

Commodore Free

Volume 3 Issue 2 February 2009

Free to download Commodore magazine
Dedicated to Commodore Computers
Available as PDF Text SEQ HTML and D64 image
www.commodorefree.com



CONTENTS

| | |
|---|---------|
| Editorial | Page 2 |
| News | Page 3 |
| RGCD Kickstart 16 cartridge | Page 9 |
| Interview with | |
| Cincinnati Commodore Computer Club | Page 10 |
| Simon Quernhorst | Page 12 |
| Simon Owen | Page 13 |
| GFX and bigger screen on the vic 20 | Page 16 |
| Understanding Vic 20 ram expansion | Page 18 |
| Commodore Computer club U.K. 2nd Meeting | |
| Shaun Comments | Page 22 |
| Nigel Comments | Page 23 |
| Peter Comments | Page 25 |
| In the beginning Part 12 | Page 27 |
| Commodore Computer club U.k | |
| Application form | Page 31 |
| Membership small print | Page 32 |

How to help Commodore Free Magazine

HOW CAN I HELP COMMODORE FREE..

Ok the best way to help would be write something about Commodore

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.

News

Club meeting

General Commodore news

New Products

Editor

Commodore Computer Club U.K. held its second official meeting and my goodness was it a meeting, for the readers who didn't attend this momentous event you were sadly missed; but more importantly you missed a meeting and a part of history. Attendance was better than the first meeting (with only 3 attendees) a total of 7 Commodore Computer Club Members turned up for the event. To find out more about the meeting and what went on you need to read the article in this issue, it's written as a personal event of the meeting.

Please remember Commodore Free is just that FREE, I have been approached to sponsor events, in fact 2 but I don't have any money as this is a free magazine, moreover as there haven't been any donations lately the website has been running "out of pocket". The Good news though is Commodore Computer Club U.K. has an official bank account, yes really we do have an account and can take cheques. So the club is officially now open and once again welcoming members, a membership form is attached at the end of this document also a link to sign up online, the PayPal account has proved a little difficult to setup but is in the progress of being completed, however as I am

Commodore Computer Club U.K. held its second official meeting and my goodness was it a meeting, for the readers who didn't attend this momentous event you were sadly missed;

treasurer you can send the PayPal funds to myself and I will transfer them to the club's official bank account, of course the Commodore Computer Club is also a "not for profit" organization with the funds used to hold events and to fund further hardware and software projects.

I also hope to add to Commodore Free magazine a list of items for sale, these will be sold with the profits going to the Commodore Computer Club. If you have an item of hardware or software you would like to donate to the club feel free to contact me and we can see if it is possible to collect the item for sale or reuse in the club. We will at further meetings have spare machines setup for demonstrations so users don't have to drag their machines with them if they feel or indeed don't have the transportation to do this.

I wish I could tell you about some of the great projects we have lined up in the Commodore Computer Club but as none members read this magazine I cant, so you will need to join up, I can tell you we will be having some Official Commodore Computer Club polo-shirts made these will be priced at £13 for none members of course it makes sense to be a member to wear one and if you're a member then you will receive a discount. The CCC U.K. website forum is however open to all and you are free to join in for the chats, normally held on Sunday at 8:30 GMT of course we chat about Commodore as well as the weather and how x or y did on strictly come dancing. I think its mostly about Commodore related items though, remember please these are friendly chats no "pistols at dawn" and no hitting below the belt.

I wish I could say the Official Commodore Computer shop is online however we are still

working with a solution to this as members receive discounts, we don't just want anyone to pose as a member enter a "borrowed" membership and take the discount, I know we are all honest and wouldn't do that but it makes sense to put the provision in place.

For myself it looks as though there may be an end to my suffering of back problems, I am now seeing an osteopath and so far the problems are subsiding the pain has eased, and I have more freedom of movement. I am still looking forward to a day when the pain finally goes and I can concentrate on working, keeping fit and Commodore computer activities.

I am still trying to put together an issue about CP/M and have been contacted by a member of the CP/M members in Germany and other CP/M users with some history, I have one particular user who has written a couple of pages about CP/M but I still would like more about how Commodore ties in with this history. If you are a Commodore CP/M user reading this I would love to hear from you, the forums seem alive at the moment with CP/M questions but no one seems to want to write anything about Commodore and CP/M.

Comments from readers have been a little thin lately so all are welcome.

Regards

Nigelp2k
www.commodorefree.com
www.commodorecomputerclub.co.uk



NEWS

FMV (Full Motion Video)

Written by Mathias "AmiDog" Roslund
<http://www.amidog.com/c64/fmv/>

Converter:

8bit greyscale input frames are cropped and or scaled to 80x48 pixels. These frames are split into tiles of 4x4 pixels and a simple histogram approach is used to select the 256 of the most frequently used ones while trying to avoid picking two very similar ones. During the second pass, mean square error is used to pick one of the 256 tiles for each and every tile in the frames. This results in frames of 20x12 tiles (240 bytes).

Packer:

The packer will try to compress each frame using a set of possible compression methods and will select one that gives the best compression. All methods are currently lossless, but that may change.

Currently supported methods:

- No compression (1 type)
- RLE (1 type)
- Delta (3 types)

Player:

The player uses the dream load loader system. The player currently only works on 1541 drives though, sorry about that. The player will start by loading the charset (tiles) and will then start pre-buffering. The buffer used is 16KB (64 sectors). The displaying and decoding is handled by an interrupt and a new image is displayed and decoded every second frame, giving an output of 25fps. In case the buffer runs empty, the player will wait for it to be filled to 1/4 (4KB) before playback continues.

Full screen playback:

If the fire button (port 2) is being pressed when playback starts (i.e. when the pre-buffering is done), the animation will be scaled to full screen while playing.

The player was developed using dreamass and VICE, but has also been tested on a real C64 with 1541C disk drive.

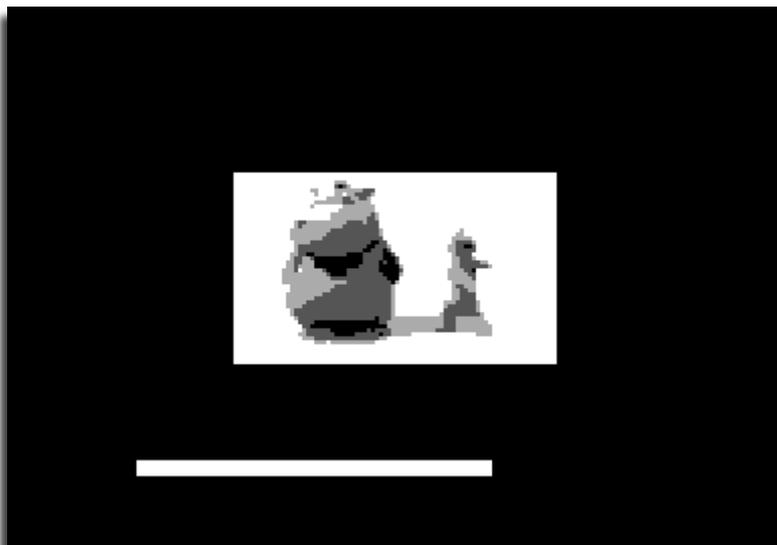
Example animation:

The famous lion_seul.mpg animation (1600 frames, 320x240 pixels) has been converted (cropped to 320x192 pixels and scaled to 80x48 pixels) to 1601 tile based frames of 20x12 tiles each (a black frame has been added to the end of the animation). The raw data (1601*240 = 384240 bytes) has been compressed to 81642 bytes using RLE (9 frames) and Delta (1592 frames) compression. This gives an average of about 51 bytes/frame, which seems to be about the limit of 1541 streaming using the current loader and compression methods, especially when using the full screen mode.

Future:

- Lossy delta compression
- Faster loader allowing higher byte rates
- Some kind of dithering
- Lower frame rate but higher resolution
- Lower frame rate but more colours

// Mathias "AmiDog" Roslund, 2008



High Voltage SID Collection: Update #49

Date: November 09, 2008

Resulting Version: 49
 Previous Version: 48

Hello fellow lover of SID music!
 Nice you found some time to read through this script, to see what has been changed in the HVSC and for what reason. After this update, the collection should contain 36,081 SID files!

This update features (all approximates):

- 1059 new SIDs
- 54 fixed/better rips
- 2 fixes of PlaySID/Sidplay1 specific SIDs
- 10 repeats/bad rips eliminated

- 431 SID credit fixes
- 232 SID model/clock infos
- 9 tunes from /DEMOS/UNKNOWN/ identified
- 40 tunes moved out of /DEMOS/ to their composers' directories
- 14 tunes moved out of /GAMES/ to their composers' directories

Download HVSC Update #49 from the usual address:

<http://www.hvsc.c64.org/>



HardSID 4U has been released

The HardSID 4U is the most powerful SID synthesizer since the legendary C64!

Features:

- Two CPUs and main memory
- USB connection (compatible with both 2.0 & 1.1)
- Isochronous USB endpoint for low-latency audio
- VSTi interface with 8000Hz update rate on all SID registers (free)
- Wave-in VSTi pin for routing 8000Hz signals to any registers (example: playing other VSTi's sound on the volume register)
- Full automation of all VSTi parameters
- Superior sound quality (...it is a HardSID!)
- Support for up to four SID chips (6581/8580/6582 in any combination)
- Updateable firmware over USB
- Microsoft Vista compatible (drivers for Win2000/XP/Vista)

..and if you're a C64 fan:

- Cycle-accurate playback of your favourite SID tunes
- Digitized sound + high-speed playback with low CPU utilization
- Seamless playback of .sid tunes while you work on your PC by providing a huge playback buffer for non-VSTi applications

The HardSID 4U Studio Edition:

- Full physical separation of the SIDs from the USB driven circuits
- 100% elimination of EMI (Electromagnetic interference) noise that may come from your PC via USB

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

Hi!

We're celebrating the 10 Year Anniversary of the HardSID project this year. We couldn't be happier, so we decided to give everyone our best special offer ever:

HardSID 4U for only 199 EUR instead of 299 EUR!
<http://www.hardsid.com/order.php>

Regards,
The HardSID Team
<http://www.hardsid.com>



SIDPLAY 4.0 has been released

SIDPLAY 4.0 is a Commodore 64 music player for Mac OS. It emulates the sound chip and other internals of the Commodore 64 home computer to play back Sid. This is the biggest version of SIDPLAY yet, and has been completely rewritten from scratch, designed as the ultimate SID music player for the best operating system.

SIDPLAY 4 boldly goes where no other SID music player has gone before.
- Requires Mac OS X 10.5 or higher, runs on PowerPC and Intel-based Macs.

- Support for HardSID and Catweasel cards is not included in this version.

Check for the list of new features at:
<http://www.sidmusic.org/sidplay/mac/>



ACID 64 Player 2.4.0 has been released

ACID 64 player is a cycle based Commodore 64 music player designed for playing SID tunes on sound cards and devices that have a real SID chip on board. A new version of ACID 64 Player has been released with the following improvements:

- Emulation improvements

- STIL info can now be retrieved for a SID file if it is not played from the default HVSC location

- When HardSID 4U device is turned off while ACID 64 Player is running and turned on again, playback can continue after song restart

- Opening SID files via Internet Explorer is now possible. This makes it possible to play SID tunes via the online SID search of www.hardsid.com.

<http://www.acid64.com/>



Digital Talk #84 released

The new issue of Digital Talk #84 has been released, for German readers. You can find the issue here <http://noname.c64.org/csdb/release/?id=72913>

Plus/4 (c64) Ultimate 264 Map

Ultimate 264 Map - This is the Ultimate Map from SVS. This Excel file contains the complete OS ROM of the Plus/4 (and C16), a large TED Registers list, graphic plans of all allowed text and graphic modes, and a table of Plus/4 bugs. This Microsoft Excel file can be also be opened with Open Office without problems and is available from here http://plus4world.powweb.com/tools/Ultimate_Map

Remember is back with a new disk

After a long break, Remember are back with more disk releases. The disk features 5 new releases (one is an unofficial release) and 2 re-releases.

New releases are (374-377):

- Archipelago
- Gaplus
- Blueprint
- Amazon Warrior.

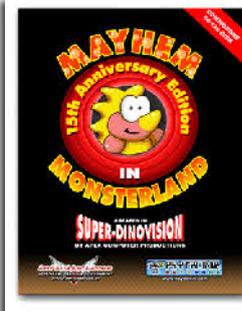
Re-releases are:

- The Master Of Magic (15) - some bugs removed.
- California Games (125) - sound bug removed and one-sided.
- Unofficial release is Micro Painter.

http://www.allgamebase.info/rem/_Rem_Nov08/REM_Nov08.zip



Psytronik Software ON DISK



At long last the Psytronik Software releases are now available on disk. Each Psytronik deluxe disk edition is packaged in a box that features full colour artwork - specially designed to look great in your retro collection! The floppy disk is presented in a full-colour glossy disk-sleeve complete with colour label. Instructions for each game are also included on a separate instruction sheet.

Psytronik Software in association with www.infinitelives.tv is extremely proud to present MAYHEM IN MONSTERLAND - the incredible cartoon platform classic from Apex Computer Productions!! To celebrate the 15th anniversary since Mayhem was unleashed into Monsterland we've teamed up with Apex in order to present you with this special anniversary edition of Mayhem In Monsterland (Deluxe Edition) PRE-ORDER [C64 Disk]€12.99 anniversary release of the stunning C64 game.



Commodore Plus4 JAVA emulator

The Commodore Plus4 JAVA emulator has been updated. You can get it from here <http://minus4.plus4.net>

"what's new.."

- Updated rendering engine ported from Minus4w, should handle more cases.
- Updated CPU processing ported from Minus4w. More games should run.
- Fix a crash in the CPU processing.

- Changed the colours to what Yape uses (much nicer)
- Added FLASH attribute, so things now...flash. (Manic Miner keys, Monty pick-ups etc.)
- Fixed the rendering frame skip - it was always skipping at least 1 frame Fixed the overflow flag (V) on the ADC instruction. - Mercenary now works!
- Added the new "joy" parameter
- Fixed commando - added several undocumented NOP codes.
- SOME vertical scrolling games work.
- FPS locking Timer works properly
- Frame skip is now much better and helps when screen is scaled.

64Copy v4.4 released

(Peter Schepers) has released 64Copy v4.4. The update has been worked on for over a year, Peter thought that he had fixed enough small bugs and added enough enhancements to warrant a final, non-beta, release.

The basic goodness is thus:

- 6502 Disassembler fixes
- HTML HELP file
- CRT module extraction
- Many other small enhancements

Download it from here

<http://ist.uwaterloo.ca/~schepers/DOWNLOAD/64COPY44.ZIP>

Commodore 64 Wikipedia

<http://commodore64.wikispaces.com/>
Welcome to the Commodore Programming Wikispaces.
Here you will find titbits of programming info on the Commodore 64 and 128 computers.

Purpose

The goal is to provide an easy to use reference for various programming topics. Much attention lately has been given to new hardware devices, with actual software development fading away. The goal of this wiki is to reignite interest in actually writing programs for these machines.

H & D C16/Plus 4 Handbook

-
Volume Issues 1 to 10
Volume 2 issue 1
Available to view online, apparently these are quite a rare find for Commodore 16 owners, and were uncovered in a box of items donated to the Commodore Computer Club
www.commodorecomputerclub.co.uk

http://www.commodore16.com/index.php?option=com_content&view=article&id=237:-h-a-d-c16plus-4-handbook&catid=57:magazinesfanzines



Enforcer 2 preview released

Enforcer 2 Level 2 Preview

http://noname.c64.org/csdb/getinternalfile.php/68893/Enforcer_2_Level_2_Preview.zip

Enforcer 2 Level 2 Preview

It still contains 1-2 bugs and might judder sometimes, but otherwise the level is like 98% finished. Only the final boss can not be killed yet. This preview is looking amazing and is totally playable, Personally I feel the game is less confusing than Metal Dust as the foreground and back ground are easier to pick out, I am sure the hate mail will flood into my inbox now

Commodore 128 Basic 7.0 One Liner Competition

<http://landover.no-ip.com/forums/index.php?topic=2553.0>
We are pleased to announce the Commodore 128 Basic 7.0 One Liner Competition 2008!

