Commodore Free

commodore computer club

C128 RGBI to S-Video

Prototype Production Unit #004
03/10/10 - assigned to Miro Karkuš
allan.bairstow@btinternet.com

RGBI
IN →

S-Video
OUT
↓

9v DC
IN
↑

Mk2db4a

DLOAD "TEST"

CONTENTS

EDITORIAL

Welcome back

I hope you enjoyed the Holidays of Christmas and New Year, and I would like to take this opportunity to wish you all a Happy and prosperous New Year.

Now without anymore small talk to spoil the text I will, with your permission give you a breakdown of what tantalising offerings we have for you in this edition of Commodore Free magazine.

From The Rear

It's the usual news, as this issue tries to catch up from December; this will be an extended feature, to cover up for the missing December issue.

The Rear Admiral News section is a really good start if you are still on the Christmas Blues bus or suffering from SAD (Seasonal Affective Disorder http://www.sada.org.uk/). I think we all suffer a little from this, however I do feel sorry for the people that this disorder is really a problem for. Still **The Rear Admiral** feature just may cure the problems away, well ok, maybe not.

PROGRAMMING

John Fielding enters programming overdrive as he explains out to death the programs that were listed in Commodore Free issue 45, and then he gives us another application to try out. It all feels like 1980`s again with type in listings,

INTERVIEWS

I managed to interview SEUCK game programmer Alf Yngve about his latest creation 1,000 Kung-Fu maniacs, and there is a review of the game in this very issue.

New Features.

More new games appear in the shape of **EWAY10 RELEASES 3 NEW C64 ADVENTURE GAMES**

Featured again in the news section so I won't elaborate here anymore.

I remember it well

I am also putting together an item about the shops where I used to buy Commodore products and have tentatively penned this as a "what happened to" style feature. Where the shops where what they stocked, and what actually happened to them after Commodore went bust. I thought about the feature after passing 2 of the shops over the Christmas period and as nostalgia welled, I remembered buying my first external Commodore Amiga disk drive and even the assistant stating the obvious after the sale like "don't turn it on and off while connected"

And you may have noticed the new logo

The old one didn't show up to well on some monitors, especially with the black colour scheme, I am no artist and you may well think it looks a little poor, but I thought it was time for a change. You are welcome to send in some of your own example Commodore Free logo`s for possible use in the magazine. I would love a retro styled 8-bit one, but unfortunately my efforts have been rather poor not being in anyway arty.

That's all from me now see you next month

Regards
Nigel (editor Commodore Free)

NEWS

Games That Weren't UPDATE

GTW 64 has a Xmas update with the following games. Putty from System 3 (1991), Undead (1990), An preview of Pulse by David Green (1990), Scimitar from Cyberdyne Systems (1990), More demos of Deadlock from System 3 (1990), Music recovered of Armalyte 2 from Thalamus (1990), Graphics found of Half Life from Cyberdyne Systems (1990), Music recovered of Savage Platforms and bits founds of Turn Change from Commodore Scene / Mike Berry (1996).
http://www.gtw64.co.uk/

COMMODORE SITAR
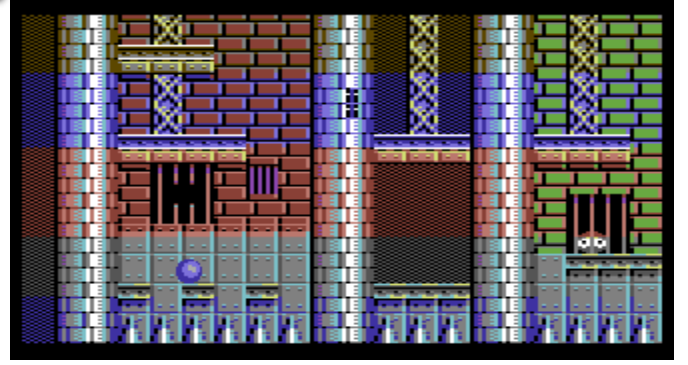
http://www.youtube.com/watch?v=1LZUH0p6cXc
A link to a YouTube video showing a demonstration of Jordan Bartee playing Giana Sisters. He uses a modified Commodore C64 to play the music and has added a second SID (6581), keyboard and the MidiBox SID platform to his C64

Here is the videos text
A quick demo of a keytar I built out of a Commodore 64. It uses two SID 6581 chips and was made using the MidiBox SID platform from www.ucapps.de . Better video to follow later

ZoomFloppy ready for pre-order, ships Jan 7

The ZoomFloppy connects 15x1 IEC drives (including nibbler parallel port) to USB. This allows reading or writing D64 or G64 images from your PC with OpenCBM or nibtools, including copy protections. Jim Brain is now taking pre-orders for $35 USD. It should be shipping January 7.

http://www.jbrain.net/products/ZoomFloppy.html

Product Description
Archive your Commodore floppies (and CMD HD partitions) to your contemporary PC, without the hassles of external parallel ports, IRQ settings, special adapters, bulky cables, or the alphabet soup of disk drive cable adapters. ZoomFloppy replaces all X*-1541 cables with a simple USB-based hardware device guaranteed to work with today's machines, using today's multitasking operating systems. Simply plug into any free USB port, connect your CBM drive via a serial cable, and transfer data to or from your software collection.

