicely I think. DC2N assembled by me I will bring this one to the HCC Show as well. This will give me the chance to test it before sending it out to the person who pre-ordered it.

03 Feb 2008

Handmade box Evening: enthusiast built his own DC2N box. DC2N boxed by enthusiast looks nice, doesn't it?



09 Feb 2008

New box Morning: I built the first version of the DC2N box. I will bring this one to the HCC Show. DC2N connected to a C64. DC2N boxed by me That's a first go, I will hopefully complete it soon.

16 Feb 2008

HCC Show
 Morning: I presented the DC2N at the HCC Show today, where I met quite some nice people. Many thanks to the organizers!

20 Feb 2008

FAT again Evening: I compiled in and tested the fragmented file support routines of my FAT driver. A Turbotape (GGS) and a Cyberload TAP image (LN2) have been played to my C64 without problems.

24 Feb 2008

Finishing the box
 Morning: I almost finished the second version of the DC2N box. Since I was not able to find the switch I need for DC2N at the local shop, I have to stop here for now. DC2N in a box Peepo has made 693 dumps so far! Excellent job progressing with mount Evening: And here it is, a DC2N almost fully mounted with the

mode selector: c2n emulation or dumping. DC2N boxed by me

23 Mar 2008

Happy Easter!
 Morning: Based on the custom DC2N mount SLC had done, I mounted my development version: Monitor-while-dumping DC2N I can now switch the DC2N mode of operation and adjust the datassette while dumping, to perform a fine tuning for those tapes that are hard to dump, without having to unplug the datassette from DC2N. That is, the signal is sent to my C64 as well, where the "Recorder Justage" program is kept running.

The way I do that actually is as per below:
 load "Head Tester" (aka "Recorder Justage") from an SD Card to my C64 using DC2N in C2N emulation mode switch DC2N into dumping mode put a tape in the C2N connected to DC2N and press PLAY align the head and rewind the tape to make a clean dump of it!

Last update 20 Feb 2008

Completed 98%
 Tested 99% (the three main features: PLAY, RECORD, and Dump)

FAT-16 support

read: 100%
 write: 90%

I developed a hard real-time implementation of the FAT-16 file system driver. Support for fragmentation handling in write operations will be added at a later time.

PLAYback

legacy TAP v0 files
 support for DC2N 16-bit TAP files will be added

RECORD and Dump

DC2N 16-bit TAP files (counter rate is 2 MHz)
 TAP version 0 files (100% MTap compliant)

DC2N TAP format

16-bit DC2N format generated by RECORD/dump operations respectively:

Offset	Size	Description
0x00	12 bytes	ID string: "DC2N-TAP-RAW"
0x0C	1 byte	Format version: actually only version 0 is defined
0x0D	2 bytes	Reserved for future needs

0x0F 1 byte Counter resolution
 [bit]: 16 actually
 0x10 4 bytes Counter rate (LSBF) [Hz]:
 2000000 actually
 0x14 any 1 6-bit (counter
 resolution) data values (LSBF)

Each data value is the delay, expressed in clock cycles (at Counter rate), between two consecutive rising (falling) edges of the

Commodore computer write line (C2N read line) signal. 0xFFFF is an overflow value, meaning that the next data values should be summed to this one to build up the total delay, up to the first non-0xFFFF value (included). Anyway, we don't really require to deal with that since the dc2nconv software converts DC2N 16-bit files to legacy TAP v1 files, which are already supported by many emulators and tools.

DC2N TAP file resolution (2 MHz) is actually 16 times higher than the legacy TAP format (123.156 kHz).

Ordering

The first batch of DC2Ns is being already shipped to large collectors who showed interest in having a DC2N quite some time ago, for preservation purposes. A second batch of devices is being processed. Delays are likely to occur before I will be able to work at the new mounts for the electronic components I need are out of stock at the supplier I used before. Even if people cannot yet order a DC2N until the abovementioned issue is solved, pre-ordering a DC2N for preservation or entertainment purposes can be done by writing me an email. DC2N will be sold on a first-come-first-served basis. If you think you should be on the pre-order list and you aren't, or if you are there and you don't want to be anymore, again email me.

People who have pre-ordered so far

Firstname	Lastname	Country
-----	-----	-----
Nigel	Parker	UK

COMMODORE FREE

===== I have left the other list out if you are curious visit the website for details =====

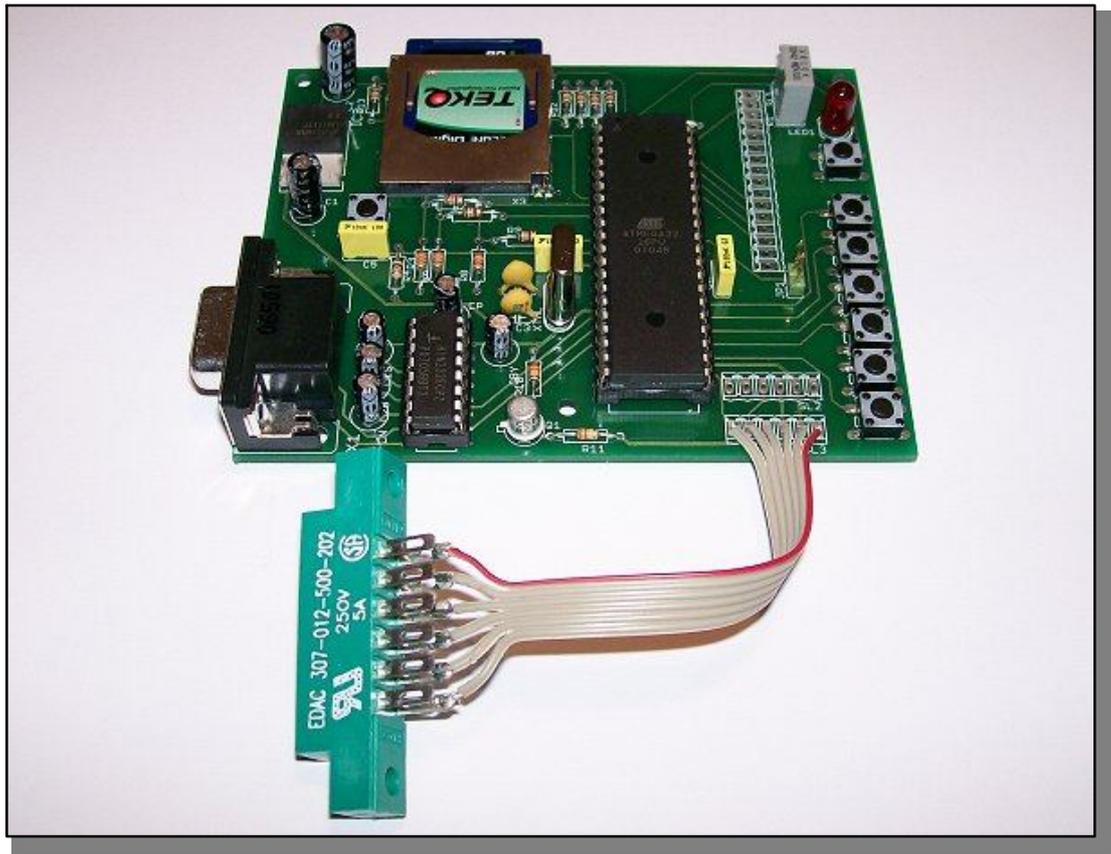
Hi Nigel,

Thank you for your email. Surprising enough, tomorrow I will start working at the DC2N to send you. Anyway, I prefer not to give out time estimations to the people who pre-ordered. I would have contacted you when the first build had been completed and tested properly. On average, it takes three weekends to be built. Sorry about that, but I have no help with this and no professional tools. I may actually be able to do it sooner for this would be my third build. I had built one that I meant to sell out, but it's now part of my test bed due to some constraints I met recently (a huge collection of tapes to image with one device while developing using the other device).

Next Project

My next coming hardware project is D1541 which I would also refer to as "True 1541". Surprisingly enough there's no 1541 replacement

That actually satisfies my requirements, so that I am going to fill this gap by myself. It's going to be all but a quick project, even if I know I will get there. My aim is to go beyond D64 and G64 formats to support all of the copy protected disks in one go.



Commodore Free Interview with Commodore Musician Rob Hubbard

Rob Hubbard possibly needs no introduction, but here goes;

Rob was one of a few musicians that pushed the SID chip way beyond what many thought possible I like many other users at the time later purchased games purely on the fact Rob had written the music.



COMMODORE FREE.

Please introduce yourself to our readers?

ROB HUBBARD

I did a ton o c64 games audio in the 1980s.

CF Are you a classically trained musician?

RH yes...

CF When creating Games music How long did it take you to switch from thinking in musical notation to thinking in terms of code/date to input into the machine?

RH. I used to sketch music ideas on music manuscript paper, and then type the music into hex define byte statements in assembler source code. After a while a lot of hex codes became second nature.

CF Rob can you tell our readers about 64 orchestra and your involvement?

RH I was approached initially by Jeroen Tel, and then once details were agreed I started working on all the scores. So I was heavily involved in deciding what the ensemble should be and the musical direction it should take.

CF So this is a real orchestra playing Commodore sid tunes on real instruments? who thought of the idea?

RH I think it was Mano.

CF Where there many problems adopting Games music to an orchestra?

RH The original SID tunes only used 3 voices plus a few software tricks. So you have to look beyond the original tunes and add other elements. With my own tunes it's easier since I have a good idea what the possibilities are.

CF I presume you originally wrote your Commodore music in manuscript format, so was this original manuscript used for the orchestra?

RH No.

CF Did you have any other arrangers help out on the orchestration, I remember from music lessons trying to work out a 4 piece orchestra and found it more difficult than just writing the music for a pianist?

RH No I did everything myself.

CF Do you think musicians can make good arrangers or is this a separate skill?

RH Well it is a skill any musician can learn. Like most things, it's a matter of putting in the time and effort.

CF writing Games music or writing for an orchestra, what do you feel is the most restrictive?

RH Not sure what you're getting at here....

CF Which of your own tunes do you think would suit an orchestral arrangement and why?

RH Most of the tunes for the c64 orchestra were chosen because they can be adapted to work with the orchestra. There are some techno type of tunes that just would not work very well.

CF Going back to the old days What features would you like to have added to your famous music routine that it did not have?

RH I don't really know. I added the sample player so that I could try to add a few real instruments with limited pitch range. So by this time there wasn't much left that it could do.