Commodore 128 Basic 7.0 One Liner Competition is a friendly competition designed to see what ingenuity exists in the C128 community.

RULES

- Entries must be original
- The result must be displayable on a real C128 (or VICE)
- The competition starts Thursday December 18th 2008
- Entries must be submitted on or before March 18th, 2009

- there will be a competition page at this site with the submissions & you'll have the opportunity to vote for your favourite submission
SUBMISSIONS
- Submissions must be made by email to [landover \(at\) tpg.com.au](mailto:landover@tpg.com.au)

PRIZES

- Prizes will be awarded on total votes & will be paid by PayPal
1st prize \$AU50, 2nd prize \$AU30, 3rd prize \$AU20

DISCUSSION

Competition discussion only in this forum: <http://landover.no-ip.com/forums/index.php?board=21.0>

QUESTIONS

Q. You state Basic 7..does that mean that SYS calls to ML are out?

A. If you mean creating a separate ML program & then calling it via SYS, then yes, it's out. SYS to a system function is ok. Lance

SEUCK website updated

Merman

To celebrate Christmas, over at the SEUCK Vault you can now:

- download the first game from our new Amiga Archive
- pretend to be Ash in Evil Dead II - Return of Ash
- play new C64 game Cipher Patrol, and Mark Chenery's Street Slaughter
- read about an amazing SEUCK remake
- and find out more about the SEUCK Redux game engine

All this and more at <http://www.seuckvault.co.uk>

Aztec C Museum

<http://www.aztec-museum.ca/>

A long overdue rework of the website, files added, etc with much help from Mike T. and others. Please see the acknowledgements for more about who helped.

Please report any broken links.
Bill

VICE 2.1 released.

VICE is the Versatile Commodore Emulator, it emulates the Commodore C64, C128, VIC20, PET, PLUS4 and the CBM-II, as well as the C64 DTV, and it runs on Win32, Mac OS X, Unix, DOS, RiscOS, OS/2, BeOS, QNX, SkyOS, AmigaOS and GP2X systems. VICE is *free* software released under the GNU General Public License, and as such it comes with full source code.

Most important changes since the last version include:

* Changes in VICE 2.1

** General

- The VICEplus project has been joined with VICE. Thus, x64dtv is now part of VICE.
- We can be contacted via IRC: #vice-dev on freenode
- Added a more accurate ReSID engine using floating point math (ReSID-FP).
- Added support for the USB HardSID.
- PAL emulation has been rewritten and optimized.
- Fixed the ACIA 6551 emulation. (x64/x128/xcbm2/xpet/xplus4)
- Monitor commands help text is now translated too (in ports where translation is available at all)
- Monitor IO command now displays IO area even if it is currently banked out.
- GCR file handling (.G64) issues an error message if the .G64 does not have the expected structure. Before, VICE just silently ignored such files.
- IRQ handling fix

** C64/C128 changes

- Fixed the digimax sound generation.
- Added the RR clock port disable functionality at SDF00.0. Allows to disable the RRnet.
- Improved REU compatibility and timing.

** C64 changes

- Added isepic cartridge emulation.
- Added Double Quick Brown Box cartridge emulation.

** VIC20 changes

- Improved the sound emulation.
- Fixed the lowest note bug.
- Improved VIC emulation (exact in-line colour/reverse mode changes).

** C64DTV changes

- New emulator.

** Unix changes

- XRandR full screen implemented
- Command line option '-full screen' is supposed to do something useful.
- Vsync code imported from win32 (based on OpenGL extension). see also doc/openGLsync-howto.txt.
- Vidmode (full screen support) is broken and therefore marked as deprecated. It will be removed in the next release if no-one takes responsibility to fix the broken code and is willing to maintain the code.
- PAL Emulation (new implementation, improved speed) should be usable again (it was broken on certain display depths).
- Log messages are always English.
- Only x11 keymaps will be installed when doing a 'make install'.
- MIDI cartridge emulation for x128, x64 and xvic.

** MS-Windows changes

- Watcom generated executables are more optimized.
- MIDI cartridge emulation for x128, x64 and xvic.
- When specifying the target for the RS232 communication (Menu Settings/RS232 Settings), the target can be specified by name now. Before, you had to use the IP address.
- Fixed some minor errors with the RS232 communication via TCP connections.

- RS232 can use real devices now, too. To use this feature, you have to specify the COM port used in Settings/RS232 Settings like: \\com1: baud=57600 parity=N data=8 stop=1

The characters after the colon can be used to specify additional settings for the COM port. The string must have the same form as the mode command's command-line arguments:

```
[baud=b][parity=p][data=d][stop=s][to={on|off}][xon={on|off}][odsr={on|off}][octs={on|off}][dtr={on|off}][rts={on|off}][tg]
[idsr={on|off}]
```

The above example specifies COM1 with a baud rate of 57600, no parity, 8 data bits and 1 stop bit.

Note that the leading \\ is used to distinguish between TCP/IP targets and real devices, thus, you have to use it.

(Sorry, no fancy UI yet.)

- vsid GUI reworked:

- * infoline appears only by pressing "I"
- * raises process priority. Otherwise, when in background, loses CPU power and music is not played at correct speed
- * drag & drop support; window also gets focus back on drop
- * correct display of emulated sid model (when using -keepenv)
- * changed colours

- fixed copy function from the VICE monitor window.

** AmigaOS changes

** OS/2 changes

- Fixed 24/32bit display depth problem.

** GP2X changes

- Fixed black screen problem.
- Pause emulation while the menu is open.
- Added experimental support for USB keyboard and joystick.
- Removed vsid.

** Mac OS X changes

- x64a.app and x64dtv.app (Cocoa) are now feature complete
- * added flip lists
- * added netplay dialog
- * added ROM resources
- * added video settings dialog
- * added record snapshot dialog

- Introduced new control window that replaces old peripherals drawer

- * with tape control buttons
- * attach/eject buttons for each drive
- * drag & drop of disk/tape images

- Introduced Cocoa compile for Mac OS X 10.5 and greater with more features (see extended Resource Inspector)

- Completely rewritten joystick driver that supports user-definable buttons for fire and all directions.
- Copy & Paste support for emulator and monitor/log window
- Added x2, x3, x4 view scaling like in QuickTime Player
- Store visibility and size of all windows in user's preferences
- Added Keyset Joystick toggle to quickly disable keyboard joysticks
- Added MIDI support

For more information check out the VICE home page at:

<http://www.viceteam.org/>

At the time this was written the VICE home page was not updated yet, rest assured that it will be updated soon. In the meantime, please use <http://vice-emu.sourceforge.net/>

The VICE team.



Cronosoft in financial trouble

<http://www.worldofspectrum.org/forums/showthread.php?t=23239>

It's a bit of bad news, but I'm sorry to say that I'm probably closing up the Cronosoft label.

I'm struggling financially at the moment, and Cronosoft has too, for a while without much interest, and I'm beginning to lose money keeping it going, plus I can't afford to buy in more tape stocks etc. Also, unfortunately, I've had a hefty PayPal chargeback against someone I sent a large order to that is claiming that it's undeceived.

Of course, all payments to authors etc. are all put aside and will be sent out as soon as possible. Maybe I'll be able to restart it in the future when times are a bit better.

I'd like to thanks everyone that's supported the label, either by buying games, writing them, designing covers, giving us invites to shows etc..

I'm really sorry - thanks again.

--- WWW.CRONOSOFT.CO.UK ----

===SNIP=====

thanks for your kind support. I will be able to continue to offer emulator versions of course, but the tapes idea was the main reason for continuing.

PRESS PLAY ON TAPE

updates of what is happening in PRESS PLAY ON TAPE

First of all don't miss out our upcoming concert in Copenhagen at the usual place: The Rock! It happens on March 21st. It was previously announced to be late February, but the venue got a chance to book Napalm Death so the concert was moved. Don't miss this one. We are preparing for a blast of a concert: Keep an eye on our website for updates. Buy your tickets now before it's too late at www.billetlugen.dk:
<http://www.billetlugen.dk/redirector.php?id=12737&i=1>

We have just released our latest show-starter video on YouTube. We have started a few gigs with this one and wanted to share it with the rest of the world. Yeah, the one with the girl in the bikini and with all the Commodore stuff. You can mail it to all your friends to show how cool and hip you are that you know where to find such videos: <http://www.youtube.com/watch?v=aoQa1OiHlRM>

Also, we have now added a DISCUSSION FORUM to our website! Come join the discussions and let your voice be heard: <http://www.pressplayontape.com/?pid=forums>

Last but not least we are now FINALLY recording a new album! Right now we are rehearsing to make the arrangements for the tunes

Maybe I could do a 'download the inlay and print yourself' kind of thing. I do have some stocks left, so I will carry on for a little while yet until they've gone. The website is still there, and It'll be tweaked a bit. Cronosoft will still be there as a name if not as a supplier of tapes.

Sales have dropped a large amount over this last year, with some being in just single figures, so maybe it would be best to 'come back' in the future when things are a bit better. The charge backs thing is a new problem to overcome. Sending things by recorded delivery adds a lot to the cost which people don't want to pay (especially for international which sticks €3.50 onto the cost, coupled with the fact that lost items don't have a proof of value which satisfies the post office, unless it's the games that are sold on Ebay). The last chargeback was for over €30, which was a bit of a stinger. It wouldn't have been so bad if the person concerned had even emailed to say they hadn't received the games.

I'll keep posting though... don't worry I've posted so much nonsense on here, there's no reason to stop now hahah!

WWW.CRONOSOFT.CO.UK

EDITOR COMMENTS

I suppose its all down to if you buy the games the label will remain viable if you don't purchase goods then the label will fold the Commodore Computer Club U.K. are working with Simon to try and provide some assistance

and we will record them during the coming months. We won't set a launch date, but rumour has it that PRESS PLAY ON TAPE has a 10 year anniversary at the end of the year so who knows what will happen then!

Thanks for your consistent support! Hope to see you at The Rock! (Come talk to us after the show and buy us beer -- we usually stick around after the concert ;-)

Regards,
PRESS PLAY ON TAPE



Kickstart C16 & Invasive Action Cartridges [C64]

<http://www.cosine.org.uk/>

Reviewed By J. Monkman <http://www.rgcd.co.uk>

<http://www.rgcd.co.uk/shop/reviews/kikstart-invasive-cart-project/>

Sometime back in the end of 2007 I emailed T.M.R about the possibility of creating 8KB cartridge versions of some of the smaller games in his highly acclaimed Commodore 64 back-catalogue. Naively, I was under the impression that it would be a simple job of saving each game executable with a special header, then burning the resulting files onto EPROM chips, fiddling about a bit with PCB kits and cartridge cases and presto! A new C64 cartridge release would be born. In hindsight, how wrong I was...



Initially poo-pooing the idea, T.M.R explained that it would be a little more complicated than I'd optimistically hoped and essentially involve re-visiting a lot of old code and relearn-

ing how cartridge loading works. However, I continued to express interest in the concept and persistently pestered him on and off for a month or so until one happy day, I opened my email inbox to find two emulator-friendly cartridge images waiting for me. This is where the fun really began.

The next step was to find someone to help manufacture the carts, a process that T.M.R still believed would be too expensive to make the project viable even on a cost-price basis. After a lot of fruitless searching (and money wasted on foolishly trying to assemble carts



myself) I eventually stumbled upon C64 enthusiast Tim Harris via his eBay Commodore Store; finally it looked as though the project would yield results.

The first game we collectively worked on was Kickstart C16, which itself was a source of many confused emails due to the fact that Kickstart C16 is a Commodore 64 port of the Commodore 16 version of Kickstart, and "not" a Commodore 16 cartridge. Using a standard black cartridge case and box artwork designed by T.M.R and myself,

early prototypes were sold via a forum post at www.retrogamer.net and proved to be surprisingly popular - even with RGCD's own Sir-Clive, who immediately voiced his regret at giving the game a 'so-so' review score back in RGCD #03 and gushingly praised T.M.R for his superb work.

At this point, we began work on Invasive Action, a quirky little arcade shoot-em-up developed by T.M.R under his Backward Engineering moniker and based on 'Ivasive Action', a game featured on the infamous Cassette 50 C64 compilation back in 1984. Now, before you write this one off ("Cassette 50? WTF! ROFL!", etc.), it has to be said that T.M.R's unofficial sequel is a pretty good laugh. It received a warm write-up in the now sadly-defunct C64 discmag Game Over(View) and is certainly a more than worthy addition to your C64 collection. In fact - and the same applies for both IA and Kickstart - considering the 8KB limitation, both releases compare more than favourably against the official Commodore library of 8KB cartridges released back in the format's popular years. However, to make things a little more interesting (and indeed saleable) Tim Harris suggested using transparent cases with the addition of an internally mounted blue LED - and if you look at the photos you'll see that the result is pretty eye-catching when the cart is plugged in!

Hype aside, I suppose that the real question is "are they actually worth buying"? The items are available priced at a very reasonable £13 each, exclusively from the RGCD shop. So in my opinion I would have to say "yes", but then I'm biased because this was my project to start with! However; it should be noted that both games are readily available for free in their original format (and are includ-



ed with RGCD #05), meaning that buyers are not really getting any exclusive content, but the fact that these are strictly limited, real hardware releases mean that they'll appeal to most C64 hardware enthusiasts and collectors. Both Kickstart C16 and Invasive Action come presented in stylish hand-made cardboard boxes with either printed or in-game instructions, and the latter game's luminous attributes make it something quite special.

In conclusion; the really exciting news (for me at least) is that there are likely to be more RGCD C64 cartridge releases to come in the near future - I've already opened discussions with several interested developers and there's even the possibility of a C64 16KB cartridge development competition financed by RGCD in 2009, which (if it goes ahead) will hopefully result in a slew of releases and breathe new life into the much underrated commodore cartridge format.

(Please note that both Kickstart C16 and Invasive Action are PAL/NTSC compatible and work on the C128 as well as the C64. If you are a developer interested in working with RGCD to have your own game published in cartridge form then please feel free to drop us a line via our contact page).<http://www.rgcd.co.uk/contact/>

Cincinnati Commodore Computer Club

<http://www.geocities.com/c64-128-amiga/>

COMMODORE FREE

Please introduce yourself to our readers



DAVID WITMER
My name is David Witmer, I am the President of the Cincinnati Commodore Computer Club. My day job is a computer desktop repair technician for a national computer chain store. Repair certifications I hold include Apple Desktop, Dell Desktop and Laptop, Toshiba Satellite, Lenovo/IBM Desktop and Laptop. I graduated from the Ohio University with a Bachelor of Business Administration, with emphasis in Organizational Communications. Scuba Diving is another of my hobbies, and I hold a certi-

fication in Advanced Open Water, and a variety specialties including Rescue Diver which is my most recent training. I also like photography and 78rpm record collecting.

CF. Can you tell our readers about your first Computing Experiences

DW. The college mainframe computer the IBM 370 was to give me my first exposure to computers and programming. The programming I did learn officially was SPSS, COBOL, and unofficially BASIC (and hacking the mainframe via a dialup terminal my roommate brought home from work-study at the Engineering Department at the university).

My girlfriend at the time was a Computer Science Major. I failed to understand what the big deal of computers was going to be farther down the road. Often we would study together when we had the same classes, but she did get one class that I decided to learn on my own: BASIC. I went to the computer lab to watch her do her homework on the monstrous sized equipment that was really more than a stand alone terminal to the mainframe. What brand of computer this was escapes me, but I remember her bringing the huge 8 inch floppy disks. On these she stored her homework, not on punch cards as I was doing. That was what intrigued me, that information could be saved on something other than a mainframe computer's hard drive, or a stack of cards. This was something I had to look into, but didn't really want to spend the time going through a class. In retrospect, I probably should have taken that class for the credit of learning basic, and to learn better programming skills to boot.

I bought one floppy from the College bookstore to hold my programming examples, figuring I'd never need more than that, I didn't. I learned the basics from my girlfriend like how to format the disk, how to erase the current program in memory without turning off the machine, probably her instructions to me were simple, but also helping her to learn to make sure she understood what she was being graded for! I followed along some of the programming examples she had from her class, then decided I wanted to do something fun. So I found a type in program for a game called WUMPUS. Yes, I typed it all in from BASIC, on a computer that wasn't mine, just to learn the fundamentals of BASIC. When I finally got it all typed in and tried to run it, of course it had some errors from mistyped commands, but troubleshooting didn't seem any harder than the other computing classes I had taken. I finally had the program debugged and played several games of WUMPUS without any errors. I felt so accomplished.

My first home computer purchase didn't happen until after I graduated from college, on September 27, 1982. I had a friend from my high school days that owned a Vic20 for several months, and he allowed me access to it to play around on and see how much fun owning a personal computer could be. So, the two of us decided to take a trip to Columbus Ohio and buy my own computer setup. I knew I wanted a Vic20, and a tape drive, so the dealer at Tech HiFi didn't have a hard sale trying to get me to buy anything. He did try to get me to buy some programs, which I did pass on, since I wanted to first learn how to program and use my new found acquisition. Anyway, I could always borrow programs from my friend. I followed the Vic20 User's Guide through

the learning process of the computer, and I typed in programs from Compute magazine. These activities kept me busy for hours.

It wasn't until several years later, that I bought a Commodore 64. Vic20 software was shrinking from the marketplace, being replaced by the more powerful C64 software. When I tried to type in a program that took up all of the Vic20 memory, I knew at that point I needed to upgrade to the C64.

CF. Now your handle Snogpitch? Can you tell our readers where this name came from?

DW. This is a question I get asked a lot. It was happening whenever anyone saw my handle on the online services such as GENie, Compuserve, Q-Link, and Prodigy. I finally decided when I knew enough to put up a personal website, I would tell the story there, and let everyone read it for themselves.

Once I logged on to GENie for the first time in 1992, I realized I would need an "Online Persona" in order to participate on the live chat channels. What would I choose? Various ideas came to mind, ranging from a Super-hero, to a fictions Star Trek character being the most likely candidate. I remembered a word that was coined, back in my high school days, when Alice and her Adventures through the Looking Glass was discussed. A poem in that story called Jabberwocky held some strange words and names as part of its prose. We were inventing similar words. One word that I "created" was SNOGPITCH.

CF. So a Handle is a "nickname" but Why still use a handle,

DW. About the same time I joined GENie, I was also getting introduced to the local BBS scene. Of course handles were used there as well, and my Snogpitch handle came with me to that community. On the BBS, folks did not want your real name for all to see, especially if the BBS was doing illegal stuff.

Back in the 70's in the United States of America, there was a big thing called the Citizen's Band Radio. Everyone had a handle to identify themselves to other radio enthusiasts, without giving their real names. Folks would chat about various things, but if you were on the trucker's channel, you would get things like where the local law enforcement officers were hiding for catching unsuspecting motorists in "speed traps". Knowing this, you could slow down before getting into the range of the local Constapole's RADAR gun, and not get caught. My Dad loved this technology, but I didn't participate in it.

Snogpitch is a rather unique nickname, one that I have not seen elsewhere. So far, I have not been confronted by someone claiming I used their nickname. If you open the dictionary, you will find the definition as:

nickname n

1. y an invented name for somebody or something, used humorously or affectionately instead of the real name and usually based on a conspicuous characteristic of the person or thing involved Encarta©

CF. You are the president of "Cincinnati Commodore Computer Club" can you give a brief history of the club. How long has the club been running?

DW. I wasn't around when the club was formed. Our club has been in existence since 1983 and grew to a membership of about eighty in 1985. I joined the club in the early 90s, so I can say I've been around for the majority of it's existence. Just this past September, I noticed we had a quiet anniversary of our web-presence (September 1998 was when I created the club's official website on the Geocities community, now 10 years old).

CF. What Commodore Machines do you cater for?

DW. We cater to all systems that Commodore made, up to it's bankruptcy. The newer offerings such as the C64-DTV and Hummer have been added to the list, as they are recreations of the original hardware. The club is however primarily a C64 club.

CF. How many members does the club have

DW. The club currently has 24 in it's membership list.

CF. What would membership entail Is there a joining fee, and what would our user get for joining?

DW. Membership entitles you to:

1. Full use of our public domain library of Commodore 8 bit and Amiga software. Our Librarian has maintained a complete catalogue summary of all disks in our library by disk content and category. You may obtain a copy of any library disk by providing the Librarian a blank diskette and the number of the disk you desire copied. Fees for copies are \$0.50 per side for 8 bit disks and \$1.50 for Amiga disks. Fees for non-members are twice this price. Providing a disk of PUBLIC DOMAIN software or SHAREWARE for the library entitles you to a disk of your choice from the library. NOTE: copyrighted software WILL NOT be accepted in the library unless the company that produced the software is out of business and the software is no longer available from any other source.

2. A Library Catalogue

Disks containing files listing the contents of each 8 bit library disk are provided to new members along with a program that will print the files to the monitor or a printer. Disks containing files of disks added after you become a member may be obtained by providing the librarian with a blank disk(s). Our Librarian is currently putting the disks in .d64 format on a PC hard drive with the idea of being able to put the library on one or two CD-ROMS. A listing of the project so far can be found at:
<http://members.iglou.com/thunderbird/CCCC.txt>

3. A Monthly Newsletter

Newsletters contain lots of helpful information and usually an article on a specific topic. The FOR SALE items are a good way to expand your hardware or software inventory. Help line numbers are also available. We all went through the experience of having problems. If you are encountering difficulties in all likelihood someone in the club has had the same problem. Please make your request for help at a reasonable hour. If you did not receive your newsletter by the next meeting, please contact the newsletter editor or membership person as soon as possible. Providing an article to the newsletter on any computer related subject entitles you to a disk of your choice from the library free of charge (this includes ads of "For Sale" items).

4. A starter disk.

The starter disk contains a variety of public domain programs for the C64 on Side One and the C128 on Side Two. Or if you have an Amiga, you will receive an Amiga Starter Disk.

5. The club has 1000's of used commercial items for sale at very reasonable prices. The list includes software, hardware, accessories, books, manuals and magazines. The items are in separate files by category available free on a double-sided 5 1/4" disk or on our web site at <http://www.geocities.com/c64-128-amiga/sale/>. Members receive a 50% discount on all items listed.

6. Tractor feed items are also available to members:

address labels 3.5" x 15/16" in pastel shades of blue, green, pink and yellow, plus white \$1.00/100. 2.75" x 1 15/16" white labels for 3.5" disks \$1.50/100. And 3.5" x 6" postcards - \$1.50/100

7. Yearly dues are \$24.00. New members receive a \$48.00 credit towards the purchase of any item in our used items inventory at the 50% member discount. All of that is a bargain! Get with our membership officer Roger Hoyer if you are interested in joining our club. thunderbird@iglou.com

CF. Does the club still meet regularly,

DW. We meet on every 3rd Sunday of the month, with the exception of June which is the 4th Sunday.

CF. what do users do at meetings?

DW. A typical meeting itinerary includes a brief business meeting, followed by a demo, or previewing the latest LoadStar diskmag. New inventory acquisitions are also brought to the meeting for members to get first choice before allowing "non-members" the opportunity of purchase. Programming assistance or program support is offered to those who have a need for additional instruction.

CF. I notice you are listed as "official Commodore Club #292." What does this mean

DW. When Commodore was still in business, it gave computer clubs official recognition of its status. We continue to acknowledge this alliance long after the bankruptcy of the company in the 90s.

CF. You arrange Commodore expos or conventions like "C=4Expo" what is "C=4Expo" where are they held?

DW. Expos today are not the expos you might have read or attended in years past. Instead of large corporations showing their latest program, or gadget for the computer, we now have individuals or small businesses showing their "wares" to the hobbyist. Instead of attendance in the hundreds or thousands, attendance at expos today are lucky to get an attendance ranging between 25-100 people. They tend to focus on the social part of the community more than the commercial activities. Our club started it's first expo in 2006, when we found out there was not going to be a spring expo that year, unless someone stepped up to the plate and held one.

Back in the 90's a user group in Okemos Michigan held a spring expo for many years. I attended a few of them up to the last one. The next year, the Louisville Kentucky group (LUCKI) grabbed the torch and held expos for the next several years. When they decided to discontinue, we started it back up. It's been like a relay race, one club runs it for a while, and another one picks it up after a while. One of these days, our club will probably pass it on to another club. We will be doing a show in 2009, contract details are not yet set, but we are planning the Memorial Day weekend for our expo. It will be held at the Drawbridge Inn, in Fort Mitchell Kentucky, same locations as our previous 3 expos. Official dates will probably be announced when your readers have this interview in their hands.

CF. We have started C.C.C.(U.K.) Commodore Computer Club U.K. note the U.K. to distinguish ourselves from your club, does this naming cause your club any problems?

DW. Not any more than CCCC or 5C's from Nevada, or another club, Colorado Commodore Computer Club.

Q. Do you think there is still a lot of interest in Computer clubs, or has the internet removed the needs for local meetings?

DW. I view computer clubs in general as a dying breed, much like the telephone booth, and the milkman bringing bottles of milk to your doorstep in the morning. There will be those few that will still be around, but it will be more of a curiosity, and public demand. The clubs today are more social clubs, especially for the "old timers" that seem to know everything. When is the last time you tried to find a PC user group for the modern personal computers? Did they have anything to offer their membership? Allowing people to tap the knowledge of a fellow member, beats reading an impersonal FAQ any day.

CF. Has the CCCC received a renewal of interest in younger users?

DW. That is a rather difficult question to answer as it seems youth is relative to the age of that person's observations. Our membership is family oriented, thus the children of the primary member are also a part of our club. If you looked at some of the photos taken at our expos, you might see kids playing with the various machines around the expo room. We are not selling expensive items, we expect people to use the machines, and encourage everyone to have fun with their Commodores.

CF. What do you think is or was the most exciting piece of Commodore hardware to be released?

DW. For me the most exciting piece of Commodore hardware was the C128 and 128D. It was 3 computers in one. It had its native mode, with Basic 7.5 64 mode for those who started with the 64, but wanted to keep using the software they had started with on that machine. CP/M mode for the users that were migrating from other platforms. If only the Amiga could have had that flexibility, I might have been more excited when that line of computers made its debut.

CF. Do you have any comments you would like to add?

DW. If you would like to discuss Commodore in real time, visit me, my fellow CCCC members, and Commodore friends on #c64friends on irc using the server on irc.eskimo.com on newnet network. Use your irc client of choice, or Q-Link, and join us on Saturday nights at 9PM EST for the next 3 hours. I am usually there during that time. Others might be available at various times during the week. Visit the webpage at: <http://www.c64friends.com> for a link to the java chat client if you do not wish to install a program on your computer. Transcripts of previous chats are available at the website mentioned above.

Thank you for giving me the opportunity to introduce your readers to our Commodore club. Perhaps, after reading this, more of your readers will feel encouraged to start their own local user group. If our group wasn't around, I know it would be missed in my life.

Interview with Simon Quernhorst:

Commodore Free:
Firstly, introduce yourself to our readers.

Simon Quernhorst:
I am 33 years old and live in Germany. In my spare time I collect old video- and computer games and develop new demos and games for my favourite 8-bit systems: Atari VCS and Commodore 64.

CF: Was the Commodore 64 your first computer?

SQ: I got my first own computer in 1987 - it was a C128D, but of course it was mostly used in C64 mode. Before this I got a Pong console and an Atari VCS console in about 1984.

CF: What other machines are you interested in?

SQ: Considering programming I am focussing on Atari VCS and C64 right now and I have a lot of different C64 models like C64GS and SX64. When it comes to collecting I also have a lot of other systems to collect for. My favourite systems game play wise are PC-Engine, NeoGeo AES and Xbox.

CF: What was your first project with the C64?



game, because I wanted it to enter a 4k game competition on the internet.

CF: It's noted that you have worked closely with Lasse of CovertBiOps, how did this come about?

SQ: I know Lasse since the release of his first Metal Warrior and as the series evolved and my own VCS game MENTAL KOMBAT was released with box and manual, I asked him if we should make a limited release with numbered disks, box, manual and bonus items of METAL WARRIOR 4 and the first three episodes, too. He agreed and we really enjoyed the final results. Now in 2008 when I decided to write a little new C64 game, I asked him about music and sound effects and as he liked the game play he helped me out on the audio part of it.

CF: Are you working on anything new at the moment? Or do you have any future plans for the Commodore?

SQ: I just finished my fifth Atari VCS game and plan to release a total of seven games for that system. On the C64 I am currently developing several mini games. Some of these are competing in mini game competitions on the internet and I am happy to announce that I won the first and third place of the 1k competition at <http://www.minigamecomp.org.uk>. My plans are to compile eight

mini games to a 8k cartridge for the C64. You can watch pictures of them at <http://www.quernhorst.de/atari>

CF: Forget about the future for the moment, what about the past? Which programmers, SID musicians and other C64 magicians do you admire? And what are your six favourite games?

SQ: I cannot say that I prefer a single programmer or musician. I like a lot of work on the C64 from different people - being demos, intros and games. When choosing technically preferred games on the C64 I might answer: Power Drift (code), Commando (music) and Mayhem in Monsterland (graphics). From a collector's point of view my favourite games for the C64 are the cartridges released between 1990 and 1992 by Ocean, System 3, Domark, Dinamic, Silverrock and Disc Company.

CF: What do you think about Commodore Hardware additions like the SuperCPU and, to a lesser extent, other CMD hardware?

SQ: I am only interested in collecting and developing for unmodified and unexpanded systems. The only extensions that I really admire are the Action Replay cartridge series for the C64 and the 1MB extension to my Amiga 500. :) Nevertheless it is great to see that people are also developing new hardware and new software for it.

CF: What sort of response have you had from the C64 community about your Commodore projects?

SQ: I got a lot of positive response for programming new games on the C64. Some people were complaining about the high price of the 20 release cartridges of SHOTGATE/KONFLIKT, but producing the cartridges was very expensive. Consider this limited edition as a collector's item and as an addition to their huge collections. Everyone else is able to download and play the games for free of course.

CF: Have you read Commodore Free, and if so how would you rate the magazine (I can take criticism no matter how harsh)?

SQ: I checked out issue #23 and really enjoy the efforts editors spend on releasing free magazines like Commodore Free, CeVi aktuell and Lotek64. These mags compile retro news and give a good overview of the latest releases and facts. As the C64 is very popular, the machine has got a lot of dedicated websites and even modern demo parties still feature C64 competitions, I think there will be enough news available to keep those mags running. Keep up the good work!

CF: Any final words of wisdom that you'd like to share with our readers?

SQ: Keep on hanging in front of the blue screen of one of the greatest computers and try programming the machine yourself. It is quite easy to start developing on the C64 as a lot of documentation can be found on the internet. Ready. Run.

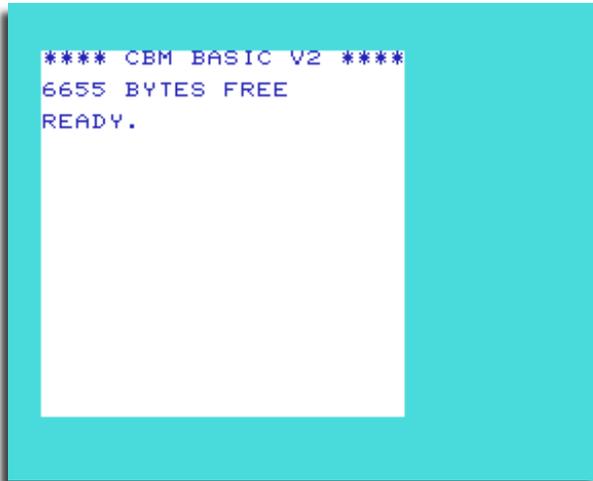


Interview with Simon Owen

Sinclair Spectrum Vic 20 emulator creator
<http://simonowen.com/spectrum/vic20emu/>

COMMODORE FREE

Please introduce yourself to our readers



SIMON OWEN

I'm Simon Owen from Nottingham in the UK. I'm 36 and work as a software developer, writing mainly Windows software for multi-function (fax/print/scan) devices. In my spare time I work on a variety of retro-themed coding projects. Most of them involve the Spectrum, SAM Coupe or floppy disks, but I'll dabble in a bit of anything I find interesting.