CF How did you get the idea for the music, do you sit down and think right I am going to create a tune in this format with this time signature and use this key or does the music just evolve, for example I usually go to bed with a tape machine or paper and pen and when i wake have a melody i scribble down and expand on that, although sometimes i just go to bed and wake up

RH I use various methods. Most of the time I write at the piano. Other times I write away from the piano just with a pen and manuscript paper. Other times I try to just improvise and see what I come up with. A lot of the time in games a producer wants something specific so you have to copy an existing piece. or go off in a specific direction.

CF What did you think of other composer's music routines at the time?

RH They were all very clever !

CF How did you create the samples used in games like BMX Kidz?

RH I had a c64 sampler that I used and then hacked the samples into my source code.

CF What was it like working at EA in the USA?

RH It was really fantastic in the early years. Later on EA became very corporate and the games were all very much alike, so a lot of the fun was lost.

CF Do you feel music can make a bad game seem good or better and do you think bad music can make a good game feel well bad?

RH No, a good game will always be good regardless of the graphics or audio.

CF Do you think some of the creativity has been lost now that games creators can just throw in a full song from a band or a full orchestral track, instead of working with the hardware itself to generate the music?

RH Yes it has. It is also due to the sameness of all the games, and the fact that it costs so much money to develop a game. The publishers can't afford to take any risks, and so everything is very generic.

CF Many people purchased games just because you were listed as the musician how did this make you feel at the time?

RH Well I don't know about that really....

CF Do you still follow the Commodore scene?

RH Not really. I don't have a commodore any more. I sold it in 1988.

CF The new game SHREDZ 64 allows players to play along with SID tunes using the PS2 guitar from Guitar Hero - which of your tracks should people try and have you seen the project?

RH I would guess Arcade Classics - I haven't seen the game.

CF Do you feel like a celebrity, and are you surprised people still remember your work?

RH Yes, I'm very surprised - I mean it is over 20 years ago !!

CF What music projects apart from 64 orchestra are you currently working on?

RH Just my own private musical projects and pursuits/interests.

CF Do you have a fan website?

RH no.

Links

http://en.wikipedia.org/wiki/Rob_Hubbard

http://www.c64.com/interviews/hubbard_softography.html

<http://www.last.fm/music/Rob+Hubbard>

<http://www.myspace.com/c64orchestra>

http://www.remix64.com/interview_rob_hubbard.html

<http://www.youtube.com/watch?v=fCMkWy3XaAY>

<http://www.mono211.com/gamegeekpeeks/robh.html>

<http://www.c64.com/interviews/hubbard.html>

In the Beginning Part 3

Lord Ronin from Q-Link

Right then lets do it again. You type in ..
LOAD"0:"*,8,1

```

**** COMMODORE 64 BASIC V2 ****
64K RAM SYSTEM 38911 BASIC BYTES FREE
READY.
LOAD "0:"*,8,1
SEARCHING FOR 0:*

```

LOAD is telling the computer what you want it to do. Which will be in the quotes. 0 is that hang over from two drives on one number. Commas are sort of like a separator. 8 is the disk drive number and that 1 at then end. Well that is used for ML <we will get to that later> programmes. Works for Basic most of the time as well. That * thing is a wildcard to load the first prg on the disk.

OK you then press return and something should happen. If everything is fine, you may see a fast "searching for *". Then loading it and the screen should change and the prg starts. Well, there is that old saying about the best laid plans of mice, men and machines. Nothing may happen at all. So lets troubleshoot that part.

First it could be the load command I gave you. Change it to ...

LOAD"0:"*,8 press return

This will load but not run the programme. When it has loaded and the ready is flashing type ...

RUN press return

That is the 70% of the time fix to the problem.

But, Yet, However there can be another problem. That being the first prg on the disk isn't the one that loads the files. This is found on some factory/commercial disks. If you are using one of them for your first time <I did> and are very lucky <I wasn't> Then you have the manual for the disk. Which should tell you the load command. More often you just have the disk, and if you are fortunate. The load command is written on the disk label. Now it is possible to type in the exact filename that you want to load. Like ...

LOAD"AD&D DICE",8,1press return

That will find the programme on the disk of that name and load it for you. As well as most of the time running it. A trick here is the wildcard. That is the * symbol. If you are like me, old and mentally incompetent. OK maybe not like me, but you can't memorise the spelling of every prg on your disk. You can type ...

LOAD"AD&D*",8,1press return

This command will load the first programme on the disk that starts with AD&D. We can discuss this a bit more later on. Right now, you want to see what is on

the disk and what happens to the screen. Or what is on the tape or what is on that cart, oral of the above. Take a look at the things for a while. We can do more after you have had a little fun.

.....:

What did you see? A game, a tool, a utility a something or another? Did you look at a factory disk, some backup copy or a disk that had a collection of separate unrelated files? Sure I know that you have seen a something that may or may not have impressed you with the C=. We are talking about 8bit graphics and all of that 80s stuff. You have just seen something that probably didn't make that much sense. Don't worry, I wanted you to have some hands on with programmes. See what can be done in some form. Now then we get to some of the meat of the C=.

Lets admit at this point that the thing works differently than what you already know on other platforms. How to operate a programme is different. You have to tell it to load and perhaps to run. What you saw on the screen, well it wasn't 16,500,000 colours, of which according to Ripley you can only see the 500,000. Nor does it have stereo sound. Unless you have a modified one or the Stereo SIDcart plugged into it.

Lets get into what the computer does, after we go into some background. This part is antidote in part. Part also from readings and talks with other users.

This unit came out in 1982ce. Some people say in January, others say in June. What it did was change the computer world. Yeah that little thing changed the computer world. I'll add that it changed the world period.

Tough talk and that needs an explanation. So here we go, and yeah I am and have been actually using that manual during the previous stuff. Take a look at the keyboard. Not the same as you have seen before. I mentioned the return vs. the enter key. Notice that at the top row the third key from the right is a British pound sterling symbol <|>. Don't have that on your other keyboard. Nor a C=key or a run/stop key or even a clr/home or inst/del key. Depending on your model. You will see that there are little graphic things either on the top of the keys or on the fronts. I'll have you play with them in a moment. There may also become abbreviations on the number keys at the top row. We will get to them in a bit. These are some of the differences. Here is the first one to tell you. While other platforms used/use ascii character. Having 128of them in the main set <there are expansions> Commodore uses the standard PETascii. You have about 256characters at your finger tips. If you think that means the characters or the letters are the same. Nope they are not. If you try to read a text type file from PET to ascii. What you will see ... iS SOMETHING ON THIS ORDER. WHERE THE CASE IS REVERSED. Also the extra

characters not found in the main ascii file will show up as strange garbage if they show at all.

Now to the story. I was around in the early days of computers. But not a member of it. I think that I still may have a popular electronic magazine with an article about the first computer for home use, the Altair. As I mentioned in an earlier part. I was a tech for Radio Shack. That machine at that time was black and white with no sound. Some coders would make a "thing". Didn't do anything on the screen. But it did play a song. What you did was tune a little portable radio off of a station. Put it about a foot away from the keyboard. Then while loading you had a song. Not a great song because all it was doing was modulating the Radio Frequencies that were generated when the programme loaded from the tape drive. That was the sound of that time period.

OK here comes the Commodore 64. We now have a three voice and 9 octave sound capability. Remember that I said the Radio Shack one was black & white? The 64 has 16 colours <there are ways to increase that for your work> More sound than any one else and more colours, and more memory. All in one built in system. I have monitors from that time period. They are monochrome. Black & White, Green and Amber screens. Nor do they have an internal speaker for any sound. In my collection of papers for the C=. I have a poster that gives the capabilities of the different systems of that time, along with the prices. Best with out adding onto the system was 4 colours on another platform. Though the C= disk drive was costly at first. No way did it rival the price of another company's drive of around \$1300.

The Commodore 64 became the most popular and beloved computer. Oh you can read that in the November 1998 ce issue of PC World. Also the 64 is in the 2000 and 2001 <seen those issues> Guinness book of world records.

Well the people have a "friendly" computer. In fact that is what the manual tells you. That "Commodore is known as the friendly Computer company..." <users manual page vii>

What you may not know is the mindset of that time period. At the price of a home, or Personal Computer in that time. Your average teen isn't going out and buying one. Parents have to buy one. So you have to sell the product to people in their late 20s and early 30s. Hey that was me at that time. So what is so hard about that? Well today computers are an active part of day to day life. Back then, this wasn't the case. We parents of that time didn't have the same ideas as they do now. You see going back to the 50s and 60s, whereas kids. We learned about computers. These were the things in movies, cheap T.V. shows, 25 cent paperback books, and 10-12 cent comic books. That wanted to destroy all life or take over the world and enslave the people. Sure that probably makes you laugh today. But that was the input that we had in those days. Not appositive building foundation for computers in the home.

So why should I, making \$4.00 an hour. Blow close to a grand on a computer system? I mean the computer, the monitor, drives and of course some software. BTW: I have some with original price tags on them of over \$40. Answer is, that I don't. I was buying a home and taking care of a wife and two sons at the time.

Well there obviously were people that bought computers. Commodore was doing around 2 million a year in sales. This is in the early days of a computer when most people didn't even know what it was or could do. Thinking that it was a big game machine, of a really strange typewriter of some form. At least it wasn't that electronic monster that was going to rule the world. Hmm, perhaps we were right about that part after all.

General story goes this way. Dad for some reason, buys the C=64. Brings it home. Connects it to the T.V. Then looks into the users manual. Here is where Commodore hooks you in, the manual is written just for that setup. You see, the first part of the book is the set up that we have already covered. Save that it doesn't really talk about monitors that much and we didn't go into how to connect the audio to your stereo system.

Lets get started on the things in the manual. Done enough sociological history. Time to experiment with the keys. So take out any disk or tape that you have in your drives. Just for safety and we will begin with the computer turned off, wait a few seconds and then back on again. This flushes the memory. Now you may find that your 64 has a small push button on it. I have saved that for this moment. That push button is a hacker additive. 99% probable that it is an reset button. Give it a push and let go. If the screen moves a bit and goes back to the blue screen. You have a reset button. Books and magazines had the plans and parts lists for installing these. I have a plug-in port expander that has a reset on it as well.

You know that I am being devious with these notes about additives to the C=. Slyly inserting bits on things that were done for the C=. Indicating a greater amount of product and background than you may have expected. That is the good part. Bad part is that you could be a reader with this extra stuff. Not knowing what it is about, and I don't mention it at all. Worse are the readers that have a stock machine and have to go through this extra drivel, on top of the ordinary drivel. At least you can see that it is a large world for the C=.