CF. Can you tell our readers a little about your first computing moments?

SO. I wrote my very first (BASIC) program on a computer my Dad borrowed from work, which was a large beige machine with green screen display and twin 8" floppy drives. My first proper home computer was a Jupiter Ace, back at the time when many people had VIC-20s or ZX81s. The Ace used Forth rather than BASIC as its built-in language, which was a bit mind-bending for me as a 10 year old!

I finally joined the mainstream a couple of years later, with a 48K Spectrum+. It was largely used for playing games, but after getting Hisoft DevPac for a birthday present I spent more and more time programming. From there I moved to the Spectrum 128+ in 1986, and the SAM Coupe when it was released in 1989.

CF. Are you a Commodore user?

SO. Only on the 16-bit side, really I've got A500 and A1200 machines. The A500 was originally so I could work on Uni projects at home, but I soon got sucked into demos and gaming. The A1200 is still very much alive today, thanks to an accelerator board and bootable Compact Flash card.

I'd still very much like a C64, though I'm not sure I could go back to loading from tapes. I'll definitely get one if I can find a ready-made CF or Ethernet solution for loading software. I'm aware of the IDE64 project, but like most retro hardware projects it expects everyone to assemble their own, and unfortunately I'm a fire hazard with a soldering iron! I also have a SID interface board for my SAM Coupe, as I'm a fan of C64 music. The 6502 core used by the VIC emulator was

actually written for my SAM HardSID player, which can play most SID tunes through the SID board.

CF. Why design a VIC-20 emulator for the ZX Spectrum? Are you stuck for things to do?

SO. Mainly because I like a technical challenge! I never really pay too much attention to how useful the final program will be, and if just one other person appreciates what's been achieved then I'm happy. The VIC-20 emulator is my third 6502-based emulator, after Orao and Apple I. It was definitely a step up in difficulty, but the hardware requirements didn't seem too extreme to be worth a go.

CF. The speed of the emulation must have been a problem. Were there any specific tricks you had to use to obtain better performance?

SO. I try not to worry too much about the final running speed, as I know a standard Spectrum will never reach 100%. Even so, the 6502 core has been highly optimised over the years, and I feel there's little room for further gains.

The emulator does use a few tricks to maximise performance. The version of the 6502 core used by the VIC emulator traps only memory writes, so reads are fast and direct. To support I/O reads from devices such as the keyboard, the write handlers prepare the memory state for the reads. For the keyboard that means writes to the row selection at \$9120 will prepare the keyboard state result at \$9121/\$912f.

Another optimisation is to use a large look-up table to map from VIC to Spectrum display addresses, to minimise the calculations during display writes. This table must be rebuilt if the number of VIC rows/columns changes, but that doesn't happen very often so it's still well worthwhile.

Perhaps the biggest outstanding performance issue is the VIC ROM keyboard scanning performed by the interrupt handler. If a quick test shows that no key is pressed the scanner exits early. Otherwise it continues to perform a full keyboard matrix scan, which eats a significant portion of the CPU time each frame as long as a key remains pressed. That's a problem for games that use keys instead of joystick for control.

CF. I know the Spectrum had a faster processor than the VIC but the graphics and sound must have caused problems when writing the emulator?

SO. The VIC high-resolution mode actually maps through fairly well, with the Spectrum using a similar attribute system for each character. The Spectrum 128 sound chip (AY-3-8912) supports 3 tone channels plus a noise channel, which also matched the VIC capabilities. Unfortunately, the output from the AY noise generator is very limited, so the roaring firing noise in GridRunner becomes an irritating background hiss on the Spectrum version.

On the issue of processor speed, the Z80 typically requires more CPU cycles to do the same work, so the speed is much closer than you'd imagine. A simple example is LDA #0 (2 cycles @1MHz) and LD A,0 (7 cycles @3.5MHz), which both take around the same amount of real time to execute.

You can see bigger differences if you look at the strengths of each processor. The fastest Z80 instruction (4 cycles) is twice as fast on the Spectrum (3.5Hz) as the fastest 6502 instruction (2 cycles) on the VIC (~1MHz). There are more general purpose Z80 registers, which provide faster working space than main memory. The 16-bit register

pairs also give easy access to any location in the 64K address space, as well as 16-bit arithmetic.

On the other hand, a typical 6502 program averages around 2.5 cycles per instruction, which is only just above the minimum instruction time of 2 cycles. I also found that many tasks typically require fewer 6502 instructions compared to the Z80. Zero page addressing also more than makes up for the lack of working registers too, and you're less likely to run out of them even with the most complex tasks.

CF. Colour clashing on the Spectrum was a problem. Has this caused issues with emulation?

SO. The VIC's high-resolution mode is very similar to the Spectrum display, so it wasn't really a problem. In both cases each 8x8 pixel block has a foreground and a background colour. The VIC background colour is system-wide, with 4 bits selecting from the complete palette of 16 colours. 3 bits from the colour RAM select one of 8 foreground colours for the 8x8 block. On the Spectrum each block has a separate foreground and background colour (3 bits for each), as well as bright and flash bits. The bright bit increase the intensity of both foreground and background colours.

The emulator uses a table to map from VIC to Spectrum colours; since not all colours are available (orange is crudely mapped to yellow). Since the upper half of the VIC colours are roughly bright versions of the lower ones, bit 3 of the VIC background colour is mapped to the Spectrum bright bit. That does mean the foreground becomes bright too, but that doesn't make much difference overall.

CF. Writing an emulator where do you start? I presume there must have been a lot of reading and internet trawling for specifications about the Vic's hardware?

SO. I'd done a little research before I started, just to make sure it was vaguely possible. I generally check that there is enough room for my emulator code in the same address space, which requires around 4K for the base 6502 emulator or 8K with a faster instruction decode table. Next is to make sure the display can be suitably represented, even if some features are missing. I also check for any custom hardware that might be needed to run typical programs.

I'd already written the 6502 emulator, so it didn't take long to make some real progress. It only took about an hour to place the VIC ROM code at the appropriate memory locations and have it boot to the start-up screen. The only special code I needed was to stop the VIC memory test overwriting the CHARGEN ROM at \$8000. Once I made that location appear as ROM, it finished the RAM test and set RAMTOP, and then continued to boot. At that point there was nothing to see on the screen, but peeking the VIC display memory locations I could see it was showing the number of free bytes. A small addition to trap writes to \$900F also gave the expected cyan border.

CF. I don't want to get into the my computer was better than yours as our readers will mainly be Commodore users, although I do know of some Spectrum readers. There is still a rivalry between machines why do you think this still exists?

SO. I think there will always be some rivalry, but (except for a few vocal individuals) it seems to be fairly light-hearted nowadays. Almost everyone I've spoken to seems to have at least some appreciation of the other machine. Perhaps not surprisingly, most people seem to prefer the one they grew up with. Would it surprise you to learn that in a World of Spectrum poll comparing the Spectrum 128K and C64 sound chips, 67% of votes were for the SID? :-)

CF. SAM Coupe! Some readers may not be familiar with this machine you also have a Vic emulator for this machine can you enlighten our readers about the machine?

SO. I see the SAM as a spiritual successor to the Spectrum, even though it's no direct relation. It features a 6MHz Z80B, 512K RAM (plus up to 4MB externally), 128 colours, high colour and hi-res modes, 6 channel sound, and up to 2 built-in floppy drives. SAM is backwards compatible with 48K Spectrum software, but only native titles make the most of the extra hardware, of course. The SAM versions of Lemmings and Prince of Persia stand up well against the Atari ST versions. Unfortunately, SAM's release in 1989 was at a time when 16-bit machines were starting to become popular. The user base was also too small for much big name commercial interest, instead relying on smaller companies, diskzines and home-grown software.

The SAM version of the VIC emulator makes use of extra hardware features. The faster processor gives a welcome boost to running speed, the original VIC colour palette is fully supported, and the SAA 1099 noise generator sounds much more authentic. With a Mayhem accelerator card for SAM running at 28MHz you can even have it running at original VIC speed!

CF. I know about 2 different spectrum models the 16 and 48k but I have heard there were other models, does this emulator run on all spectrums. Can you enlighten us on a little spectrum history?

SO. The 16K and 48K versions were both released in 1982. The very earliest software ran on the 16K machines, but it wasn't long before 48K was the minimum requirement. Fortunately, there was a RAM add-on to boost 16K machines up to 48K. The vast majority of Spectrum titles were aimed at the 48K market. The Spectrum+ update in 1984 was little more than the original board in a new case, with a hard plastic keyboard replacing the rubber one.

The Spectrum 128K was released in 1986, featuring extra memory and a 3 channel sound chip (AY-3-8912). Games used the improved sound for menu or background music, and additional in game sound effects. Only a handful of games were 128K only, with most just using the extra space for the music player, digitised speech, or pre-loading game levels.

After Amstrad took over the Spectrum 128 was repackaged in a CPC-like case with improved keyboard and built-in cassette recorder, and released as the Spectrum +2. The final release was the Spectrum +3, which replaced the cassette recorder with a 3" disk drive, as used by other Amstrad machines. To simplify production the later +2 machines (known as the +2A) used the same main boards as the +3, with boot-time detection of the disk drive to know how it should identify itself.

The VIC-20 emulator uses the extra capabilities of the +2A and +3 to page RAM into the full 64K address space, so no memory mapping is needed between VIC and Spectrum addressing. That does mean it won't run on the original 48K or even the base 128K model, but since most users will be using a Spectrum emulator it's not a problem.

There is a second VIC-20 emulator for the Spectrum, written by Jimmy. It uses his own 6502 core and does include the memory mapping needed to run on 48K machines. That does mean a speed penalty for each memory access, but the emulator does still run surprisingly well.

CF. How long did creating the emulator take?

SO. Start to finish was 18 months, but I only spent about 2-3 weeks of evenings actually working on it. The first evening was to confirm the ROMs booted, and to get a sample game cart booting. After that I left it alone to think about how best to emulate the display. I only picked it up again recently to finish it off.

I originally believed that changes to chargen RAM would be used by games for character animation. That would mean tracking the visible characters, to know which needed updating when the underlying character memory was changed. To do that would require multiple levels of tables, with a speed hit for all display writes. Before I'd got that straight in my head I'd moved on to other projects.

The final emulator is actually much simplified, ignoring the chargen memory writing that (it turns out) very few programs need. Display writes are still trapped, but the only work is to write the current character representation to the Spectrum display. The only complications are when the chargen pointer (\$9005) or rows/columns (\$9002-3) are changed, when the entire display must be redrawn.

CF. How many people are involved with creating the package?

SO. Just me for the coding, but I've had input from a number of people. Jimmy helped kick-start me into finishing the emulator, and various optimisation ideas we've discussed have benefitted both our emulators. Dr Beep also provided a great opcode-decode speed-up, where the opcode itself forms the low-byte of the handler address, for around a 5% speed boost.

CF. What comments from the Spectrum world have you had?

SO. Actually not very much, with more feedback from the Commodore world! Most of the Spectrum interest has been in the 6502 core, rather than the emulator itself. Perhaps the VIC-20 is a little too unknown to the average Spectrum user, compared to the C64.

CF. Do you have plans for a Commodore 64 emulator for the Spectrum or SAM Coupe? I guess speed would be a major stumbling block.

SO. The C64 would definitely be a lot more challenging. It would probably need the extra video capabilities of the SAM Coupe to support the colours and modes for games. The need for memory paging would mean moving from direct to slower mapped memory accesses, and trapping zero-page paging I/O would slow all zero-page writes a little. Graphic sprites would complicate the display updating too, requiring twice the amount of drawing when they're moved. I'm very tempted to look at supporting the C64 so that BASIC is usable and if that goes well then maybe I'll enhance it further.

CF. With the emulator not running at full speed are there any programs that just refuse to run?

SO. The running speed isn't a problem for most programs, as they're not aware of how fast they're running. Any awareness they have of elapsed time can be adjusted as part of the emulation, if required. The most likely reason for non-working programs will be incomplete emulation. For instance, I make no attempt to support the hardware timers or scan line counters, so anything relying on them will fail. It may be possible to add some basic support for them, but I've not needed them yet.

CF. How accurate is the emulation?

SO. The CPU emulation is quite accurate, supporting all the officially documented instructions. It's actually a 65C02 emulation, which includes some page-wrapping fixes and new instructions that were

missing from the original 6502. The VIC chip is not fully emulated, but I've added enough support for the sample programs to run. The display is also narrower than the original machine, but I believe that's just the way the VIC display is generated, where the pixels are fatter.

CF. Do you intend to support Commodore hardware to attach to the spectrum, maybe a cartridge port?

SO. It's unlikely original carts will ever be supported directly as it would require special hardware to connect them to a Spectrum. I would still like to support dumps of game carts, as used by VICE and other emulators. Cart support isn't completely straight-forward either, as some will move the address of the VIC display, and the emulator is optimised for the normal addresses. It also depends on the memory banks provided by the cart, as space is needed for the Spectrum display and the emulator itself.

CF. Did you consciously remove some VIC features so that the emulator would run faster or be quicker to produce?

SO. The biggest feature omission is the high-colour mode, which will be needed by most newer VIC-20 games. Not supporting it doesn't always make a big difference to appearance; it just shows with fewer colours. The flags in Blitzkrieg still look like flags, but they're missing the proper blue and white colouring. High-colour mode would only be possible with the SAM Coupe's 4-bit display mode. The change would also mean 4 times the amount of display data for any drawing, which would slow the emulator down a bit. I also cut the available memory from 24K in the original version to 3K in the released version. Part of that was for software compatibility, but it was mainly to give more space for the Spectrum display and some look-up tables. I should be able to provide at least 16K by moving a few things around, perhaps having the Spectrum display in a different memory bank.

CF. You have given the source code away, how would you feel about others altering and maybe tweaking the software?

SO. I'm quite happy for any or all of it to be used, if they credit me somewhere. That could mean using the 6502 core in a completely different project, or submitting improvements to the existing emulator.

CF. Do you have any other software; we better say Commodore related in the pipeline?

SO. Nothing currently, though new projects do have a habit of springing up from nowhere! There has been some recent discussion about a SID board for the Spectrum, for playing C64 tunes during Spectrum games. It's certainly wacky enough for me to look at getting involved if it takes off.

CF. Well whatever your motives I think this is a major achievement and I have to take my hat off to you well done.

SO. Thank you for your interest and an interesting interview!



Understanding VIC - 20 Ram Expansion

One of the most confusing things about the VIC-20 is the screwy way that memory locations shift around when you add RAM expansion. Add 8K of RAM or more and everything changes: suddenly Screen and Color Memory move to different locations. Old programs for the unexpanded VIC that POKE to the screen or use redefined character graphics no longer work. Depending on the amount of RAM expansion you have, BASIC even begins in different locations. Aargh!

The diagrams on this page all come from Leif Bloomquist's Commodore VIC-20 Memory Blocks Illustrated. <http://8bitcentral.com/files/vicmemory.pdf> This PDF helped me a lot in my own quest to figure all this out. Click [HERE](http://home.ica.net/~leifb/commodore/index.htm) <http://home.ica.net/~leifb/commodore/index.htm> to visit Leif's site.

Also, Anders Persson has written a great tutorial: <http://user.tninet.se/~pug510w/datormuseum/expandedvic20.html> (ED Reproduced in this issue of Commodore FREE) Graphics and Bigger Screen in BASIC on the Expanded VIC-20. Anders' website is [HERE](http://www.commodorefree.com).