That all said, I must get back to the keys. Manual takes you through things in a simple, and friendly way. Great method to hook the father into the C=. He has the cheque book, and the greatest fears of the new tech of the computer. That might sound silly today. It was the reality of the time.

Press a key and you see the letter show up on the screen. Today that is taken for granted. First time the guy did that in the 80s. Seeing his name or what ever was typed on his T.V. Now that was radically impressive. A sensation of wonder and a touch of fear. OK now then I don't have to tell you about the fact the keys are like a typewriter with some extra ones. You are experienced in that already. Lets mess around with the different keys.

The CONTROL [CNTRL] key and the C=key have a specific use at this time. Yeah they do more, but not just yet. So press the CNTRL key and the number1 key. That light blue flashing box just changed to a black one. Type your name. Yeah it is all in black.

A fast look at the number keys, depending on your model, show some abbreviations as I mentioned earlier. Except for 9 and 0. Lets get to those now. You should still be in black. So just press CNTRL and the number 9. Hmm doesn't look like anything

happened right? Try typing in your name again. Ah there it is, but in what is called reverse video. Looks rather cruddy too doesn't it? Black sort of doesn't go well in reverse video over that blue background. Well you can change the background and the border. In fact there are some little home brew programmes that floated around that will give you all the colour combinations. So you can see what looks the best for you. Now just press the space bar a few times. You have now a black bar. Press CNTRL and 0. That turns off the reverse video.

Next I want you to press the C= key and the 6 key. That box, called the cursor has now turned into a bright light green colour. Try the reverse video now as well. Type a bit and do another bar. Don't look that



great next to the black does it?

A chart for you.

Control key and 1 = black
 Control key and 2 = white
 Control key and 3 = red
 Control key and 4 = cyan
 Control key and 5 = purple
 Control key and 6 = green
 Control key and 7 = blue
 Control key and 8 = yellow
 Control key and 9 = reverse on
 Control key and 0 = reverse off

C= key and 1 = orange
 C= key and 2 = brown
 C= key and 3 = light red
 C= key and 4 = dark grey
 C= key and 5 = medium grey
 C= key and 6 = light green
 C= key and 7 = light blue
 C= key and 8 = light grey

Should add here for those that are doing this on a 128. 80c mode has two different colours.

Lets do bit of adjusting. Press CNTRL and 9. Putting you in reverse video. Now go through the colours above. Use the space bar press for about 6 times and change the colour. Put all 16 of them up there. Sure the blue won't show up. Now once you have that done. Adjust your Monitor or T.V. until the colour, tint, hue and brightness are acceptable to your eyes. We all have different levels of comfort with the settings.

Now are you bored with that border and screen colour? Well I do and like to change it. So here is a way to do that before we go onto more things. Cursor to the start of a new line on the screen. Type POKE53280,0:POKE53281,0 then press return. You should now have a black screen and a black border. Along with those colour bars.

Now that is a fancy thing to see, especially for the first time in your life. Well it isn't in the start of the users manual. I'm tossing it in here for you to see a feature. Remember this manual was written for the first time ever computer user in a world that didn't have any computer background. OR better said, the manual was written to interest you in the computer and keep you interested. At the same time not to get

to techy and "nerdish". You see the image of that time of a computer user was the geeky guy in the glasses, white button down shirt with the plastic pocket protector, stuffed with pens. I don't think we fit that image anymore <VBG>. What I am saying with this is that the C= changed the way people thought about computers. For years the 64 was the #1 seller. Before the 64, Commodore had the Vic-20. Before that the PET line. Taking us back to round 1977ce. I heard a report that when they started in 1980ce the Vic line. There was still a 2 year backlog for the PET system. But hey if you are interested in that stuff, it is online in mail lists and websites, and very well stated in that On the Edge book by Brian Bagnall. Plainly stated, Commodore won the people over to the idea and use of a Personal Computer.64 being the greatest one of the lot. As to why they lost out and the things that they did that were wrong. Well I again refer you to that book. At this point I am telling you about how great the 64 is and how to use it. Idiomatic practices of people in the rarefied atmosphere of that company is for other writers.

Right then, back to the keys. There is that one which is labelled as "CLRHOME". On this 128D it has the home under the CLR part. Anyway trivia aside. Press it now and see what happens. Right the flashing cursor goes up to the top left hand corner of the screen. Before you ask, yea hit will overwrite anything up there if you type new stuff on the screen. Speaking of the screen. It's pretty cluttered up now isn't it? Well you could turn off the 64 and back on again to clear all that stuff. Wasteful of time and energy. So there must be a better way to clear the screen. Well yeah you could cursor down a bit until all the stuff is off the screen. That also takes time. Tell ya what. Cursor down a few lines. There that is enough, now press the SHIFT key and the CLRHOME key. Bingo, screen is cleared and you have the cursor at the top left. Generally this spot is called "HOME". Techy words for it is row 0 column 0.A way of telling you that we count in many computer related things, starting at 0 rather than 1. Pick the colour that you want to use for this next part. Now type in CITY GOT KLAWS Spelled worse than I usually do in these things. Bad grammar as well. Must be a way to fix that, besides sending me back to remedial English.<G> There is and this is where we mess with the cursor keys. They are at the bottom right of the key board. Stated CRSR with arrows. First one in a left to right read has the arrows up and down. Press it as you it before in moving the cursor. Real easy it goes down. How to make it group? Since there isn't an up cursor key? Press the shift key and

that cursor key. See it goes up. Takes a little getting used to the springs in the keyboard for accuracy. I'm not going to tell you about my attempts at the start in getting the cursor on the right line. Now there is the next key. This has left right arrows. Press it and the cursor moves to the right. Bet you can guess how to move it to the left. Yuppers that shift key. More on the shift key later on. Now then correct the spelling in that example by moving the cursor over the wrong letters and type in the correct ones. If you feel like it. Type over the got and turn it to a has.

Real simple, and now to the next part. Type in HERE ISAN EXAMLE This should read "there is an example" How to fit it? Cursor of course to get to the bad areas. But then what? Well lets go step by step. First cursor to the HERE, put it right over the H. Now press the shift key and the INST DEL key. Every thing moves to the right one space. You now type in the T. That done, cursor over to the ISAN. Place the cursor on the A and press shift and INST DEL. Making the blank space. Finish with that and cursor over to the ML part and do the same. Adding now in the space the P. That is the way to INSerT spaces. To DELETE something. Just press the INSTDEL key. See it moves right to left and deletes everything in it's path. Not that hard, but it is for those without the manual. Seen too many of them in the past. BTW: with my crippled hands. I have to use this feature quite often in writing.

Now I want you to delete the things you have put on the screen. Either with the isn't del key or that shift clear home command. Good, now type a few things. Lets just do one row of keys on the keyboard for this experiment. Do the one the one that starts with the A. Don't worry about the : ; = keys. Just the letter keys. Got that done? OK now cursor off of that line and then I want you to press the SHIFT and the C= key. Hey there you are now in lower case. Try typing something that uses Capitals. Yeah that is there as well. It seems from what I have read and heard over the years that UPPER case, or now called screamers, was the standard or text on the screens. Commodore changed that as well. Oh yeah press the C= and Shift key again and you will go back to upper case. OK now then we get a bit complicated. Also a bit more typing for you. Clear the screen. NOW! Pick a colour you like. I use Cyan mostly. I want you to do the keys row by row. In this manner. First as it is on the screen. Start with 1 and end with the /. Yeah I want the + - | * ~ : ; = , . and the / keys as well.

Right, you see a meaningless line of letters, numbers and punctuation. Cursor down two lines and start at the left edge of the screen fo this next part. I want you to do the same as above. BUT!!!! hold down the shift key or use the SHIFT LOCK key. Hey there are different characters on the screen. What are called graphic characters. In fact they are the ones that are the illustration on the right hand side of the keys.

Now, cursor down two lines. Start at the left side of the screen. Release the Shift lock, if you used it. Type all the keys again. BUT THIS TIME hold that C= logo key as well. Yeah there you have it. This makes the characters on the left hand side of the keys.

Now I have a T.V. show to watch and a RPGame to run. You, probably are feeling a bit creative at the moment. All those characters to use and the colours that are there as well for the characters. Go ahead and make a box with your name in it. Change colours, of things. Make a step pyramid. Move on the

lines a bit, change the grey scale colours and see if you can make it look a tad bit 3D.IOW go play right now, and I'll be back later. Oh yeah, don't use the RETURN key at this time. I know you will. But it was a friendly word of warning. Won't hurt anything. You will just get a message from the 64on the screen. Nothing to worry about. What did you discover? Boxes that you can use colour on to shade and make look 3D. Using the round circle parts with the right angle parts to form letters? Perhaps you made a little scene with a stick figure? Tell the truth I did that on my BBS to illustrate some placement of the characters in the PBEM game.

Bet you didn't make a Xmas scene with tree, presents, snow, fire place and all of that. OK if you did, you weren't the first one. I have that scene from some artist. Anyway that sort of drawing is called keyboard art. Another item for the future discussions. I wanted you to see the characters from the keyboard and play a bit. Can't do this on other platforms. Since we are short on space in this part. I'll lay this on your fast. There were/are software programmes that will let you REDESIGN these characters from the keyboard. Not just as in a new font. But yeah the graphic ones as well. We aren't going in that direction right now.

RETURN or RTN or other shortened styles of writing it. Said before it is like the enter key on other platforms. In essence it takes what you put on the screen and sticks it into the computer memory. Now you see why you got that funny message when you pressed return while doing the colours and characters. RESTORE key is to "restore the computer to the normal state it was in before you changed things with a programme or some command." Confusing isn't it? Lets not muck about with it at this point in time. Most of the time I use it with the RUN/STOP key to stop and reset. Speaking of the RUN/STOP key. By itself, it will stop the operation of a Basic programme. Hang on a moment. That is what the book says. However there is a command in programming that will defeat that RUN/STOP key. SHIFT and RUN/STOP together, will load a programme from the tape drive. Advanced disk drive load command is LOAD"0:"*,8,1: <use shifted run/stop> That does the return and will start loading most programmes. If you have jiffy dos, Shift Run/Stop will load and run the first file on the disk.

So now press run/stop and restore.Restting everythng. Here is a code to type in, SYS64759, and press return. Pretty much same thing as Run/Stop and Restore. One that isn't in the manual for you to try. Type inSYS64738 and press return. I use this one for resetting back to normal. Try the two different codes and see what you like. Last point in this part. Not covered in the book. How to do that screen and border colour change. Laid out the poke command asPOKE53280,X for the border andPOKE53281,X for the screen. X here indicates a number. But what number? Ah, that is where it gets confusing. The numbers on the keyboard that you used for colour changes aren't the same for this part. Only because the count for this starts at 0. Meaning that for the border and screen colours in the POKE statement. Black is 0. But on the keyboard it is Control and 1. Does get confusing at times. Play a bit and we will do more in the next section.

The Commodore 64 book 1982 – 19xx

Information

the website address - <http://c64goldenyears.com>

£17.99 one fixed rate for post & packing worldwide
pay by PayPal or credit card

Foreword

written by Jeff Minter

Over 200 game reviews, all in full colour with screenshots and cover scans

Size of a Spectrum

Physically the book is about the size of a ZX Spectrum in size, this ties in with an earlier work about the history of Spectrum games and so the both books (should you decide to purchase them both) look nice on the shelf. The cover shows a



Commodore 64 wrapped around the front and onto the back the Commodore keys have been doctored to read the book's title and picked out in a lighter colouring than the other keys to make the title stand out.

Opening the book and you could be back in the 60's as the use of very vivid almost acid style colouring. If you were a fan of the Jeff Minter colour cycling then you will feel at home with this. Talking about Jeff Minter he provides the forward for the book. After a few small intros and the usual thanks and acknowledgements and references used for the book we are on the contents pages.

Contents

The contents page is broken down by year and then each year by game listed alphabetically from A to Z. Some years have more entries than others and I suppose some users reading would complain that his or her favourite game isn't listed and wonder why, it would be an impossible task to accommodate every game ever created into such a book. Some games are listed because they are classic games others because the graphics and sound stood out from other titles of the era.

History

After each year or chapter there is a page with some history or general information about Commodore tying into the year. The first section for example details the birth of the Commodore 64 from early prototypes that were on show at the CES in 1982 going to later chapters about the SX64 portable

Commodore 64 and the Commodore 16; Later chapters detail the introduction of the Amiga range of machines and there is even a section about CMD and the supercpu and the DTV designed by Jeri Ellsworth.

Scoring

Each game has a brief description and some witty one liner to tie in

with the game's title or content. Each

game lists the Developer and distributor with a couple of in-game screenshots and a picture of the Cassette or disk cover where applicable. Also shown at the top of the page is an icon system showing if the game was released on Tape, disk or Cassette or multiples thereof.

Very amusing to me at any rate is that each game made was obviously reviewed at sometime by the various Commodore magazines of the day; the amusing part comes from the fact that their were such wide varieties of scores given by these magazines, I presume some form of money changed hands for the reviewers and the low reviews were a result of no funds available for the reviewer. The reader of said magazine would look at the screenshots see the review purchase the game with rave reviews and then take it home load the game up and find it was in fact well should we say "rather poor", I always took these with a pinch of salt, preferring a "try before you buy" option that some of the independent shops provided. Reading the book reveals some of these rather amusing anomalies in the scoring system of the magazines for example on the first game review we see Axis Assassins Your Commodore gave a score of 6 out of 10 Commodore user decided to award 1 out of 5 and ZZAPP 64 awarded 70% how does that work out then.

Index

Listed at the rear of the book is a full index from a to z of all the games covered by the book the final few games bring the reader up-to-date with mood 64 that started life in 1996 and has been worked on but is still unfinished and the rather excellent Bomb Mania from Protovision <http://www.protovision-online.de/main.htm> released in 1996 later on we have a page referencing the Metal Warrior series of games 1999 to 2004 and then Metal dust for the Super cpu Ending with Jeri Ellsworth and the DTV joystick with 30 included games

Summing-up

Although the book is relatively expensive costing £17.99 the book is 248 pages including the index and intros and can you really afford to miss this piece of history Mine came with a note from Andrew Rollings thanking its reader for the purchase and support

Favourite games from the book

My 5 Favourite games in the book in no particular order are :

Mood 64	Well everyone said it cant be done on a stock c64
Bomb Mania	Because my wife enjoys it
Lemmings	Again it cant be done they said
The Sentinal	Because it was so strange and kept me off school for so long
Iridia Alpha	Brilliance
Leaderboard	The animation for the time was breathtaking
Monty on the run	The music shocked me so much it was absolutely great (see rob Hubard interview)
Paradroid	Excellent idea for a game
Batalyx	6 games for the price of 1 truly an idea of brilliance
Anticipal	Finally I get the chance to kill a gerbil
International soccer	Great multiplayer fun

Drat that's 11 games! But as I am reviewing I have decided to make 11 the new 5

F.A.Q (from the Website)

1. How many pages will the book have?

254 pages, a similar size and format to Andrew Rollings' Spectrum book here is the web link. <http://www.zxgoldenyears.com/>

It will be full colour throughout. There will be an introduction, a contents list and then chapters for each year starting with 1982. Towards the end of the

C64's commercial life, there is one chapter dealing with 1993-1994, and a final chapter looking at 1995 to the present day. Each chapter opens with an historical look-back at the computer industry of the time and what was happening with the C64. Each review then has screenshots, a scan of the cover artwork and the review itself. This is split into three main sections - a review of what I think of the game, trivia about the game or its creators, and finally a synopsis of the story and how it plays.

2. I'd like to see "Game X" in the book - will you include it?

There will be a definitive list of games on the site as soon as it is finalized (see question 8), but feel free to make suggestions via the contact page. In the case of sequels and long-running series, there will be references to them in an entry for part of that series (e.g. Last Ninja Remix is discussed under the entry for Last Ninja 2).

3. Will it be all arcade games, or will there be games that require a lot of thought?

There's a good cross-section of styles - from platformers to driving games, text adventures to board games, role-playing games to shoot 'em ups.

4. How were the games chosen?

A combination of methods was used to whittle down the list. First of all, top game lists from sites like Gamebase64 and Lemon64 were combed through. Secondly, many of the titles are games I played and remember fondly for various reasons. There are also a few obscure titles and several that have a historic significance. The final list was then sorted by copyright date and alphabetical order, and organised into chapters for the book.

5. How were the screenshots taken?

It would be impractical to organise screen photography, so PC emulators like Vice and CCS were used to capture the screenshots. The palettes used are designed to give a "bright" image on the page.

6. What are your qualifications for writing this book?

I've been playing games for 25 years and owned a Commodore 64 for more than 20 years. I have a large collection of games and still play on my "real" machine regularly, as well as taking advantage of emulation to play titles I didn't get the chance to play before. As a freelance writer, I did reviews for Commodore Format in its last few issues, and the 2005 "Def Tribute to ZZAP 64". I was also "Professor Brian Strain", aka The Mighty Brian in the pages of Commodore Force. Throughout the 1990's I wrote for fanzines and disk magazines. I've also been published regularly in Retro Gamer, interviewing programmers and writing about classic games.

7. The Spectrum/Amstrad/MSX/Atari was a better machine

All of the 8-bit machines had their own particular strengths and weaknesses, and often a game would be better suited to a particular machine. As a gamer I played a lot of Spectrum and Amstrad games back in the day, but the C64 was the first machine I owned and loved. So that's what I am writing about.

"What Time Is It?"

ReadMe for Commodore 64 Version

(C) Copyright 2008 Bill Buckels All Rights Reserved.



Canadian Schools.

Bill's partner on the project (and friend and mentor) died of a heart attack, was revived, and has spent from that time to this recovering, so was not capable of delivering the C64 portion of the project which he had undertaken.

As many of us know programming is a "team sport" and sometimes (sadly) a game needs to be

postponed or called-off when one of the players (or in this case half the team) gets hurt. Bill decided not to produce the C64 version at that time.

Introduction - Apple IIe Version

"What Time Is It?" was (and is) an Apple II program written by Canadian Software Developer Bill Buckels and was once distributed in the early 1990's by a publisher called Class Software in Winnipeg, Canada in both an English and a French Version. Class Software is now gone.

"What Time Is It?" was intended for use as "Courseware" in Canadian Classrooms for teaching Elementary School children how to "tell time" using various analog and digital clock representations.

The French Version of "What Time Is It?" was (and is) called "Quelle heure est-il?".

Both English and French Versions for the Apple IIe are available for free download as emulator disk images at:

<http://www.clipshop.ca/DiskImages/>
<http://www.clipshop.ca/c64/timed64.zip>

Introduction - Commodore 64 Version

In early 2008 as part of his efforts to produce an Aztec C programming environment for the Commodore 64 (C64), complete with sample programs, the developer of the Apple II version of "What Time Is It?", Bill Buckels, decided

that he would create an English-only version for the C64 as a demonstration program in the C programming language using an obsolete C64 compiler program called "Aztec C". This was the same language and the same make of compiler that he had used for his Apple II versions.

During the time years before when he created the Apple II versions, "What Time Is It?" was one of the programs that he had purchased the C64 Aztec C

compiler to produce. Due to an unfortunate chain of events at that time, programming the C64 version was never started, although some C64's were still in use in

The Market disappeared for C64 programs almost immediately after. (The Apple IIe followed and vanished a year or so later from the Software Budgets in Canadian Schools). The idea of a C64 version of "What Time Is It?" (and the compiler to create it) were shelved and forgotten until recently.

Although it is now sometime in 2008, and he had intended to have his partner write the C64 version almost 20 years ago using a compiler produced almost 25years ago, Bill has finally produced the C64 version of "What Time Is It?" in Windows XP in less than 2 weeks of his spare time including the time required to prepare the C64 graphics and music.

It runs nicely in the WinVICE C64 Emulator in Windows XP (it has never been tried on a real C64), and is as complete in every way as the other versions, accounting for differences between the C64, the Apple IIe, and the IBM-PC. It turned-out to be a good choice for an Aztec C64 production demo program.

Also, for Bill, this now answers the question whether or not Aztec C for the C64 would have been capable of producing a comparable version of "What Time Is It?" to that of the Apple IIe. This was just one of those things that had to be done or he would never have



really known the answer.

There is now no doubt that Aztec C was the right tool for the job, and only the fickle finger of fate that took away the livelihood of his friend and partner prevented the release of "What Time Is It?" before today.

History - Apple IIe

"What Time Is It?" was originally developed for use on the IBM-PC by French-Canadian "Courseware" developers Denis Coulombe and Robert Boivin on behalf of Centre de recherche appliqué pour l'ordinateur (CRAPO) in 1989 and marketed to schools across Canada as part of a six program set called "Collection Les Petites espadrilles".

Note:

Espadrilles are casual flat sandals originating from the Pyrenees. In Quebec French, however, espadrilles is the usual term for running shoes.

The Apple II version was written by Canadian Software Developer Bill Buckels on the IBM-PC in the C programming language using the Manx Aztec C 6502Cross-Compiler for Apple IIe ProDOS 8. It was completed in mid-1991.