Another good resource is the book VIC Revealed, by Nick Hampshire. If you want to really understand how the VIC works, you owe it to yourself to get this book. It's fairly common, you can often find it on eBay. <http://www.ebay.com/>

Like most things, it doesn't seem that hard once you figure it all out ...

FIGURING IT ALL OUT

| Dec | Hex | Block# |
|-------|------|---------|
| 57344 | E000 | Block 7 |
| 49152 | C000 | Block 6 |
| 40960 | A000 | Block 5 |
| 32768 | 8000 | Block 4 |
| 24576 | 6000 | Block 3 |
| 16384 | 4000 | Block 2 |
| 8192 | 2000 | Block 1 |
| 0 | 0000 | Block 0 |

Let's imagine the VIC's memory as a series of 8K "blocks," starting at 0000 (block 0) and ending with block 7 (E000). Expansion RAM can be placed in the yellow blocks: 1,2,3 and 5. There's also a little 3K slice available in block 0 that we'll talk about later.

If you put 8K in blocks 1,2, and 3 your VIC will have 29K total RAM available to BASIC. That's the 5K the VIC came with plus 8K in blocks 1,2 and 3. RAM has to be contiguous to be seen by BASIC. That is, BASIC only sees RAM as a single big chunk of memory. Since blocks 1 - 3 are right next to each other, the VIC can see all of it as one big 24K hunk of RAM. So, looking at the diagram, you can understand why 24K is the maximum amount of RAM expansion available to BASIC.

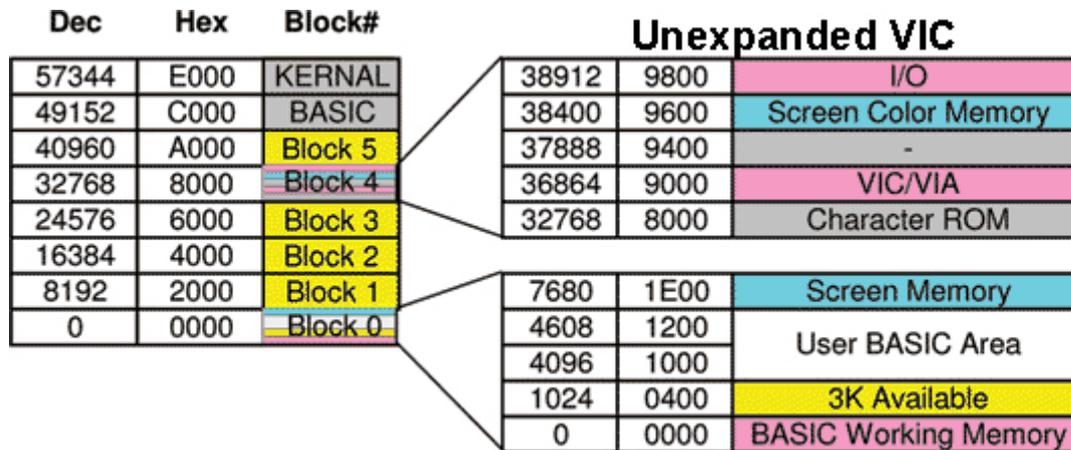
BLOCK 5

You've no doubt observed that block 5 is not contiguous with blocks 1,2, and 3. Block 4 is in the way. So BASIC can't see RAM in block 5. You might be asking yourself: If RAM in block 5 is invisible to BASIC, what good is it? You can place machine language programs in block 5, or use the RAM to store data that can be read with BASIC's PEEK statement. So, RAM in block 5 can still be quite useful. The only thing you can't use it for is to add to the RAM available for BASIC programs.

Another feature of block 5 is that programs stored in this area can be "auto-started" when the VIC is reset. Most cartridge games are located in block 5. You can tell because these games start automatically when you turn the VIC on. To further illustrate, the Programmers' Aid Cartridge puts its program in block 3. That explains why you have to enter SYS 28681 to launch the Programmers' Aid. Only programs located in block 5 will start automatically when you reset the VIC.

It sounds pretty straightforward, doesn't it? Can it really be this simple? Unfortunately, the answer is no. We're going to have to take a look at what's in block 0 and block 4 to see where things get a little tricky.

BLOCK 0



The diagram above blows up blocks 0 and 4 so we can see their contents. Take a close look at block 0 on the unexpanded VIC. The white area, from 1000 to 1DFF, is our User BASIC Area. This is the RAM that's available for BASIC programs. This one tiny little slice of block 0 is all the RAM we have to work with. Not much, is it? Let's start adding some RAM and see what happens.

ADDING 3K

The yellow section from 0400 to 0FFF in our block 0 diagram is a 3K area set aside for RAM expansion. This is where the RAM in Commodore's 3K RAM Expansion cartridge or 3K Super Expander goes. After adding one of these cartridges our block 0 diagram looks like this:

| | | |
|------|------|----------------------|
| 7680 | 1E00 | Screen Memory |
| 4608 | 1200 | User BASIC Area |
| 4096 | 1000 | |
| 1024 | 0400 | |
| 0 | 0000 | BASIC Working Memory |

And there we have it-- The beginning of BASIC moves down to 0400 and we now have an extra 3K of RAM to work with. Now let's add an 8K RAM cartridge to block 1.

| | | |
|-------------|-------------|---|
| | | Block 1 8K RAM CARTRIDGE INSTALLED |
| 8192 | 2000 | |
| 7680 | 1E00 | Screen Memory |
| 4608 | 1200 | User BASIC Area |
| 4096 | 1000 | |
| 1024 | 0400 | |
| 0 | 0000 | BASIC Working Memory |

The diagram above represents our VIC with 3K expansion, plus 8K in block 1. Uh oh, there's a problem. Do you see it? Remember, all RAM available to BASIC must be in a single continuous chunk. Screen Memory (1E00 - 1FFF) is separating the User BASIC Area from the 8K expansion we want to put in block 1. What do we do? There's only one thing we can do: we have to move Screen Memory somewhere else. The VIC does this automatically when you add expansion RAM to block 1. Here's what block 0 and 1 look like after adding 8K to block 1

| | | |
|-------------|-------------|---|
| | | Block 1 8K RAM CARTRIDGE INSTALLED |
| 8192 | 2000 | |
| 7680 | 1E00 | User Basic Area |
| 4608 | 1200 | |
| 4096 | 1000 | Screen Memory |
| 1024 | 0400 | 3K Available |
| 0 | 0000 | BASIC Working Memory |

Screen Memory moves from 1E00 to 1000. Now we can add 8K to blocks 1,2, and 3 and all the BASIC RAM will be in one continuous chunk. BASIC now begins at 1200 and runs until the end of expansion RAM. One more detail: Color Memory in block 4 has to be a precise distance from Screen Memory. Since we moved Screen Memory, Color Memory will have to move too-- from 9600 down to 9400.

| Dec | Hex | Block# | Dec | Hex | Unexpanded | 3K Expansion | 8K+ Expansion |
|-------|------|---------|-------|------|----------------------|----------------------|----------------------|
| 57344 | E000 | Block 7 | 38912 | 9800 | I/O | I/O | I/O |
| 49152 | C000 | Block 6 | 38400 | 9600 | Screen Color Memory | Screen Color Memory | - |
| 40960 | A000 | Block 5 | 37888 | 9400 | - | - | Screen Color Memory |
| 32768 | 8000 | Block 4 | 36864 | 9000 | VIC/VIA | VIC/VIA | VIC/VIA |
| 24576 | 6000 | Block 3 | 32768 | 8000 | Character ROM | Character ROM | Character ROM |
| 16384 | 4000 | Block 2 | | | | | |
| 8192 | 2000 | Block 1 | 7680 | 1E00 | Screen Memory | Screen Memory | User Basic Area |
| 0 | 0000 | Block 0 | 4608 | 1200 | User BASIC Area | User BASIC Area | Screen Memory |
| | | | 4096 | 1000 | | | |
| | | | 1024 | 0400 | 3K Available | | 3K Available |
| | | | 0 | 0000 | BASIC Working Memory | BASIC Working Memory | BASIC Working Memory |

Commodore VIC-20 Memory Blocks Illustrated

Leif Bloomquist, leif@schematafactor.com

| Dec | Hex | Block# | Usage | Dec | Hex | Unexpanded | 3K Expansion | 8K+ Expansion |
|-------|------|---------|----------|-------|------|----------------------|----------------------|--------------------------------|
| 57344 | F000 | Block 7 | KERNAL | 38912 | 9800 | I/O | I/O | I/O |
| 49152 | C000 | Block 6 | BASIC | 38400 | 9600 | Screen Color Memory | Screen Color Memory | - |
| 40960 | A000 | Block 5 | (1), (2) | 37888 | 9400 | - | - | Screen Color Memory |
| 32768 | 8000 | Block 4 | | 36864 | 9000 | VIC/VIA | VIC/VIA | VIC/VIA |
| 24576 | 6000 | Block 3 | | 32768 | 8000 | Character ROM | Character ROM | Character ROM |
| 16384 | 4000 | Block 2 | | | | | | |
| 8192 | 2000 | Block 1 | | 7680 | 1E00 | Screen Memory | Screen Memory | User Basic Area ⁽¹⁾ |
| 0 | 0000 | Block 0 | | 4608 | 1200 | User BASIC Area | User BASIC Area | Screen Memory |
| | | | | 4096 | 1000 | | | |
| | | | | 1024 | 0400 | 3K Available | | 3K Available |
| | | | | 0 | 0000 | BASIC Working Memory | BASIC Working Memory | BASIC Working Memory |

Legend:

| | |
|------|----------------------------------|
| ROM | |
| RAM | Built-in RAM |
| EXP | Available for external expansion |
| I/O | Used by Kernal, BASIC, or I/O |
| SCRN | RAM Used for screen display |

- Notes:
- (1) 8K Cartridges normally here (Autostart).
 - (2) SYS64802 performs a soft-reset which usually starts Autostart cartridge images.
 - (3) User BASIC Area continues beyond \$2000 with 8K+ Expansion.

- Tips:
- To find where BASIC Area starts: PEEK(43) - 256*PEEK(44)
 - To find where BASIC Area ends: PEEK(55) + 256*PEEK(56) (These can also be changed to protect memory from BASIC)
 - To find which "page" screen memory starts: PEEK(648)
 - 30 = \$1E, meaning \$1E00 (Unexpanded or 3K expanded VIC)
 - 16 = \$10, meaning \$1000 (8K or more expansion)

v2

Graphics and bigger screen in Basic on the expanded vic-20.

This article has previously been posted on the vic-20 forum 'Denial', <http://sleepingelephant.com/denial/index.html> but this version has a few updates...

First, (as everyone probably know), when having expansion memory (ram in block 1), then both the basic memory and the screen is moved to other positions in memory.

The screen now starts at: 4096 (7680 on unexpanded).
The colors for the screen starts at 37888 (38400 on unex.)
The basic memory starts at $18*256+1=4609$ (4097 on unex.)
If you are going to use expansion ram and want to use your own graphics (or a bigger screen), then you need to move the basic memory. The reason for this is that the vic chip only can access the internal memory (not the expansion memory). And by default, it only have enough memory for the default screen before the basic memory starts. It's made this way so that all of the memory can be used for basic programs by default.

So you need to alter the start address of the basic memory and move it forward in memory to make room for things that needs access of the vic chip. So that these things can be put in the internal memory.

(It is however possible to put graphics data in the internal tape buffer if you think it's enough with only 38 characters and a normal sized screen. Then you don't need to move the basic memory at all. More about this later.)

Moving the basic memory
If you for example do this:
`poke44,32:poke32*256,0:new`

Then your basic memory will start at the expansion ram (8192) and leave the whole graphics mem free for your own graphics, a bigger screen, etc... ($32*256 = 8192$). You might not need to move it that far though. For example on my Tribbles game,
<http://user.tninet.se/~pug510w/datormuseum/tribbles.html> I have it at $28*256$ I think (`poke 44,28:poke28*256,0:new`)

REMEMBER - Always load basic programs with just ,8 !!! If you load it ,8,1 then it will be loaded to the same memory position from where it was saved, and that is not very good when you have moved the basic memory.

A bigger screen
Now when you have moved your memory, you can use some of the internal memory for a bigger screen. (Mostly usable on PAL where quite much of the display is unused.) You alter the screen size and positions with the following vic registers:

36864 bits 0-6: screen horizontal center
36865: screen vertical center
36866 bits 0-6: number of columns
36867 bits 1-6: number of rows
In basic, to set:

screen x position: `poke 36864,x`
screen y position: `poke 36865,y`
screen width: `poke 36866,(peek(36866)and128)+w`
screen height: `poke 36867,(peek(36867)and129)+h*2`
For an example in basic (and for trying different settings out), download my "overscan" program.
<http://user.tninet.se/~pug510w/datormuseum/oldvic20page.html#overscan>

When you have opened a bigger screen with a different column size than the default, then the lines will behave strangely, but don't worry, this is normal! The system screen still is 22x23 regardless of how big you screen is, and this is why the screen seems to behave strangely. If you don't like this, then keep the original width and only make the screen taller. It's here much of the unused space is anyway (on PAL). You can't use print commands to put stuff on the extra space. So the only way (in pure basic) is to poke. The screen just continues beyond where the default screen ends at 4602, so poking values greater than that changes the contents of the extra space on the screen. It works exactly as poking on the normal screen, but you just have a bigger space now. Another way is to use my "Extra Screen" program,
<http://user.tninet.se/~pug510w/datormuseum/extrascreen.html> then you can use normal print commands (almost). (You are allowed to include the machine language part in your programs)

An example using "Extra screen":

```
10 print "{clr}This text will be put"
20 print "on the extra space below"
30 print "'the 'normal' screen."
40 sys 5352
50 print "{clr}And this will be on the"
60 print "very top of the whole screen."
```

SYS 5352 just copies the "normal" screen to below the "normal" screen in a jiffy. In other words, you don't need to start the whole "EXTRA SCREEN" program, just use this little routine to copy the screen. I did this in my Mega Omega game.
<http://user.tninet.se/~pug510w/datormuseum/oldvic20page.html#omega>

Graphics
This is pretty straight forward. You reserve memory the way described above and then find a suitable place in memory (after the space used for the bigger screen). The register to use is:
36869 bits 0-3

In basic: `poke 36869,(peek(36869)and240)+a`

The addresses a you can use is

13 for 5120
14 for 6144
15 for 7168
8 for 824 (chars 103-127) and 664 (chars 83-95) (Thanks Mobsie!)
0-3 for the character rom.
So for example `poke 36869,(peek(36869)and240)+14` will use the graphics data at memory address 6144 and forward.

The easiest way to do graphics is to use some program to draw directly into the memory and then just save the memory out as a file (for example with a machine language monitor). I usually use "The final Cartridge"'s ML monitor on the C64 to input and save the graphics. Then when you are to use it, you just load ,8,1 and it will load in the right place.

The tape buffer - Selecting 8 as "a" above makes it possible to put graphics in the internal tape buffer. For this you don't need

to move the basic memory or anything. And it should work on both expanded and unexpanded vics!!! Also seems to be the only possible position if you intend to compile your program with the Austro Compiler.

"ftp://ftp.zimmers.net/pub/cbm/vic20/utilities/8k/Austro-Compiler.prg">Austro Compiler Here is a demo of how to use this position: Winter simulator.
<http://user.tninet.se/~pug510w/datormuseum/snow.html>

Working / and Saving

When working on a project, it's easiest to first load any graphics and ML files,8,1, then move the basic memory as described earlier and last load the basic ,8. Then just save the basic program ,8 as you progress with better and better versions.... But when you are ready and want to turn it into something that others easily can load on their vic... Then there are two approaches.

1. The mutiple file approach
2. Single file approach

The multiple file thing is simply that you make a loader that moves the memory and loads all the files. It can be a little tricky as the NEW command is used to move the basic memory... A tip if you like to use this approach is to use the keyboard buffer that starts at position 631. Position 198 holds the number of letters in the buffer. So by doing this: poke631,131;poke198,1 You put a "load/run" keypress in the keyboard buffer. And if you before that print something like LOAD "PROGRAM",8{up}{up}{up} on the screen, then you can make it load a program even after the new command. Take a look at the first file of my "VIC EXTRA SCREEN" as an example.

The Single file thing is a lot nicer, and it also makes your game work on both disk and tape. It involves having two basic programs as well as any graphics and ML parts in memory at the same time and then saving the whole thing. The main basic program at your new moved basic position plus a little starting program at the original basic position that moves the memory and runs the main program.

To run a basic program that is somewhere else in memory and not in the current basic memory, you have to set these pointers first:

| | | |
|-------|--------------------|--------------------------|
| 43-44 | Start of Basic | |
| 45-46 | Start of Variables | First byte after program |
| 47-48 | Start of Arrays | First byte after program |
| 49-50 | End of Arrays | First byte after program |
| 51-52 | String storage | End of memory+1 |

Let's say we have the whole memory set up like this:

- Small Basic start
- Room for bigger screen
- Machine Language
- Graphics Data
- Main Basic program

When a program is loaded, the pointers 45,46 etc. are automatically set to behind the loaded program, and because I have the basic program last in the resulting file, those pointers are set by themselves when the file is loaded. Because of this, I only have to change the start pointer.

So, if you have the basic starting at 8192, you only have to do the following...

(reset)

10 poke 44,32:run
 (make the small basic start)

load "gfx and machine language part",8,1
 (loads into the graphics mem)

poke 44,32: poke 32*256,0: new
 (move basic to 8192)

load "basic part",8
 (load main basic part)

poke 44,18
 (move back the beginning of basic to default)

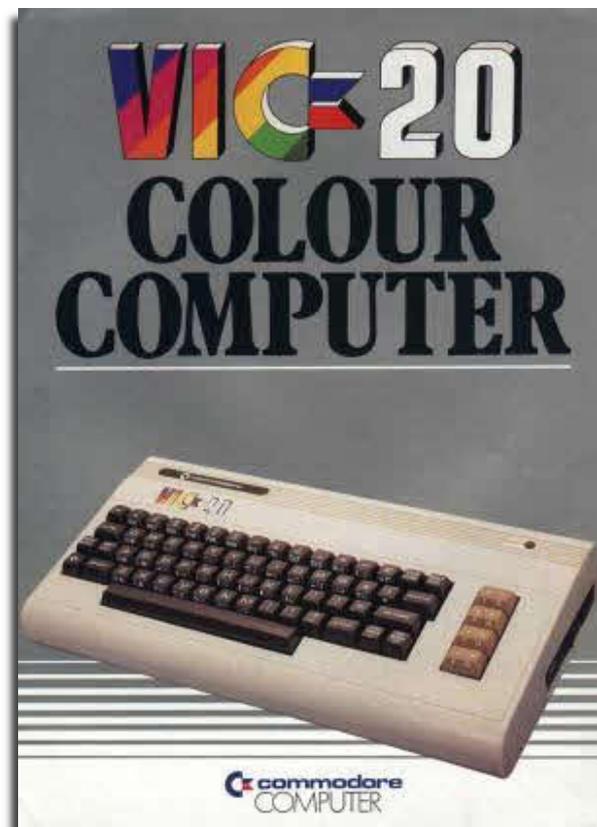
save "whole program",8
 (And everything saves as one big file)

Very nice as you don't even have to bother to look what the ending address is...

Good luck!

Anders Persson
<http://listen.to/boray>

 Commodore Free would like thank Anders Persson for the reprint of the article
<http://listen.to/boray>
 Anders Persson



Commodore Computer Club U.k. Second Meeting

Commodore Computer club 2nd Meeting

Shaun:

For me, the weekend started on Thursday, as I had invited our new chairman Peter Badrick to venture down from the North East to the West Midlands a day early in case there was any club business to mull over. It also gave me an excuse to have an extra evening "playing with those old pieces of junk", as my partner Lisa might say. I was due in work in the morning, and had arranged to finish early, however I had been in discomfort with my back for the eighth day, and finally gave in and went to see the doctor.

Pete had said that he expected to arrive at 2pm, and would you believe it, he arrived on time. "What are the chances", I thought, "of a bus driver arriving so punctually?", though I did not pose such a question to him. I offered Pete a brew (for those not from the North, a 'brew' refers to a hot beverage, usually tea or coffee), and in usual fashion, he asked for a black coffee with milk. After being distracted by my Commodore 2001 PET, we soon checked if my Commodore - which I will shamefully say hadn't been switched on since before Christmas - was still in some sort of working order. Happily, it worked, and I loaded a few pieces of software such as Metal Dust, Block Frenzy (a new release on cartridge which I got from the RGCD shop), and Wheels, amongst others.



Pete had brought with him his 1541 Ultimate+, an excellent upgrade for the C64 and 128, which also works with the any other

Commodore with a serial bus, and at some point my Plus/4 and VC-20 appeared. I demonstrated Eslapion's cartridge which has some 127 games on it (or is it 128? I can't remember exactly).



Fortunately, all of my kit seemed to be in working order, and although I was as usual very unorganised, at least I knew where it all was. I get out a couple of PET machines, one was working but soon started to fail, which wasn't a nice experience, so unfortunately there was little chance of a PET having a presence during the Saturday. As no one else could make it on the Friday, there is little to say about it other than the pub which had many ASBO-owning folk from the near-by 'problem' estate. After paying £1 (!!!) for a game of pool, and being surrounded by the locals, we decided to head into Birmingham City Centre, to Scruffy Murphys.

Finally, the big day arrived. To my relief, Nigel Parker had made the journey to attend, which made things a whole lot easier, and Chris Snowden, of www.commodore16.com, also arrived at Lisa's door. Our newest member, Patrick Bakker, was travelling from Holland for the event, and was bringing with him a Commodore 116.



After buying tea, coffee, biscuits and juice, we started to load up the convoy of stuff. Luckily, Chris had his SatNav ready and we headed out. Chris then went to pick up Patrick from the city centre.

Despite the small turn out, the day went quite well. Mike Dailly arrived, bringing with him "yet more junk" (I'm sure his wife is relieved), and we also got to see Xeo3 on the C64 and Plus/4.

Various games were played on the Plus/4, and a few trades were made, mostly from the stuff that Mike had brought with him. Chris was busy fixing several Plus/4 machines, including two of mine, one of which was working that I'm sure didn't. With exception to the Plus/4s piled around Chris, there were surprisingly very few technical problems. A lot of the day was mostly talk and drinking tea and coffee. All too soon, the day had ended, and Chris, Patrick, Pete and I went to Lisa's place to enjoy some alcoholic beverages. Being a light-weight nowadays, I only had a few, and talk covered everything from the problems with VISTA and the joys of 8-bit computing. The next meet is being planned for June, and will be held in the North East of England, hosted by Pete, and I have something of a "holy grail" in Commodore terms for those attendees to see, so keep an eye on www.CommodoreComputerClub.co.uk for more news about this. Hopefully, I'll see you then.

Fortunately, all of my kit seemed to be in working order, and although I was as usual very unorganised, at least I knew where it all was

Commodore Computer club 2nd Meeting

Nigel (Commodore Free)

Loading up slow process for me just one item at a time! The 2nd official Commodore Computerclub U.k. Meeting www.commodorecomputerclub.co.uk was held on the 23/24/25th of January 2009

Still suffering from back problems, I was only able to attend the main meeting event on the Saturday 24th, I loaded up the clubs "for sale" Items into my car, as we had some hardware this was a slow process as I had to transfer each computer one at a time from one box to another, it took me two whole days to move all the items downstairs, not constantly but I moved a small amount of items one at a time. Of course if you have any good back cures now is the time to let me know!

With the car loaded up, and the wife dropped off at work, the postcode was punched into SatNav. It told me that the journey would take around 1 hour and 45 minutes from Blackburn. I drove down the very clear motorway setting off at 8am. The journey went well and my back felt good. I stopped several times to stretch and ensure I did my specific exercises I have been assigned by the osteopath and physio.

Eventually the tones on my SatNav informed me "You have reached your destination" and I started for the house of Shaun Bebbington. I was more or less at the door so turned off the SatNav locked the car and wandered over.

Shaun open the door welcoming me - he was glad I had made the effort to come down and also glad I had with me the cash to pay for the hall and drinks. Shaun showed me into his house that seemed stuffed with retro items in every cravess of every room - a PET here and Commodore there, and of course a 128 setup with a RAMLink, SuperCPU, laser printer, CMD HD and FD 4000 floppy drive with some other items littered around.

Shaun introduced me to Peter Badrick, having never met except in the virtual world he looked taller than I had imagined him, we shook hands and then set about looking at Pete's DTV that was faulty. We replaced the batteries and would you believe the device sprung into life! Pete had purchased the device second hand and it was a NTSC version, we looked at some of the hidden gems and the interlaced picture of the creator Jeri Ellsworth. Shaun's telly would only display NTSC in monochrome but that didn't stop us playing a few of the games. This was the first NTSC model of the DTV I had seen, Shaun was keen to point out the differences, we chatted about the DTV and I made comments about the colours being inaccurate and the sound problems with the device (I am sure everyone will hate me now, its good but its not like playing on a real c64, I expect most people don't notice the difference). [Of course, there are ways to fix the colour problems, Nigel. You just need to be good at soldering, which I am not great at - Shaun].

Shaun gave me a tour of some of the Commodore PETs, they look so good, just what you would call a real retro computer. Pete had been offered one of the rarer models by Shaun, but commented that if he took it home his wife would leave, so it was the PET or wife, I hope Pete enjoys his Commodore and doesn't miss his wife too much!

Chris Snowden of www.commodore16.com said that Patrick was also attending from Holland (Patrick Bakker is one of the moderators at [commodore16.com](http://www.commodore16.com) and also writes for <http://commodore-gg.hobby.nl>) and may have to pick him up because he always gets lost in the UK.

Shaun put his young baby girl into the push chair and we set off to the shops for the clubs supply of tea and coffee, all 3 of us chatting eagerly about Commodore and before we had walked 5 steps, Ruby Mae fell asleep in the pram. We collected the sundries from a local shop and set off back to Shauns house. We then loaded up our 3 cars full of goodies and set off to the Hall a short drive away.

In the hall we paid the lady for our 5 hour stay, excepted the hall rules and set about lining up tables and putting out a sign for others to find the way, and curious people who might pass by past wondering what the C= logo was all about. Everyone beavered away except for myself as I couldn't lift heavy boxes and felt guilty asking people to do work for me but little by little the hall was set up and ready to go.



PICTURE OF THE HALL

The next person to arrive was Mile Dailly, everyone greeted him and Mike kindly donated three boxes of junk to the club. These were various machines working/not working and other small pieces of hardware that we could use a spares or sell of for club funds as we thought fit. Chris noticed some rare books for the C16/+4 and wanted to purchase them, then Patrick bought a boxed lightpen adding to club funds and also became a member. Other items were sold through the day and we didn't do to badly even with the cost of the hall. The shop is certainly now well stocked anyway!

Chris repaired Pete's broken Plus/4, replacing and testing various chips from a "stock" of working items he brought with him on the day. Pete also gained a transplanted keyboard from a spare Chris had. Other machines were tested and badged as working or dead and dealt with appropriately. Tea coffee and biscuits were on hand provided by Shauns girlfriend, who manned the refreshments area of the hall.

Chris then went into detail about the game he created called "Church of Death". This is a Plus/4 text adventure. Chris explained the tools needed and used for the games creation and gave a brief demo of how it would be played, also producing a finished version on tape with professionally produced inlay and tape through Cronosoft

Shaun wanted to know what differences there were between Commando on the 64 and the Plus/4. Both versions were loaded side by side and there were pluses and negatives for both. The Plus/4 suffers from lack of hardware sprites and the c64 suffering from small glitches on screen and multiplexed sprites disappearing, and of course the SID was superior, or at least Shaun thought so.

I suggested we watched the Shades demo on a real Plus/4, after various attempts to load the demo on the 1541-III, we managed to finally have the demo run on Pete's 1541 Ultimate+, must be down to the quality of hardware emulation, the other problem of course was we only had the demo on an SD card and no cable to create a real disk from the D64 image. The 1541-III was a rare sight for me but seeing two in one room was unbelievable, bit like a bus you wait for one then 2 come all at once, a nice piece of work with the screen displaying the time needed to load each file I thought!

Shaun loaded the old Mathematica demo by Reflex on a lonely C64 at the back of the room. As we watched the demo, which includes a 3D "Doom-like" environment my thoughts turned to Doom on the



64. Mike suggested everyone was fed up with Doom now and thought that was why there was little interest in converting or attempting a conversion to the c64, I am now wondering if a cut down version of Doom for the C64 would be anywhere near enough to call itself Doom because so many features would be removed that it would really be a poor imitation.

We asked ourselves why demo creators spend so much time creating a demo rather than a tool or game. Mike suggested demo and Games coders are totally different - needing different logic and program thinking to do each task. Mike then demoed Xenon TriOxide (Xeo3 - <http://www.xeo3.com>) on the Plus/4 and then on the C64. Although its still early days for the game we had to admit it looked very promising. I especially liked the c64 version as it appeared to run smoother Mike said this was down to the C64 using hardware sprites. Mike said that the game would probably take two months to complete, if he could find the time.

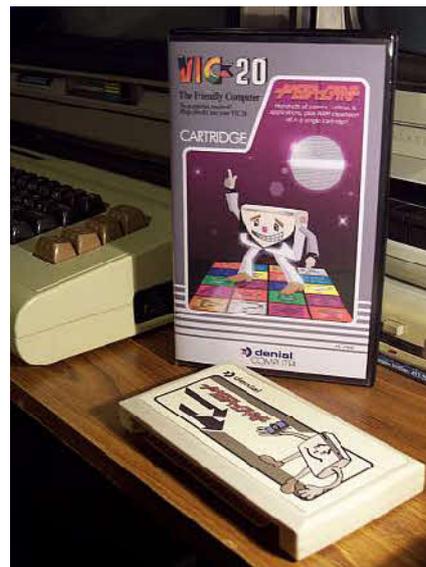
Shaun then demonstrated Metal Dust, which is still the world's only SCPU-specific game, by Protovision. Many people claimed I was too hard on the review and even Shaun though I was a little critical. True I am no programmer, but my comments do stand in that they spent all the processor time running the digital music and not enough time on the game content. I think that Enforcer 2 looks and plays better, and that doesn't need a SCPU to run. I also cant tell what is background and what is foreground on the game. Mike commented that the background should be much darker in colour. I do feel its a lost opportunity somehow.

Many of Jasons Kelk's games were shown and played, like the excellent ViColumns for the unexpanded VIC-20. I spent a good amount of time playing this game. It's another Jason game where everything works and shows his attention to detail.

Shaun also showed me a copy of Block Frenzy on a cartridge (again by Jason), which is based on a java game recreated on the C64 (and GS) from RGCD's webshop. It looked like a professionally produced cartridge, including a colour box. If you like tetris type puzzlers then this is a must purchase, its difficult to describe the game play but here goes: you guide a square and can't touch the edges of the screen, nor the various shapes that bounce randomly around the screen - it sounds simple but is this the most frustrating game ever, possibly. The simple task is to last longer than the last guy to get the highest score.



Next was the VIC multi-cart was shown, by Eslapion which is sadly not currently available. Once the machine is switched on, a simple menu system is displayed allowing you to select from over 100 games which load almost instantly.



Chris pointed out that it was dark and all of a sudden the day had come to a close. Sadly we packed and tidied up the hall. The CCC UK has more items for sale now. Shaun and Pete have donated some cassette games and of course Mike's odds and sods made it more than worthwhile. We left the hall and unloaded cars at

Shaun's place, at that point I had to leave and so I said my goodbyes and set of back home. By now the back pain that had eluded me all day was starting to creep in, and I couldn't take any pain killers as it would have made it unsafe to drive home due to the strength. I made several stops but eventually arrived safe at home. Shaun sent a text to see if I arrived safe and wish me happy birthday, all in all an excellent day. Well worth the time, and showing there is interest in a UK meeting, we decided that maybe June would be a good date for the 3rd official club meeting, more information as and when it arrives in.