Bill did not have the source code for the IBM-PC version of the program and it wouldn't have helped much anyway. The IBM-PC was a much more capable and sophisticated platform than the Apple II, so producing "What Time Is It?" for the Apple II required a complete rewrite "from the ground-up".

The Manx C compiler provided a rudimentary Graphics Library, but came nowhere close to what Bill needed to complete "What Time Is It?" and the 3 other programs in the "Collection Les Petites espadrilles" that he had undertaken to write. So he set to work to create the graphics and the sound routines that the Apple versions needed to behave as closely as possible to their IBM-PC counterparts.

The IBM-PC graphics of the day that CRAPO used (4 Colour CGA Mode) were vastly different from Apple II graphics (which were a coarser resolution), and each and every screen and graphics image needed to be redrawn, and reformatted for the Apple II. Using the Graphics from the IBM-PC version as a starting point, Bill developed his own Apple II graphics file formats and programmed his own set of conversion tools which he ran on the IBM-PC to produce the Apple II graphics after creating and editing these in IBM-PC



format to suit the Apple II display. He did this in conjunction with writing his Manx C graphics library, since both tasks were complimentary. When the graphics images and graphics and sound library routines were completed he moved forward to writing the program.

Not only did the Apple II have less capable graphics than the IBM-PC; it also ran more slowly with only 128K of memory, some of it unusable, and with slower disk access and a smaller floppy disk size. To overcome all of this Bill broke his Manx C Apple programs into small modules called "overlays" which ran in very little memory, and he used the upper ram bank of the Apple II to store his graphics libraries and other data to avoid loading from disk where possible. Since the Manx C compiler translates its programs into machine language which runs as quickly as can be on the Apple II, no optimization or additional "tweaks" were required, except to script the game levels using external scripts (also of Bill's design) which further saved on disk space and program memory, and which avoided slow processor intensive operations that would otherwise have been needed.

The finished result ran more slowly than its IBM-PC counterpart, but for all practical purposes, "What Time Is It?" for the Apple II had all the same functionality.

Recent History - Apple II

In mid-2007 Bill Buckels resurrected the Apple IIe versions of his programs from "Collection Les Petites espadrilles". and released them as Apple II Emulator Disk Images, rebranding them "The Little Running Steps Collection".

Although Bill is the sole author and also the architect, the designer, and the programmer of the Apple IIe English and French versions of "What Time Is It?" and the other 3 programs in the "The Little Running Steps Collection", he still includes the two Quebecers' names who wrote the IBM-PC versions on his title screens, listed in blue below his own.

History - Commodore 64

In December 2007 after resurrecting his C64 Aztec C compiler, Bill Buckels set to work to produce a C64 programming environment that would provide equal capabilities within the C64's limitations and features as to what he had prepared many years before for Apple IIe programming in Aztec C. After completing the C64 environment in less than a 3 month period, he was finally able to produce a C64 version of "What Time Is It?" as complete in every way as the other versions, accounting for differences between the C64, the Apple IIe, and the IBM-PC (as noted above).

In fact the C64 Version runs at the same speed as the Apple IIe version in less memory, with equal or better graphics, and better sound. Some of the techniques that he used to save space on the Apple IIe are the same, such as the use of overlays and some C64 techniques are more efficient memory-wise, like the use of compressed graphics. He wrote additional tools as well that were similar to the tools on the Apple IIe that allowed the embedding of graphics,

sound, and even font sets and cursors directly in the program.

The Aztec C Compiler for the C64 is not as sophisticated as its Apple IIe counterpart, and provided no Graphics Library or Sound Routines at all. All this was written "from scratch" for the C64. Despite all that, which after all these years is a given more than a surprise, Aztec C was still equal to the task and was close-enough to the Apple IIe version, that the re-use of many routines from the Apple version was possible with fix-ups for the differences between the two machines. Because he had already designed the architecture to run in a low memory environment, in many ways the C64 version is a "port" and an optimization of the Apple IIe version, which in turn was an optimized but original rewrite of the IBM-PC version that he never saw the source code for.

System Requirements - Commodore 64 Version

"What Time Is It?" requires a Commodore with 64K of memory and a floppy disk. It runs under the BASIC 2 Operating System.

It can also be run from a disk image file in an C64 emulator. It has been tested in Windows XP using WinVICE, and probably works in others using the d64 diskimage format.

Starting "What Time Is It?"

Put the "What Time Is It?" disk in the floppy drive and start the computer. Type LOAD "***,8" and press [Return]. After the program loads, type "RUN" and press [Return] and wait until the Main Menu Screen appears.

If you are running "What Time Is It?" in an emulator, follow the same basic steps. In WinVICE just click-on the disk image if you have file associations set, or select it from within WinVICE and auto start, or from within your cmd window if you have file associations set, type time.d64 and press enter and it will auto start.

Keys and Navigation

The Arrow Keys are used to navigate the menus and to select choices during activities (Game Play). The [RETURN] Key or the Space Bar can be interchangeably used to enter selections.

The X Key is used to return to the Main Menu and to Exit the program.

The Sound Toggle Key - S toggles the sound on or off at the Main Menu or during any of the activities.

The 3 Key - Pressing the number 3 at the Main Menu will use 3 digital clocks for the first 10 screens of multiple choice answers during activities

(Game Play). This is "level 1". After the first 10 screens, 5 digital clocks will be used. This is "level 2".

The 5 Key - Pressing the number 5 at the Main Menu will use 5 digital clocks for 20 screens of multiple choice answers during activities

(Game Play). This effectively skips "level 1" and goes directly to "level 2".

Note: Each time the Main Menu is displayed the default of 3 digital clocks is reset. There should be no need to press the 3 key unless the 5 key has been

pressed and needs to be reset back to 3 digital clocks before starting a new activity (Game).

The AlphaNumeric Keyboard keys are used where required to enter information during activities (Game Play) in all programs in the "Collection Les Petites espadrilles" including "What Time Is It?". A mouse is not supported.

Title Screen

When the program loads the Title Screen will display during loading, then the Main Menu will appear when the program starts after it is loaded.

Main Menu Screen

"What Time Is It?" has 3 Learning Activities (Games) which can be selected from the Main Menu by moving the Snail Cursor with the Arrow Keys to the desired activity then started by pressing the [RETURN] Key.

Each Learning activity is a Game based on a different type of analog clock formatted as a series of 20 screens presenting the time on the analog clock as a question and 3 or 5 multiple choice answers in digital clock format.

For the first 10 screens 3 digital clocks are displayed. This is "level 1".

For the following 10 screens 5 digital clocks are displayed. This is "level 2".

The student selects the answer for each screen, and when 10 screens are complete, the Student is presented with a musical reward. After the reward for completing 20 screens, the student is returned to the Main Menu.

Activities followed by musical rewards is the basic format of all the programs in the "Collection Les Petites espadrilles".

The fourth Main Menu choice exits the Main Menu and ends the program and returns to BASIC. It is the equivalent of pressing the X Key.

Sound can be toggled on or off at the Main Menu or during any of the activities by pressing S.

Activity 1: 12 Hour Analog Clock - Hours Only

Background

The 12-hour clock is a convention of time keeping in which the day runs from midnight to noon, then from noon to midnight, and is divided into 2 periods of 12 hours, numbered from 0 to 12.

The 12-hour clock is only dominant in a handful of countries, particularly the United States and Canada (except Quebec). The 24-hour clock is the most commonly used time notation in the world today.

Game Play

This activity teaches hours in the 12-hour clock time format. For each screen, a random time is displayed on the 12-hour analog clock.

To the right of the analog clock is a multiple choice menu of different times in digital clock format, with only one that matches the time that is displayed on the analog clock.

For each screen, when the Snail Cursor is moved to the digital clock that matches the analog clock, and the [RETURN] Key is pressed, a section of the Drain

Pipe on the left of the screen will fill with water. For every 10 screens the water will empty from the drain pipe, and give the Snail a "bath".

Activity 2: 24 Hour Analog Clock - Hours Only Background

The 24-hour clock is a convention of time keeping in which the day runs from midnight to midnight and is divided into 24 hours, numbered from 0 to 23. This system is the most commonly used time notation in the world today. The 12-hour clock is only dominant in a handful of countries, particularly the United States and Canada (except Quebec).

24-hour notation is in the US and Canada also referred to as military time or astronomical time, and in Australia as army time. In some parts of the world, it is called railway time. It is also the international standard notation of time (ISO 8601).

Game Play

This activity teaches hours only in the 24-hour clock time format. For each screen, a random time is displayed on the 24-hour analog clock. The analog clock is in the form of a "cuckoo clock". To the right of the analog clock is a multiple choice menu of different times in digital clock format, with only one that matches the time that is displayed on the analog clock.

For each screen, when the Snail Cursor is moved to the digital clock that matches the analog clock, and the [RETURN] Key is pressed, a section of the Hour Glass on the left of the screen will fill with sand.

After the first 10 screens ("level 1") the hourglass will be half-filled with sand and the cuckoo will come out of the clock and "sing".

After the next 10 screens ("level 2") the hourglass will be completely filled with sand and the cuckoo will come out of the clock and "sing", then the student will be returned to the Main Menu.

To make the activity more challenging in "level 2", the

higher-order numbers on the clock face will disappear every time the time changes. If the wrong answer is pressed however, they re-appear until the time changes again.

Activity 3: 12 Hour Analog Clock - Hours and Minutes

Background

The other two activities in "What Time Is It?" present time in an "Hours Only" format. Adding a second objective of minutes introduces an additional base of 60 to the base 12 and base 24 arithmetic that the student must perform in order to tell time in hours.

The approach that "What Time Is It?" uses is to break the minutes into 5minute intervals which allows the student to count by 5's and 10's to gain a relative framework for the base 60 arithmetic that minutes and seconds require.

Game Play

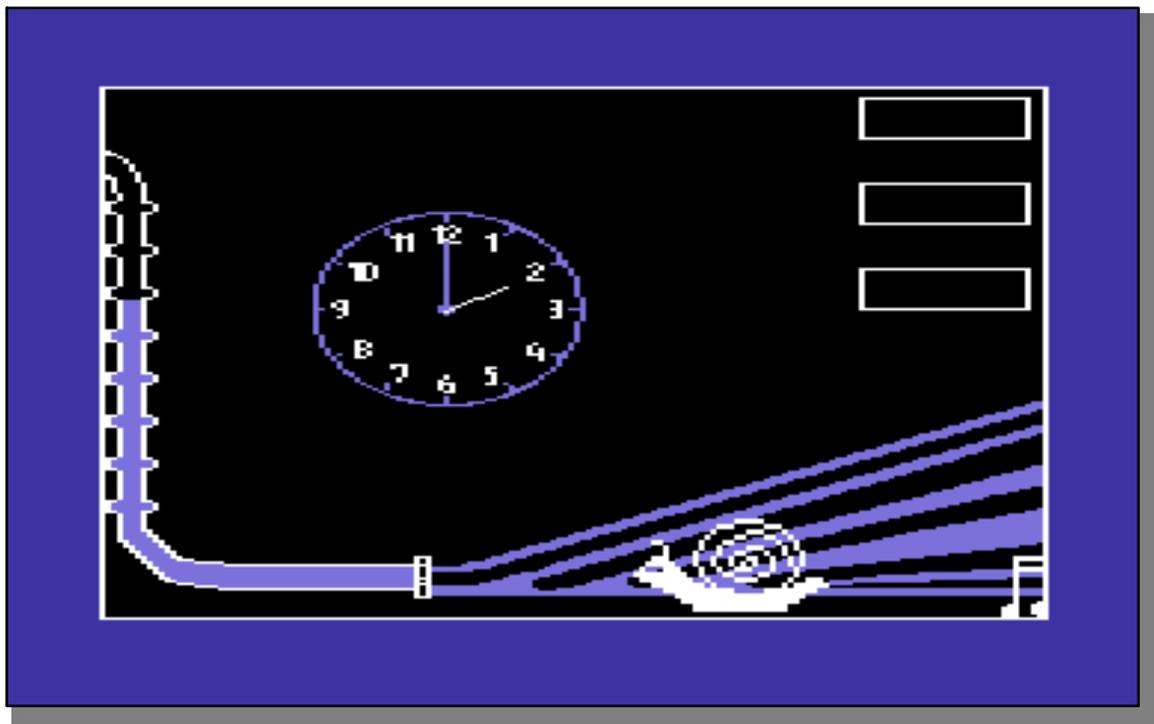
This activity teaches hours and minutes in 5 minute intervals. For each screen, a random time is displayed on the analog clock.

To the right of the analog clock is a multiple choice menu of different times in digital clock format, with only one that matches the time that is displayed on the analog clock.

For each screen, when the Snail Cursor is moved to the digital clock that matches the analog clock, and the [RETURN] Key is pressed, a light bulb on the left of the screen will be "turned-off". There are 10 light bulbs.

After the first 10 screens ("level 1") all the light bulbs will be turned-off and the Snail will take a "bite" out of the apple on the lower left of the screen, and an angry worm will appear from the apple, music will play, and all the light bulbs will re-appear.

After the next 10 screens ("level 2") all the light bulbs will be turned-off again and the Snail will take another "bite" out of the apple on the lower left of the screen, then the student will be returned to the Main Menu.



Aztec C for the Commodore 64 and the "What Time is it" source code

The TIME program for the Commodore 64 contains a completely rewritten English Language version of an application that I originally developed and published for the Apple IIe in 1991. The formal title of this program is "What Time Is It?". The goal of the application is to teach Elementary School Children how to tell time. Regardless, my own goal in producing this program was to provide a full-blown application that demonstrates what a seasoned expert Programmer can do in the Aztec C Environment for the Commodore 64 (C64). Now if all you are in this for is run programs and not to make them the TIME directory like all my other program directories contains the finished programs including a disk image called time.d64. These programs are preloaded on the disk image and run nicely in the vice emulator. If you are using these in WinVICE remember that while graphics load quickly in Warp Mode, the sound routines are time-based and you should take your emulator out-of Warp Mode for proper sound playback and time-based events.

Introduction

If you are reading this but have not reviewed all of the samples and all the projects in this programming environment please do not be offended if most of this goes completely over your head, or worse yet sounds like a nattering old man (which it should if it's flying high enough, since that was my intent). Take comfort in the fact that I am giving you all the source code and graphics images that I used to produce the TIME application. All this is completely transparent. There are no trade secrets here and no TIME deadlines so take your TIME and enjoy the confusion because it's part of the game. It feels so good when you finally understand, for a moment... 'nuff said. So let's get started and talk about the architecture and I am assuming that you have read the other make files and gained an understanding that memory management and mapping is "Job One".

Balancing the TIME Overlays

I really enjoyed this part. Some of you can imagine how I hurried through getting all my samples and library routines written and tested so I could actually do something more-or-less meaningful. I wondered myself at times whether I would be able to fit all the stuff I needed to into memory. My work on this particular application on the Apple IIe using Aztec C was in the PRODOS 128K environment which allowed me to store data in auxiliary memory. Even in that environment which is about the same size in conventional memory as this one, I needed to break down the Apple IIe version of TIME into more overlays than I did here and to keep the graphics data in upper memory... an option I don't have here. So when I did all this and it just barely fit I knew I could show the rest of you how far this can go without breaking, Like I said above; memory management and mapping is "Job One" and showing someone what can be done makes the job fun. You should have already figured-out from my other programs that I have been playing around with program memory holes, and sticking data into bits of memory that I couldn't run my programs in. This is the way we do it. You should also have figured-out by now that the linker outputs the code and data sizes,

and that my make files can be redirected to disk file so you can review errors and do something about them, or to examine actual data and code size and adjust the memory usage until there is barely any wasted space. When I do overlays I am always trading-off what to put into the main (root) module and what to link into the overlays. If the main module gets too bloated and the overlay wipes-out the upper ram the program will finish but never get back to the BASIC prompt. My goal is always to get the user back to the OS whether I program for Windows, Linux, MS-DOS, Apple IIe ProDOS, or the C64. One must therefore consider all the overlays at once and realize where one must put what and when to discard and when not to. In making TIME fit into SPACE I created some pretty neat tools to help save memory, and kept the code that I could in the main module, and the code that I couldn't in the overlays. I run-length encoded my graphics and re-used buffers as well. The C64 disk drive is notoriously slow, so it is bad enough that I needed to read these overlays from disk but it would have been much worse if I needed to read additional graphics from disk, or worse yet, to read more overlays (like I needed to on the Apple IIe). My comments are pretty good in the TIME modules, but sorry kids, they are intended for experts. So become one yourself if you aren't already and read the code and the MAKEFILE and run the thing and have some fun with it. 'Nuff Said on the balancing of overlays... some additional eulogies on the topic are in the TIME source code.

Using The Graphics and Sound Routines

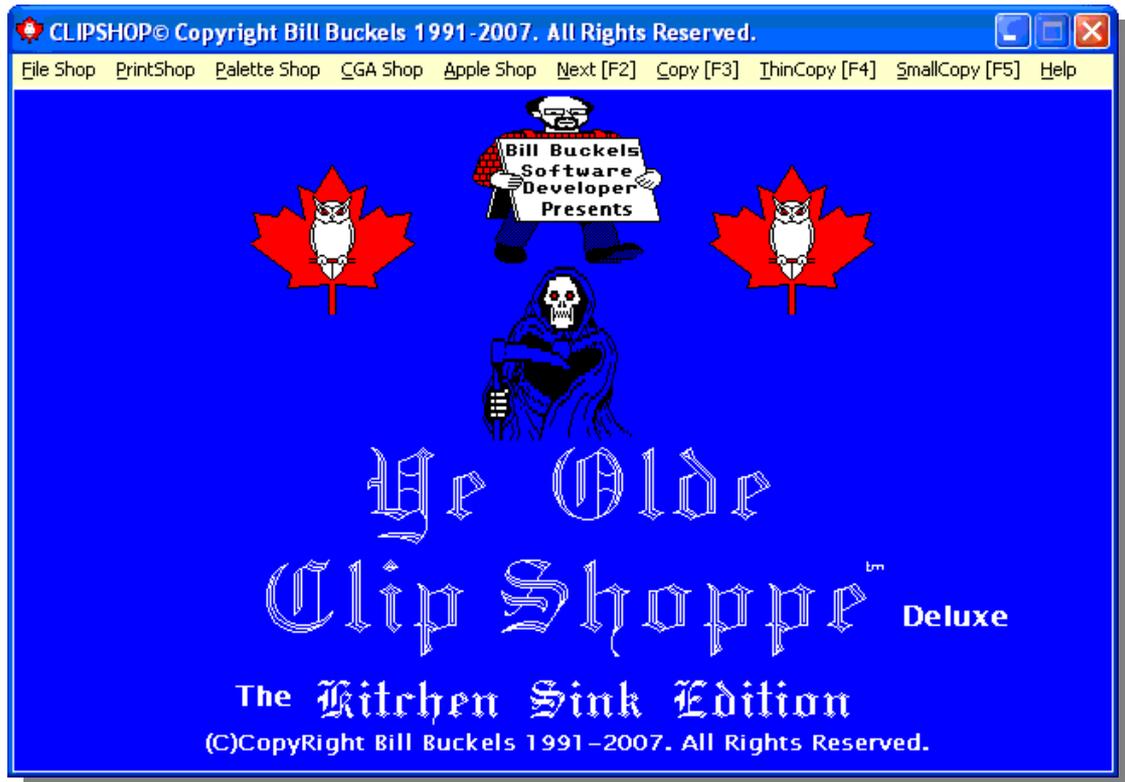
If you have looked at my graphics routines you know by now that I could have gone much further, but that was never my goal. I just wanted to have a little fun with this old compiler and show-off a little bit and leave the field wide-open for competition. I also could have optimized a little further, or a lot further. But I already know where all this breaks and by now I think I got some of my money's worth back on my original purchase price in enjoyment. Part of my goal was to showcase the B64NAT.LIB in all of this. I think that the C64 routines that I am leaving you with are better than what I did on the Apple. The sound is better of course and the game play is without doubt the same since the processor is the same. The memory saving technique of run length encoding can be taken further, but I kept it simple. Still, by comparison the C64 stuff is easily as robust as the Apple IIe equivalent of using bitmap graphics from upper memory. I am satisfied and can now leave this alone. So by all means change the code, and have some fun yourself. My selection of multicolour mode was deliberate. The graphics are simple since I originally drew them for the Apple II to avoid aliasing which effectively puts them somewhere close the same coarse resolution as on the C64. There are strange considerations when comparing low resolution graphics on the IBM-PC and Apple IIe and C64 which I have discussed in length in my Wikipedia articles etc. But the common denominator that I mostly went with was 4 Colors, and 2 of them being Black and White, with the other two mainly Red and Blue, and I will leave the rest for you to figure-out on your own. Read the code, run the program and have fun!

Over and Out.