Commodore Computer club 2nd Meeting

Peter

My part in this meeting began on the Thursday. Shaun had asked me to come down a day earlier, as it looked like Nigel was not going to be able to make it. Happily, Nigel improved enough to be able to make it, but I kept my plans and set off 10am on the Thursday.

Loaded up in my car was my C64C system, which was to be used to demonstrate the 1541U+, a faulty Plus/4 which needed the keyboard either cleaning or replacing and a big box full of goodies to try and sell. There was also my sleeping bag and pillow. All the luxuries I could possibly need.



1541 Ultimate

The route I decided on was straight down the A1/M1, then onto the M42 for a short spell, then the M6 for 2 junctions. The run down was nice a quiet, a few road works but no real hold ups. As Shaun said, I left the M6 and junction 6, and was pulling up outside Lisa's house 10 minutes later. Nice quiet run that took around 3.5 hours with a pit-stop.

Shaun was there to meet me on arrival, coffee was made and I was introduced to Lisa and Ruby. Coffee in hand, I was shown into Shaun's Lair where I immediately felt at home. Commodore gear everywhere I looked, and a great 128D system with SuperCPU, RAMLink and various other bits of hardware sticking out all over the place. Heaven!

After a demo of Wheels, Shaun fired up Metal Dust, a game he knew I was dying to see first hand. After a few games, I started getting used to it so Shaun plugged in a 2nd joystick and we embarked on a 2 player game or 5. We were starting to make good progress when it locked up! We switched the system off to let it cool down, and started on the VIC-20.

I had a look at the 127 game cartridge and enjoyed Donkey Kong for a while. An excellent version of an excellent game. Cosmic Cruncher followed for a while. I've always had a soft spot for that game as it was the first ever game I played on a Commodore machine. We then got a couple of Shaun's Plus/4 machines out to test.

One was clearly faulty, garbage on the screen. The other seemed ok, but had keyboard problems.

I only had a stuck key on mine, so I brought it in along with the 1541U+. We downloaded some games onto SD and hooked up the

1541U+. After messing around with some commands, then reading the fabled instructions, we got it working. Shaun had heard about Tom Thumb and how it was an amazing game. We loaded it up, and it is indeed a technically great game. However, I won't be adding it to my list of games to complete before I die ;-)

We took a break and took Ruby to the shops, where we bought provisions for the night and beer.

After dinner, we had a couple beers to unwind. I turned in around 11, was quite tired after the drive down.

Friday was a quiet day. No one was due to arrive, so we just chilled and talked Commodore and the Club. After dinner we decided to head to a pub for a drink or 3. We went to a local, but quickly realised that we were probably the oldest there. We paid £1 for a game of pool (I wanted to play, not buy the table!!) and were quickly surrounded by a load of chavs. We finished our game and drinks and decided to leave while we still could.

We got on a bus and headed into town and went to Scruffy Murphies. Loud rock music and good beer. Oh, and NO CHAVS.

The Saturday of the meet soon came round. We started getting things ready to load up the car(s). After a short while, there was a knock at the door. Shaun greeted Nigel and showed him in. It was nice to finally meet him, we've been emailing for a few months now and now I had a face to go with the email address. He looked well considering his back problems, and I told him it was good that he could make it.

A short while later, Chris Snowden (www.commodore16.com) arrived. Introductions were made and we all hit it off from the start. We took a walk to the shops to get the refreshments for the meeting, and also some batteries. Shaun had looked at my "broken" DTV and after putting my batteries in his unit and discovering that his was now "broken", he came to the obvious conclusion.

When we got back we put the new batteries in my DTV and it worked straight away. So, expensive Duracell batteries are not always the best option. Sometimes cheap and nasty is the way to go. We also discovered that my unit was an NTSC unit, show he showed

me how to get into the hidden features, including a picture of the creators of the DTV.



Time was pressing on, so we started loading up all 3 cars, and after some jiggling around, everything and everyone was loaded up for the short drive to the hall.

Chris took the lead using his SatNav, so Nigel and I fell in behind. When we arrived, Chris unloaded his car and headed off to Birmingham to pick up Patrick who had flown in from Holland via Duesseldorf.

We got set up, Lisa and her helpers got busy with the refreshments and we started wandering around to see what had been brought. There were two large tables filled with goodies to sell, should anyone be interested. A couple items were sold, and the rest went back with Nigel for the club shop.

Soon after, Chris returned with Patrick. It's amazing that somebody had taken the trouble to come such a great distance for such a small gathering. True Commodore dedication. Patrick had brought with him his C116. It was the first time I'd seen one of these machines, and it was a very fine example. There were also two 1541-III SD card readers in use on the 116 and Plus/4 machines.



Commodore 116 the first one I have seen in the "flesh"

Soon after we got sorted Mike Dailly arrived with some goodies to donate. It was great meeting him, and hearing his views on the machines and software. Don't forget, Mike Dailly is certainly qualified to talk about this, he is responsible for Lemmings on the SNES, and is currently working on a great looking shooter for both the TED machines and the C64. XEO3 looks amazing, and I'll certainly be buying a copy when it is complete.

Shaun has his 128D set up and showed us Wheels and Metal Dust, before loading up Sub Hunter for the fantastic music. I had the Mathematica demo running on my 64 from the 1541U+ which attracted some attention.



Shauns 128D Setup with SCPU and RAMLINK

Mike said his goodbyes and left as he had to get back to his family, his day release over. Shortly after, Chris gave a little talk on how he developed Church of Death using cross development. It was great to see another Mac user. Macs and Commodore users both, we started talking about using the Mac instead of Windows and both concluded that that is the way to be.

Before we knew what had happened, we noticed that it was dark outside, and it was already time to pack up. I felt really sorry for Nigel and he had so much gear to take back for the shop. We loaded his car, leaving just enough room for Nigel himself, and headed back to Lisa's house. After unloading what was staying, Nigel said his goodbyes and headed back. His back was starting to play up again and he couldn't take his pills as he would have been unable to drive.

Chris, Shaun, Patrick and myself settled into the lounge to chat and reflect on the day. Lisa made us a fantastic casserole and we had a few beers. It was getting late by this time and we started to get tired. We headed off to bed in the early hours of the morning.

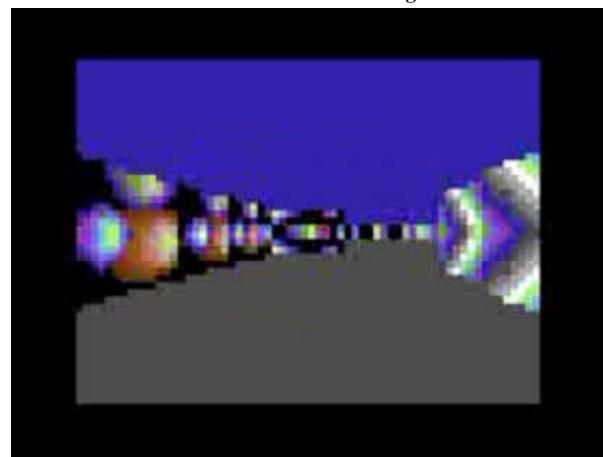
Sunday was the day for going back home, but I had a nice surprise. Shaun very kindly gave me a Commodore PET 2001 series computer. It is not working, so that will be a great project for me in the coming months. I'll be doing a blog on my progress and what I learn while doing this. We are hoping to have this and Shaun's 2 other PETs working for the next meeting, which I will be hosting in the North East, probably Durham. Watch the forum for further information.

All in all it was a fantastic weekend, and well worth the long drives. I'd do it again tomorrow.

Pete (Badders) Badrick
Commodore Computer Club (U.K.)



Group picture Less Mike showing how important crossing arms is while Demo watching



In the Beginning Part 12

Lord Ronin from Q-Link

Sound was a bit short for my rambling prattle, the problem was I ran out of lines as well as workable examples for the different C= machines. I mentioned that there are programmes to make sounds and music for you, there are also many books to teach you more on the subject of music. What I didn't mention is that in the users manual has data for the notes all 124, along with other information for their use.

"? OUT OF DATA ERROR IN 10".

We are now at page 91, moving into Advanced Data Handling. Time has past since that last sentence and now; my directory said that at that time there were just three blocks written in this section. While there are around 13 pages in the Commodore Users Guide left to handle before the appendices. Going over the information in that last section, before writing my editorial comments to the material. Man they jumped through the stuff faster than I did with the sound section. This might scare you, and so might the titles of the topics covered. Read and Data, Averages, Subscripted Variables, One dimension arrays, Averages revisited, Dimension, Simulated dice roll with arrays and Two dimensional arrays.

Alien and scary terms, when I went through it the first time I was scared myself. Scares some of the locals as well when they see the information, and not just for the first time. Yet the authors crammed that information into about 13 pages.

What we are going to do now is cover that information, along with my comments on my experiences in the use of these sections. Including things that we have done, tried to do and want to do in programming.

READ DATA is the first thing presented on page 92. Doing that sprite program we had the start of with Read Data. Type in the following little program, It will give an error msg. But that is a part of the lesson.

```
10 readx
20 ?"x is now : ";x
30 goto10
40 data 1,34,10.5,16,234,56
```

You can see that it is going to read the x variable. Then print it in that X IS NOW : print statement. Got that loop in 30. On screen you see that it prints the value of X as each one of the data entries on line 40. Then gives "? OUT OF DATA ERROR IN 10".

Each time it loops through the X part of the read. A pointer is given by the computer to keep track of the value to be used next. In this program the loop keeps going till there is not a single part in the data statement to use. So it is OUT OF DATA. That error is in line 10, this is Because there isn't anything more for it for it to read, as ordered in line 10. Well that is nice, but a bit useless, and I have seen many programmes give me that same error. A handful I have been able to fix. No not ones that I typed in, ones that came on disks to me written in Basic.

Parts of the DATA statement to understand. You have Got to start the line off with the word DATA. OK you can write it as dA for the shorthand. Each part of the line of information must be separated by a comma. This thingy here , and nothing else. The Program will crash with different error message to you. If you use any other symbols, Or it may give you a wrong data read out.

Now then what can go into a DATA statement line? Most of what we have dealt with already. Integer numbers, real numbers, you

remember the ones with the numbers to the right of the decimal point. Number in scientific notation. You can even put in words as well. But you CAN'T put in other variables or math functions. DATA Z\$,12+5,6/3,5*5 is RIGHT OUT!! Not going to happen as you would like.

Next type in is a little different because it uses afor next loop to keep that out of data error off the screen.

```
new
10 forx=1to3
15 reada$
20 ?"a$ is now : ";a$
30 next
40 data i, did, this
```

Yeah it will print out each line, going down the screen saying what a\$ is each time. What happens if you add

```
50 goto10
```

Right the return to 10 starts the program back but you are going to get the out of data error message, not a good thing; So in order to reuse the data just add this line.

```
45 restore
```

You don't get the out of data, and the program keeps going on forever. What happens is that the data pointer is restored each time through. Not a big thing at this time for you to follow.

Next is a more practical programme for read/data.

```
new
10 t=0:ct=0
20 ifx=-1then50
25 ct=ct+1
30 t=t+x
40 goto10
50 ?"there were ";ct;"values read"
60 ?"total = ";t
70 ?"average =";t/ct
80 data 75,80,62,91,87,93,78,-1
```

You can probably guess what is going to happen in some degree. Programme is going to read all the data and then print out the number of values read. In this case it is 7 then by the math in line 70 the program prints the average after printing the total. Here the total is 566 and the average is 80.8571429.

The print statements are easy to see. Setting the T and CT variables to 0 at the start is simple. In fact just about all of what we see in that program is stuff we have done already, everything except; there is this strange thing in line 20 about a -1. Also in line 80 is that -1 at the end of the data statement line what does that one mean? They call it a flag, it is the indicator that the read/data part is over. The Program sees that -1 and then goes to line 50. Where it does the print and math work. Line 20 is the check for that flag of -1. No it doesn't have to be a -1 in your programme. Can be anything that isn't a part of what you are placing in the data statement, I have seen words and strange numbers used in some type in programmes. Anything that is out of sorts with the theme of your programme can be used for a flag. You can make a large collection of data statements, several lines worth, and not worry about the count thing. When using this flag concept.

Now here is the last part they give on the read/data part, showing how to assign data to a variable.

new

```
10 read n$,a,b,c
20 ?n$;"s scores were: ";a;" ";b" ";c
30 ?"and the average is: ";(a+b+c)/3
40 ?;goto10
50 data mike,190,185,165,dick,225,245,190
60 data john,155,185,205,paul,160,179,187
```

Running this and you will get something like...

```
mike's scores were: 190 185 165
and the average is : 180
```

Then the other three follow below that one. Difference in this one is that the order to be read is set in line 10. Starting with n\$ <name string> and following with three sets of variables for the integers. This data is written in the data statement lines in the same manner a text and then the numbers.

Small little program that is to give a basic understanding of how this works. Obviously it can be bigger and things like entering the data from a programme can be done as well. I have seen read/data statements in many a Basic programs that I saved from Q-Link. The first one that I tried to work upon was an Inn Menu for Fantasy Role Playing games. Altering the text to fit different games I messed it up royally. You see I didn't understand the concepts of the flag and the read a certain amount of data Or the point about correctly placing in text in the text area and numbers in the number area. Now that I have learned more I really should try that project again.

Subscripted Variable, is the next part; ends right after that above program. Don't let the words freak you out. May not have been the best choice of terms to use for this type of variable. What this leads into is one of the mega important parts of Basic programming, at least from the part that my group is trying to do at this time. I'll talk a bit on that later.

Variables we have seen are like F\$, F%, F. Covering the text integer and real numbers with the floating decimal. We also learned that you can have two characters for a variable. Like F1\$, F1%, F1. Or FA\$, FA%, and FA. That can give us a mess of variables. Little slots of memory that we can fill in with things. Now we are going to expand upon that idea.

Z(1) is a new thing. Got that the Z is the variable. What then is the (1) part? That is the subscripted section. Even tells us how to say it as "Z sub 1". Really this is different than Z1 or even Z. Z(1) is subscripted. Yeah I know first time through this is confusing. They give a chart thing to try to show you what they mean. Looks something like this,

```
Z(0) { }
```

```
Z(1) { }
```

```
Z(2) { }
```

```
Z(3) { }
```

```
Z(4) { }
```

As an attempt to show you the memory aspects of a subscripted variable. The way that I see it is that there is in the above 5 <0-4> slots in the variable Z. Slots that something can be put into.

```
10 Z(0)=25:Z(3)=55:Z(4)=-45.5
```

A modified form of what they present next, as a way to see the thing about putting stuff into those slots. Modifying the boxes again, it would look sort of like this in the memory for Z.

```
Z(0) { 25 }
```

```
Z(1) { }
```

```
Z(2) { }
```

```
Z(3) { 55 }
```

```
Z(4) {-45.3}
```

Personally I like the effect of the expanding box to illustrate the fact that the size isn't set to just one or two characters.

New word now, ARRAY. A group of subscripted variables is called an array. This one is a one dimensional one. You can make multidimensional arrays as well. Mind boggling isn't it? Gets even more so, as you can be more complex by adding other variables or "computations". These are CORRECT subscripted variables.

```
Z(X) Z(X+1) Z(2+1) Z(1*3)
```

Got that? Well it is still weak for me. Point is that here we have a lot more power than expected for our PC. Even if you don't programme, you can see now that we have more than people think we have, these days. Now add to this what we have seen already. Couple this sort of subscripted variable with things like Input and Read statements. Right it is past us at this time. They don't even try to explain the how it is done part, only mention it on page 97. New type in program for you.

new

```
5 ?"<shift clear home>"
10 input"how many numbers :";x
20 for a=1 to x
30 ?"enter value # ";a;:input b(a)
40 next
50 su=0
60 for a=1 to x
70 su=su+b(a)
80 next
90 ?:"average = ";su/x
```

Right it wants to know how many numbers you are going to insert. Then it will have you enter each one. Printing it to the screen. Finally it will give you the average of all the numbers.