Bill Buckels February 25, 2008

Ye Olde Clip Shoppe Deluxe

ClipShop© Copyright Bill Buckels 1991-2007. All Rights Reserved.
Version 2.0 (Win32 Release for XP)



ClipShop2Setup.exe - ClipShop Installation

www.mts.net/~raek/ClipShop/ClipShop2Setup.exe

VgaFanSetup.exe - Complimentary XP ScreenSaver Installation

www.mts.net/~raek/ClipShop/VgaFanSetup.exe

About ClipShop

Overview

ClipShop is a "Legacy Graphics" utility program which allows you to view, edit, save and print Old PrintShop, PrintMaster, and PrintPartner ClipArt.

ClipShop also supports viewing, editing, saving and printing of other older graphics and text file formats including BSaved, PCPaint Pictor PIC, PCX and BSV.

Apple II HIRES Images and many Commodore 64 Image Formats are supported as well.

The currently displayed ClipArt or graphic can also be copied to the Windows Clipboard to be used in other programs, to compose a new screen image, or converted to other formats like Windows BMP, Pictor PIC, PCX, or BSaved.

You can choose a copy command from a variety of 3 clipboard copy commands. Each of these 3 commands are used to get different proportions for the appearance of the ClipArt or graphic. Saving to BMP or other formats also supports this feature.

ClipShop supports both page size and "screen-scale" printing and Clipart catalog printing and saving to BMP is also provided.

Online Help

ClipShop's help file is extensive, and you will have no trouble finding the help you need to change colors in a graphic, or to use the many other graphics editing options that ClipShop supports. The online help should provide more than enough information to get you started.

Extras

ClipShop comes with a large collection of old ClipArt and many sample graphics.

ClipShop is distributed with authorized versions of PCPaint, the very first Paintbrush program for the PC (by John Bridges), which can be optionally installed to run under Windows XP, and many other extras.

The installation of Legacy Applications (PCPaint, etc) is optional. During testing of ClipShop, I have found that some hardware platforms are unable to run CGA Mode DOS Legacy Applications without using DosBox.

History and Development

ClipShop was originally written by me (Bill Buckels) in 1999, and was reworked as a Win32 application with support for long filenames in 2006 followed with significant functional and content additions and improvements.

When ClipShop was first released, its use was limited to supporting Legacy ClipArt in Windows. ClipShop was expanded a great deal for the 2007 Windows XP rewrite (ClipShop Deluxe), including support for other Legacy Graphics formats.

ClipShop now supports the features that we have come to expect in a Windows XP application, including File Associations, ShortCut Menus, and integrated printing functionality.

This release of ClipShop includes the following Applications previously released by ClipShop's author or the (expanded) functionality of them (and much more):

ClipShop Version 1.1 Old PrintShoppe ClipArt Utility

XSHOP Old PrintShoppe Clipart Converter

ICONDRAW - Old PrintShoppe Graphics Editor
ShopSave Graphics Plaything - Old PrintShoppe

ClipArt SDK for DOS

WINBMP - DOS BSAved Text Screen to BMP Conversion Utility

WSQUASH - BMP Image Scaling Utility

CGA2BMP - CGA Graphics Conversion to BMP Utility

STIX - Children's Sticker Draw

ClipShop was written in C and compiled under the Visual Studio 2005 command line using the Win32 SDK for Windows XP. The main application's Source Code is not bundled with the application since it is not for distribution. The source code for several of the programs that come with ClipShop is included with ClipShop.

Effective July 15, 2007 ClipShop has been "Officially" released and may be publicly distributed.

Thank you "Offical Beta Testers" for taking time to help me with getting ClipShop ready for release.

Find out more about Bill Buckels

General

<http://www.clipshop.ca/>
<http://www.clipshop.ca/DiskImages/>
<http://www.aboutus.org/Apptius.com>
<http://www.aboutus.org/ClipShop.ca>

Wikipedia

http://en.wikipedia.org/wiki/BSAVE_%28graphics_image_format%29

C128 Alive C64 Article Links

Windows User

<http://landover.no-ip.com/forums/index.php/topic,317.msg3685.html#msg3685>

<http://landover.no-ip.com/forums/index.php/topic,2203.msg8795.html#msg8795>

<http://landover.no-ip.com/forums/index.php/topic,2203.msg8797.html#msg8797>

BSaved Text Screens

<http://landover.no-ip.com/forums/index.php/topic,203.msg8798.html#msg8798>

<http://landover.no-ip.com/forums/index.php/topic,203.msg8799.html#msg8799>

<http://landover.no-ip.com/forums/index.php/topic,203.msg8800.html#msg8800>

<http://landover.no-ip.com/forums/index.php/topic,203.msg8801.html#msg8801>

Directory Lister

<http://landover.no-ip.com/forums/index.php/topic,1070.msg8802.html#msg8802>

<http://landover.no-ip.com/forums/index.php/topic,1070.msg8803.html#msg8803>

Compiler

<http://landover.no-ip.com/forums/index.php/topic,70.msg8808.html#msg8808>

CP/M

<http://landover.no-ip.com/forums/index.php/topic,2218.msg8796.html#msg8796>

<http://landover.no-ip.com/forums/index.php/topic,2219.msg8804.html#msg8804>

<http://landover.no-ip.com/forums/index.php/topic,2219.msg8838.html#msg8838>

<http://landover.no-ip.com/forums/index.php/topic,2219.msg8881.html#msg8881>

<http://landover.no-ip.com/forums/index.php/topic,2228.msg8878.html#msg8878>

<http://landover.no-ip.com/forums/index.php/topic,2228.msg8912.html#msg8912>

<http://landover.no-ip.com/forums/index.php/topic,2228.msg8919.html#msg8919>

<http://landover.no-ip.com/forums/index.php/topic,2231.msg8910.html#msg8910>

All C128 Posts

<http://landover.no-ip.com/forums/index.php?action=profile;u=262;sa=showPosts>

Commodore Free interview with Bill Buckels

Programmer and maintainer of the Aztec c museum



COMMODORE FREE

Please introduce yourself to our Readers

BILL BUCKELS

My name is Bill Buckels. I am a Canadian Software Developer. Rather than tell you about the larger aspects of my very good life, I would like to tell you a little about my life and times as a computer programmer.

I started my career in Software Development in the 1980's while working as a product designer with vector based computer graphics on "mainframe" Computer Aided Drafting (CAD) systems. Heavily influenced by computer magazine articles about graphics programming by other programmers of the day like Steve Rimmer (Computing Now Magazine) and the already well established PC Graphics program market, I moved very quickly to a new career as a professional Graphics Programmer.

The market for graphics programs was (and is) highly competitive and by the early 1990's it became clear that all but the niche markets for specialized graphics programs were filled with very good products, and not much more than an average living could be made unless one was both an exceptionally lucky or brilliant programmer or both. Being only average in both these areas, and after finding the same held true while trying my hand programming in the already established computer music and educational program markets, I moved to programming in the larger market of specialized business systems, which I still do professionally today.

From that time to this I have produced Shareware and Freeware in the areas of Graphics, Computer

Music, and Children's Educational Programs, which I do as a hobby as time permits. My latest efforts continue to focus in those areas, and in re-writing my older Shareware and Freeware for the systems of today. My recent efforts also include documenting the

almost forgotten systems of yesterday and providing the latest generation of programmers and users with articles about these old systems some which are posted on Wikipedia and other online forums.

CF. How long have you been working with Commodore Computers?

BB. I am relatively new to Commodore computers and the first one I saw was in 1983 when I was in my 30's when I sat down with one of my cousin's boys who was around 12 at the time and copied an example music program into their new home computer, a C64, then modified it to play a melody or two of my own. After a brief introduction to the computer he wandered off and did something else while I sat for an hour or so in silent and profound amazement at this new technology that was starting to make its way into many Canadian homes. Up to that point and from the time I had entered the workforce in the late 1960's the computer had evolved from the big institutions, academia and the banking industry to an affordable but still expensive proposition for small-er business for inventory and industrial production control. For the decade before the introduction of the C64 and the next five years or so that followed I never paid much attention to Commodore computers since my focus on the use and programming of computers was confined to business and industrial uses like the emerging

Computer Aided Design and Manufacturing (CAD/CAM) and automation technology that had taken over in the work place. Interestingly, by 1985 my mother who is 83 had purchased a C64 with all the amenities and completed a C64 programming course at one of our local Universities. I on the other hand was becoming quite proficient in the use and programming of IBM-PC's as well as of the still expensive mini computers that had already replaced the mainframes of the 50's, 60's and 70's. From my own perspective as well from that of family and friends in general, the use of computers was accepted by this time as part of an evolution of literacy that extended books and creative mediums like writing, art and design through the use of technology. My career through those years had moved from Industrial Sales which included much automation, to Mechanical Designer and CAD/CAM including that of trainer to pure application programming. By late 1988 my proficiency with computers and programming had led me to my present career of Software Developer and Business Consultant. I still program in the workplace daily and even at almost 56 years of age occasionally still will program around the clock if required by the project and the client. My passion for programming is still as keen as it ever was at that time, despite the fact that the computers and programming languages of today have become increasingly more sophisticated and complex.

However, from the time that I sat down with my young cousin at his C64 in 1983 I have always indulged myself in an activity that I like to call "Recreational Programming" which has resulted in my production of thousands of large and small software applications for many platforms including the C64, and Apple IIe, as well as the IBM-PC family of computers. In recent years I have resurrected many of my recreational efforts, some which were distributed in the 80's and 90's as Shareware and Freeware, and some that were written for "real markets" like Graphics, Educational and Music Programming. I often and have from that time to this used my recreational programming as "proof of concept" for commercial projects that I work on. The lines between "vocation" and "career" do not exist for me. I would always rather sit-down in front of my computer and program than read a book or watch television. Recently I have become quite infatuated with emulators like the excellent VICE emulator for the C64. The whole computer industry is moving to a general model of "virtualization" and what that means is that anyone in this business needs to educate themselves and stay informed in this area. I regularly use "industrial strength" virtualization as part of a "toolbox" of client deliverables, and incorporate products like Microsoft Virtual PC, and VMWare into those solutions. I also, as a "matter of course", and have for years, used cross-platform development environments and emulators which have existed long before I wrote my first computer program. I have never seen a time that such excellence exists in the "virtualization" area, and I subscribe whole-heartedly to one of the many little adages that we programmer-types seem to like; "Everything Old is New again!".

CF. How much documentation was there available on formats like Print Shop and Print Master for you to work from?

BB. There was never any documentation for these old formats, but they weren't all that mysterious either. From the time I got into computing I have deciphered machine language and binary file

formats. Like many in my business I am very astute at pattern-recognition and reverse-engineering. Since I understand how computer graphics work in general in a computer program deciphering a simple file format that is uncompressed like these Old Print Shop libraries is as trivial as tying ones shoelaces for someone like me. Consider that the Old Print shop graphics came in two parts (a paired library) on the IBM-PC and on other systems like the Apple II were not in libraries at all but in "Mini-Pix" (individual files with descriptive names). The Apple II could have long names but the IBM-PC couldn't so the designers of The Print Shop stored the long names in a sequential binary .NAM file of 16 characters in length for each name and stored the corresponding graphics at the same relative position in a sequential .DAT file. Since we all knew from using The Print Shop that the resolution of the graphic is 88 x 52 with a bit-depth of 1 (monochrome) it was not hard to use a calculator to divide the file size of the .NAM file by the 16 character name length to determine the number of graphics in the .DAT file, then to peer into both files to decipher the other binary parts and guess what they might be. Being a programmer, I simply wrote test programs to view these on my screen and through trial and (very little) error developed the necessary loader and conversion routines that I then released with example source code about 20 years ago. I have honestly lost track of how many times I have performed this same type of analysis, as I am sure many other programmers have, but that is the short version of how it is done!

CF. How long have you been programming in C?

BB. I started programming in C in around 1986. To that point I had programmed mostly in various BASIC's and specialized engineering languages similar to FORTRAN. However my work in programming already extended to programming in Assembly and pure Machine language. I didn't so much start programming in C as simply doing so. I am like many programmers not language dependent. My syntax and pattern recognition and logic skills' come naturally as do my musical skills and other creative skills.

One of my favourite quotes over the years that about sums this up is by Robert A. Heinlein; "A human being should be able to change a diaper, plan an invasion, butcher a hog, con a ship, design a building, write a sonnet, balance accounts, build a wall, set a bone, comfort the dying, take orders, give orders, cooperate, act alone, solve equations, analyse a new problem, pitch manure, program a computer, cook a tasty meal, fight efficiently, die gallantly. Specialization is for insects"

CF. What other languages have you worked with, can you give the pros and cons of each?

BB. At this point the number of computer languages that I have worked in and continue to work in fills a very large "toolbox" (or a "workshop" would probably be a better analogy). I am no different than most career programmers and we all (to varying degrees) have provided ourselves with the opportunity to continue to grow and expand into all available omniverses of language, syntax, and dialect. I could write many books on this and still not completely answer your question.

I have worked with a variety of programming languages like Assembly Language, BASIC, and C (AB and C) in many flavours and on many platforms.

My programming language skills extend from the days of the Mini Computers and the various Job Control languages to current .NET languages in Windows and Windows Mobile, and on the Linux Side and Windows Side both, C++. Even on older computers like the Commodore 64 and Apple IIe, I prefer to program in the C Programming language, primarily due to the readability and maintainability of C over both BASIC and Assembly Language, and C's general efficiency. Whenever I looked at languages like Pascal which is as easy learn as any other language, and DELPHI and other Pascal dialects, it was just to be proficient, and I translated everything to C or C++ for consistency if I needed to translate at all. I continue to use C code that I wrote 20 years ago in programs for today's computers will little if any changes, and I also will take large C++ methods and classes and translate them directly to the similar grammar of VB.NET to use in an ASP.NET application. It is all the same to me.

However, for someone who does not program in C, and who simply wants to have fun on the C64, BASIC is the easiest language to program in. However since programming in BASIC on the C64 can be quite "low level" to get-at the fun stuff like sound and graphics, generally speaking programming effort is equal between C and BASIC. If one then needs to augment BASIC 2 with machine language, C wins "hands-down" because native machine language can generally be mixed with C in the same program and with arguably the same learning curve, and C is for the most part portable and efficient except for these machine language pieces. C generally "abstracts" a program design into more readable and again arguably higher level concepts and control structures that are essentially unencumbered by clutter like line numbers. C is modular and lends itself to writing larger and more readable programs with more comments etc. It just depends on individual needs. The drawback of using C on the C64 is that all the references are written using BASIC and Assembly Language examples. Translating this into C becomes quite easy with practice but it does take practice.

CF. What's your favourite bit of old clip art that you've found and converted?

BB. The 3 disk set of Mini-Pix by Beagle Brothers for the Apple II that I translated to the IBM-PC in the early 90's was one of my favourites. My all time favourite is the clip art library that was produced by Don Joyce for the original release of PrintMaster. I haven't corresponded with Don this millenia. However the artist that produced perhaps the best clip art library of all time had this to say in 1999: "When I first came upon computers in the early 80s, I was impressed with the then current philosophical approach to creating graphics interfacing which suggested that the goal was to get the computer to act like your brain does, to be intuitively useable. Then this all reversed and didn't seem to matter anymore. Rather suddenly, we had to learn with difficulty art programs that had and have little to do with how our art brains want to work. When stuff like Adobe appeared, the goal was then to force your brain into compliance with distinctly unintuitive graphic processes. Computers were no longer trying to be like us, we were trying to be like computers."

CF.. Do you draw much yourself?

BB. I compose rather than draw on the computer mostly. I cartooned my way through most of my

school rather than pay attention in class and produced sales catalogue as well as engineering drawings in my CAD days in the 80's. It depends what you mean by drawing really. I also do freehand and scan the composition then clean-up in a paint program. I don't really separate composition from Art, and have written many interesting screensavers and other vector as well as bitmapped graphics programs which makes me a Graphics programmer rather than a User of Paint Programs. However my skill with the mouse is easily as proficient as most I have seen in my time in computing. I have produced countless drawings in both Analogue and Digital mediums in my lifetime.

CF. Can you briefly explain BSAVE (graphics image format)?

BB. Not briefly but I wrote a large article on Wikipedia which can be reviewed for more detail. A BSAVE Image (aka "BSAVED Image") as it is referenced in a graphics program is an image file format created usually by saving raw video memory to disk (sometimes but not always in a BASIC program using the BSAVE command). This format was in general use when the IBM PC was introduced. It was also in general use on the Apple II in the same time period. The Commodore 128 followed with the addition of the BSAVE and BLOAD Commands a short time later. On the IBM, BSAVED graphics and text images could be created for any video mode, with more complexity for the newer modes. On the Apple II and Commodore 128 BSAVED Graphics were generally all that was used. The BSAVED format is a device-dependent raster image format; the file header stores information about the display hardware address, and the size of the graphics data. The graphics data follows the header directly and is stored as raw data in the format of the native adapter's addressable memory. There is no file compression, and therefore these load very quickly and without much programming when displayed in native mode. No additional information (such as screen resolution, color depth and palette information, bit planes and so on) is stored. Video adapters were simple when this format was in wide use and the other information to load these could usually be inferred by programs that loaded these.

CF. What C64 applications do you really like?

BB. I am not so much into C64 applications written by others. We used to have a saying in programming that you are either into input or output, and that is the difference between programmers and other users like "gamers". Give me a good C compiler and an application to write and I am happy. I am currently happy with my Aztec-C cross-compiler for the C64 and with the WinVICE emulator, both which run under Windows XP and are not C64 applications at all. Not meaning to be egotistical (that comes naturally to a programmer) but I like the work that I did on the various C64 programs that I wrote for my Aztec C Museum Website.

CF. Do you follow the Commodore Demo scene, if so have you seen anything that amazed you and though "how was this done?"

BB. I was into the demo scene a little on the IBM-PC in the late 90's when it spilled-over from the Amiga along with the MOD files and I remember a really good demo by Future Crew that my kids enjoyed.

CF. Are you writing any other applications related to the C64?

BB. At the moment I am considering porting my Aztec C C64 compiler environment complete with all of my graphics and sound libraries to the C128. If I add additional functionality I will probably port whatever is appropriate back to the C64. My focus is not platform specific so much as providing specific functionality for various platforms. Having said this it

BB Yes, but generally in the areas of business rules and time estimates, but this is more project management than programming. For the simple programs that I write which is just about everything these days I start with a clean screen and write the program narrative. The architecture and everything in general conforms to standard patterns and strategies that are widely accepted in the industry and second



would not be hard to write most types of applications for the C64 using the Aztec C compiler that augmented with my many tools and routines. I am more concerned as well with adding support for additional C64 graphics formats to some of my other applications and tools for Windows XP and so forth. The simple answer is no.

nature for me by now. A plan like a road map is necessary if one is just visiting, but after one has driven the same road daily for so many years it is not. The user documentation is another matter and spend as much time on that and commenting my code as I do on the programming.

CF. Can you explain the history of Aztec C i see you have an online museum
<http://www.clipshop.ca/Aztec/index.htm#commodore>

CF.. I notice you use Winvice do you think emulation is important to keeping machine alive, which would you prefer an Emulation or a real machine?

BB Aztec C is a programming language environment for a variety of older now-obsolete platforms including the C64, MS DOS, Apple II DOS 3.3 and ProDOS, older Macintoshes, and Amigas. Manx Software Systems of Shrewsbury, New Jersey, produced C compilers beginning in the 1980's targeted at professional developers. Throughout the 1990's they continued to make their Aztec C. As their market share dropped, they tried to make the move to specializing in embedded systems development, but it was too late. They disappeared following the loss of market presence of some of their target platforms'.

BB I would prefer a real machine for using a program, but for testing and programming an emulator works best.

CF. Do you disassemble other peoples work to find out what makes the code work?

CF.. What do you think about the new hardware that is being developed for Commodore machines?

BB No. This is not usually necessary. I will occasionally disassemble portions of the operating systems etc. to determine how they work but the need to do so seems not to be present much anymore with so much information available on the internet. There is not really much mystery in how programs need to be written for me, or for many programmers.

BB Since I am not so much into gaming and even the Amiga I am not really interested. But having said that I am always keenly interested in virtual environments but with only so many hours in the day even I need to sleep.

CF. i spend so much time on the program design and flowcharting that once the actual programming or coding has to be done i loose interest in the project, do you spend much time planning and documenting?

CF.. Do you have a favourite machine?

No.

CF.. Have you any final comments to add, or is there a question you would have liked me to ask?

BB Yes Nigel. If you ever run across some software that will extend a 24 hour day to something longer like 48 hours please let me know. Then I might have time to play with all the new technology while I enjoy the old. I have lived from a time when blocks of ICE were delivered by horse-drawn wagons for ice boxes for those in my city who still did not have refrigerators to a time when computers respond to voice commands. I bought my lovely wife a robot vacuum cleaner to play with at Christmas time. What a hoot! And thank you for a most interesting interview.

COMMODORE FREE MAGAZINE
www.commdoorefree.com