When you run this I want you to say that you want 5 numbers and the numbers that you type in will be 125 167 189 167 158. The reason for this will be shown a few paragraphs below. These are the numbers that are presented in the Commodore users guide.

We have x as the counter for the loop. The values are entered and go into the subscripted variable of b. Every time it loops through the variable of A is increased. That is pretty straight forward. Ah but what is happening is not that straight forward with what we

understand at this point. Certainly a is changing. But not the way we have learned before. First time through the loop. We have a=1. OK got that part, the difference is that it is entered into the subscripted variable of b. So it would be written, in the manner of the boxes above as B(1). Next loop through the value of a=2. This is entered in the b subscripted variable looking sort of like B(2) from the above chart theme thing. This goes on till the number of entries is completed.

Remember that when we discussed the Input and the Read stuff, there wasn't any way to get back or retrieve the inputted information? OK we never tried in the first place. The Commodore users guide is the one that brings up that subject. Well type in the following, while you still have the above program in memory.

```
FOR A= 1 TO 5: ?B(A);:NEXT
```

The reason that I asked you to type in the amount of numbers as 5 and those specific numbers. Simply is to illustrate the point of this line. You will see on the screen...

```
125 167 189 167 158
```

The numbers that you typed in for this example. See they are stored. Now try out the programme again and use your own amount and numbers. Then adjust the line above. Note it isn't a line number, just something you can type in on the screen. Alter the 1 to what ever you chose for the amount. Then see what happens.

Welcome now to DIMARRAYS. Well they say Dimension on page 98. But I am cutting to the chase here. Because as soon as you tried more than 10 numbers in the above example you got an error. Probably said "DIMENSION ERROR". Makes you wonder what sci-fi movie you entered. You can only have 11 <0 to 10> elements in a one dimensional array. OK add the following to the programme.

```
5 dim b(100)
```

You just told the C= that you will have a maximum of 100 things, or elements in the array. Can be bigger, I am not going to say how big at this time. But it is BIG. Better to make several different DimArrays than one gigantic one. OK I'll confess the maximum that you can put in a DimArray is <hold onto your hats> 32767. If you do, your program is bulky and will be very slow.

OK then lets run that program with the new line 5. Then we are going to change it to...

```
5<press return>
```

Right that kills the line completely. Have to do that to enter this one.

```
15 dim b(x)
```

Now what ever x is will be the number of elements in the dim array. Note here is that once an array has been <dim>ensioned. It can not be redone in another part of the programme. However you can have more than one dim array in a programme and even on the same line. Following is just an example, NOT a line to type in!

```
10 dim c(20),d(50),e(40)
```

Soon we are going to type in a new programme. But first this is something that isn't in the users manual. Comes from a book that teaches making text adventure games. Put the DIM ARRAYS at the start or as close to the start of your programme as possible. That

Commodore users guide uses several dim arrays for the games. These are all "declared" in the first several lines of the programme. Though defined a bit later in the programme.

An example from a programme that I modified. No it isn't really that impressive, remember I am a lamer at this stuff. I just used some of the things that we have or will cover in this users manual. Well the program is one that I saved from Q-Link. Done in Basic and it for the creation of 1st edition of Advanced Dungeons & Dragons monsters. Saved it to disk and printed it out. I did a few things to it, such as adding some stats. Those where different variables, and we covered them already. What I did that applies to this part of the series, is that I created a Dim Array. Originally, after you typed in the stats for the monster for your game. You could write a description. However the author made a one dimensional array. Meaning that I had only 11 lines max for a description of the monster and its habits. Not enough when I am copying them out of old and crumbling magazines that are over 25 years old. What I did was simply turn that into a Dim Array looking like DIM C(40). C being the existing variable. Now I have many more lines for a description. See it isn't that fantastic. Just a simple re-write of one line. Oh yeah that was in the first 15 or so lines of programming/code.

```
new
1 rem dice simulation:?"<shift clearhome>"
10 input"how many rolls:";x
20 forl=1tox
30 r=int(6*rnd(1))+1
40 f(r)=f(r)+1
50 nextl
60 ?"face","number of times"
70 forc=1to6:?c,f(c):next
```

Running this and you will get a list on the left of the faces of a D6 <6 sided die> and the number of times that face appeared out of the total number of times that you entered for the number of rolls. F(R) is the array for the random number generator in line 30. L is the variable for the number of times the die is to be rolled. Line 70 has the C= 1 to 6 Then it wants the value of C printed. The faces on the left of the screen. But see that next there is the array of F(C). Bit confusing as the explanation in the Commodore users guide is short and lacking in depth. My understanding, and it is quite possibly wrong, is that the F(R) is the array for the number of times and the F(C) is the array for that number for each face or side of the die.

They next turn the page to 100 and show the same programme using IF THEN statements. Doubles in the number of lines used. More than double in the space in the computer that would be used.

TWO-DIMENSIONAL ARRAYS is the title for the next part. They start off with showing us what it would look like Z(4,6) is a two dimensional array. First part is the array name and the other two parts separated by the, symbol are the subscripts. Written out in a

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|----|---|---|---|
| 0 | | | | | | | |
| 1 | | | | | | | |
| 2 | | | | | | | |
| 3 | | | | 71 | | | |
| 4 | | | | | | | |

matrix type box with the following value of $Z(3,4)=71$ would look like the following in the above defined two dim array.

First subscript is sort of the row and the second one is sort of the column. You can get the idea that there can now be a lot more installed than before. OK there are some rules to follow in creating a two dim array.

Gotta dim the array: $\text{dim}Z(4,6)$ as the first part. That sets up the areas for use. Next you need to assign the data that is going into the array sort of like: $Z(3,4)=71$. Next assign the values to other variables like: $ZB=Z(1,1)$. Then print the values. Complex, confusing it doth sound at this point. Trust me that I dig that part, no not the variable thing, that it is complex and confusing.

This is not something you can get in one sitting. Takes several goings over, and other sources of information to slowly click. Then you ask yourself why you didn't catch it at the start, since it is so simple. Or so I have been told. I'm still at the part of the light is just turning on for these arrays. My advice is to just ride this part for now. Work on the ideas as you feel comfortable. Now then lets move onto a type in example.

new

```
20 ?"<shifted clear home>"
30 forr=1to4
40 ?"Question # : ";r
50 ?" 1=yes 2=no 3=undecided"
60 ?"what was the response : ";
61 getc:ifc,1orc>3then61
65 ?c:?
70 a(r,c)=a(r,c)+1
80 nextr
85 ?
90 ?"do you want to enter another":"response (y/n)";
100 geta$:ifa$=2then100
110 ifa$="y"then20
120 ifa$<>"n"then100
130 ?"<sifted clear home>";"the total responses were:";
140 ?spc(18);"response"
141 ?"question","yes","no","undecided"
142 ?"-----"
150 ?r=1to4
160 ?r,a(r,1),a(r,2),a(r,3)
170 nextr
```

Give it a run and see what happens. You end up with 4 questions. Well not exactly, since they are never stated as what is the question. Then you have a choice of one out of 3 for a reply to each of the 4. This is going to be tabulated at the end with how many times each of the possible three answers were given for each of the 4 questions. Line 142 is a bit of a problem in getting the - lines correct. Don't worry if you flubbed that one. I always do myself.

Line 140 has a spc(18). That takes the writing to the 18th space on the row. Those comma things in line 141, space out by 10 remember. Making columns for the final print out on the screen. 100-120 is our simple get a reply from the kb. In fact most of this we have seen and done in some form already.

The Commodore users guide covers this entire programme in three paragraphs, and that is on one page, with a quarter of the page blank and a chart at the top of the page. Right; not a lot of data for you to read and learn how this all works. That is something I was recently talking about on the some IRC chats, how it is easier to learn with

others than by yourself, and the lack of in-depth information in the user's manual. Before the comments though, lets look at the array stuff in this programme.

Really the array stuff is the new part. We see an array at line 70. BTW that will update the element. So says the rem statement on that line in the Commodore users guide. What type of array is it? That is a bit harder to see. $A(R,C)$ this tells us it is a two dimensional array. Because of the two elements inside. But now then, what are the parameters of these elements? That takes a little bit of looking at the programme. In the array of $A(R,C)$ we look through the programme and see that the R is in line 30, and is for the number of questions. There are 4 questions or 4 elements. That makes the $R=4$, so $A(4,C)$ is what we have so far. Line 61 sets up the ends for C being 1-3. OK that must be the yes and no and undecided part. Line 65 prints c. C then is 3 and that means our array of $A(R,C)$ is $A(4,3)$. Right a 4 by 3 two dimensional array.

At this point it is important to know that the first row and column of $A(0,0)$, or the first row and column of any array is mainly never used. So is the impression from the Commodore users guide. Line 160 prints the values, the values of the elements in $A(R,1)$ $A(R,2)$ and $A(R,3)$ The numbers for the yes, no and undecided. Yeah it is confusing and complicated at the start. I honestly at this time am still learning this section.

Outro will have some more statements on this part. Last of the Commodore users guide now. Appendices and these are a list of accessories for the 64, Advanced Cassette Operation, Basic term explanation, Key abbreviations, Screen Codes, Ascii & CHR\$ codes, Memory Maps, Math functions, Printouts, Programmes to try out, Basic conversion, Error Message, Musical notes, and if you have a copy and are lucky. There is a quick reference card at the end. May you find a copy and with this series, have some fun in programming.

COMMODORE FREE

I tried to put the Commodore Users Guide into the Loadstar Quick menu, you can download a copy from the website www.commodorefree.com because of the size I had to split the manual up into 2 d64 images

<http://www.commodorefree.com/tools/guide/guide1.zip>
<http://www.commodorefree.com/tools/guide/guide1.zip>

The text is taken from Project 64 version of the user guide in plain text so many of the diagrams are converted to text art



COMMODORE COMPUTER CLUB (UK) RULES AND REGULATIONS

These rules were agreed on Saturday 26th July 2008 at our first meeting held at Blackburn, Lancashire. They cover all rules of the running of the Commodore Computer Club (UK), herein referred to as 'the club', and also regarding complaints made to the club, and membership of the club.

(1) Membership subscriptions, raising funds and re-selling items.

(a) All members will pay a membership fee as follows: £3 for six months, £5 per year and life membership at £30. This fee will entitle the holder to free entry to the meetings, and special limited areas of the website, such as private forums and exclusive downloads, should we get anything exclusive to download. Membership will always be back-dated to the start of the month in which the member took out the subscription, so that everyone joining in the month of April for one year will see their membership expire on the 31st of March the following year.

(b) We should have a 'Commodore Computer Club Shop', which will stock all of the latest hardware mods and sods for Commodore computers where possible. To stop the 'Maurice Randall' effect, in which the club will have to repay people for not receiving their goods because they haven't been delivered but have been paid for, items will only be on sale if they are in stock.

(c) There will be two prices, one for members (cost of item + postage and packing + 10%), and one for non members (cost + postage and packing + 20%).

(d) Any members that do work for the club, organising events, donating items for auction, coding, or are otherwise active, with exception to posting on forums and turning up to meetings, will be considered for free membership and/or lifetime membership on merit based only on work they have done for the club. In certain instances, will include what they have contributed overall to Commodore computing or gaming during their life-time, should any 'Commodore legends' show sufficient interest to join the club.

(2) Events, software and other developments.

(a) Any money that is raised by the club should be used primarily for setting up events, or bolting onto other events as appropriate This is to go towards, or cover costs of van hire, hotels, and food and drink, so the person or people who are willing to travelling to these events, man stalls and generally promote the club and its work are not be out of pocket as far as possible.

(b) Profits made from items sold at events should contribute to cover the costs of attending, or hosting, and/or expenses accrued during the event. This will not include monies raised from membership subscriptions paid for during the event.

(c) The club should also seek to raise money for the purpose of developing hardware and/or software that will benefit Commodore users in the UK and world-wide, and such items could therefore be sold through the club Shop.

(d) Payments to developers who are commissioned to work on behalf of the club should not be made in advanced or up front unless otherwise agreed by the treasurer and chair-person, and any other two members. This should be openly discussed with all members either in private members areas of the site, or at an organised meeting as appropriate.

(e) Hardware that is commissioned on behalf of the club which reaches production should be sold at a small profit, and monies raised to put back into the clubs funds.

(f) If it is agreed that the club should commission entertainment software, the productions should be available to download for free from the site for members only. Real-media versions should also be sold through the shop with non-members able to buy copies, though at a higher price than members.

(g) Any software commissioned by the club will either be purchased outright, paying the programmer an agreed fee on completion, or paying a lesser fee and splitting the profits at an agreed rate. This should be discussed on a case-by-case basis. The chair-person and treasurer, and two other members, must agree which method should be implemented.

(3) Meetings and monies.

(a) The club should hold an annual general meeting in which members have a say in its running, and are able to make suggestions and table official club business for the year ahead. Membership subscriptions should be reviewed at the annual general meeting, and any price increases must be agreed by the chair-person, treasurer and at least two other members.

(b) There should be an annual audit of the clubs finances, with a news-letter at least every three months. The audit should be published before the annual general meeting, and this and the news-letter should be available to current members online in the private member areas. Former members may request this information, which will be granted on a case-by-case basis.

(c) Members will be able to attend any events that organised and run by the club for free when ever possible, whilst non-members will pay a small signing in fee of at least £2.50. With agreement with other event organisers, and at events that the club is attending in an official capacity, we will work towards getting members a discounted entry fee.

(d) All monies raised will go back into club funds.

(4) End of line.

(a) If it is apparent that the club is not running within its means to the extent that it is likely to fold, or that legal action against it will lead to the club being dissolved, all club assets should be sold or auctioned off, the monies raised pooled and members will be refunded their current subscriptions based on the length of time they have been members. The longest-serving paying members will be refunded first as appropriate, either partially or fully depending on the financial circumstances at the time. The newer members will be dealt with last.

(b) Personal donations to the club's funds can never be fully refunded, and are not guaranteed to be paid back at all depending on the circumstances.

Club complaints procedure:

Phase 1: Where a complaint is made against the club, or one of the club members, there should initially be a private apology between the club or individual and the plaintiff. This apology should be for 'any undue harm or upset caused', and will not amount to an admission of guilt or a retraction in any way. The club will not be able to force any of its members to make this initial apology except in the instance that the individual has clearly and admittedly worked on the clubs behalf in the matter specifically relating to the complaint that has been lodged.

Phase 2: The matter should then be investigated to establish the facts. If it is deemed that an individual club member has not been acting on the clubs behalf with regards to the specifics of the complaint, then this becomes a personal matter between the two parties. The club should therefore stop any further investigations or involvement in the matter.

Phase 3: If the complaint lacks any real evidence, or it is felt that the findings are not conclusive, then the matter should be closed. Neither the club, nor any of its members, should therefore discuss the matter publicly. All findings should be reported to the plaintiff, and the matter should be considered closed from the club's point of view.

Phase 4: Where a complaint is upheld, a public apology and/or retraction should be published through the official website, and in the newsletter. The club should also give the plaintiff the opportunity to give his or her point of view through the website and/or newsletter as appropriate In this instance, the case will be considered closed from the club's point of view unless the plaintiff wants to take the matter further through due legal process.

Emergency phase: If at any point during this process the plaintiff feels aggrieved to the extent that he or she instructs a solicitor to take the matter up against the club or club members who have clearly being acting on behalf of the club in this instance, the club should then consider its legal position on the matter, and a meeting should be set up with the principle members of the club within two weeks of receiving legal notice to discuss the matter, and what to do next. Obviously, one would hope that any complaint would ever get to this stage.

Membership:

People who join the club will have a personal membership to it. They may not join the club as a company, publisher or software distributor or hardware vendor.

Newsletter and reviews:

The Commodore Computer Club (UK) is an independent user group which will review and stock all appropriate wares. We will do so on merit only, and invite all members to have their say about any literature published through the newsletter or any reviews written on behalf of the club. We will invite hardware and software vendors and publishers to have their say on reviews written, and we will publish their comments through the newsletter.