ZoomFloppy Features:
•Attach any CBM serial drive to your PC
•Works with Windows, OS X, and Linux
•Open Source solution, no vendor lock-in
•Active development community
•Minimal footprint, no external power supply needed
•Parallel access hardware support
NOTE: ZoomFloppy requires an IEC cable (sold separately) and a USB to mini-USB cable (available any many computer stores).
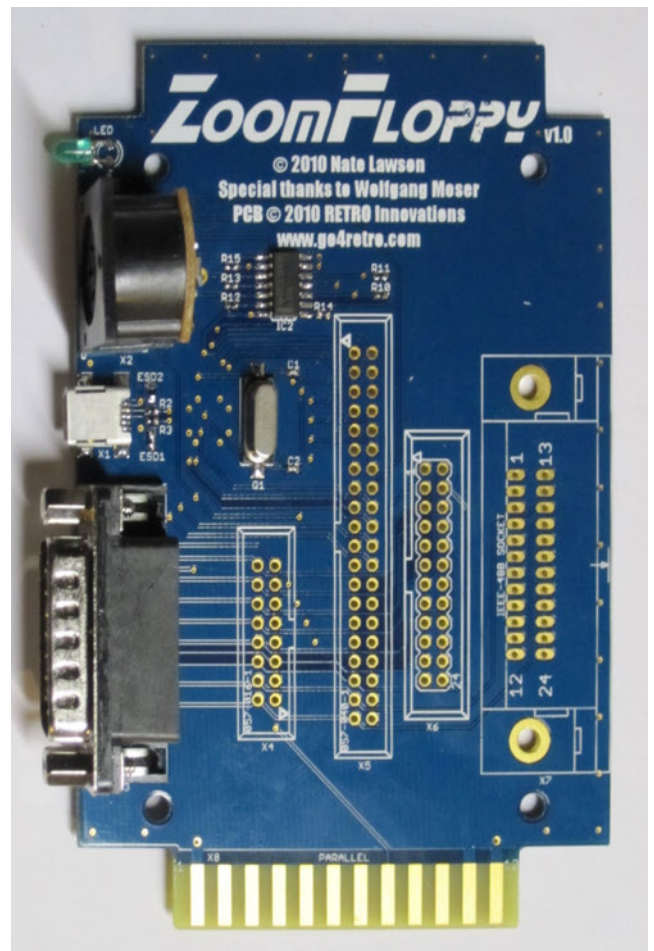
SCACOM Aktuell #20/21 out

Issue 20/21 (October/December 2010) of the German language magazine SCACOM.aktuell is available. Download it at www.scacom-aktuell.de.vu , or read it online.

High Voltage SID Collection Update 54

Another Update to the growing collection of Sid tunes, after this update, the collection should contain 39,626 SID files!

This update features (all approximates):
933 new SIDs
217 fixed/better rips
5 PlaySID/Sidplay1 specific SIDs eliminated
21 repeats/bad rips eliminated
561 SID credit fixes
258 SID model/clock infos
28 tunes from /DEMOS/UNKNOWN/ identified
6 tunes from /GAMES/ identified
41 tunes moved out of /DEMOS/ to their composers' directories
9 tunes moved out of /GAMES/ to their composers' directories

http://www.hvsc.de

ARCADE RETRO GAMING

MERRY CHRISTMAS!

We would like to thank all Arcade Retro Gaming friends for supporting us in our efforts to preserve the classic computer and classic game consoles for the future.

Thanks to faithful friends like yourself, this past year has been very successful.
We have a Christmas present for everyone who owns already the MCC-216

The ATARI 2600 implementation is now available online for **FREE** download.
As if this isn't enough another release of the C64 is planned between Christmas and New Year!

ZZZZZZZZ   HAPPY RETRO GAMING   ZZZZZZZZ

http://www.arcaderetrogaming.com/shop/page/12?shop_param

All NEW MCC-216 now come with the C64 and ATARI 2600 core
We are pleased to announce that the Multiple Classic Computer™ comes with the Commodore C64 and ATARI 2600 re-implementation factory installed

2011 C64-RetroInvaders calendar: NEW GAMES

Here we go for the New Year a calendar for every month of the year with recent Commodore 64 games.

Download it, print it, cross off dates and amend it but most of all enjoy it .

http://noname.c64.org/csdb/release/?id=97007&rss

AMIGA FOREVER AND C64 FOREVER 2011 RELEASED

Amiga Forever (http://www.amigaforever.com) and C64 Forever (http://www.c64forever.com) are the easy to use emulation, preservation and support packages published by Cloanto, Commodore/Amiga developers since the 1980s.

The 2011 version introduces features like multiple snapshots (saved states) and further improves usability, performance and memory consumption. As a result of various optimizations, in some scenarios loading times are reduced by 90% and runtime memory usage dropped by 80%

Amiga Forever and C64 Forever work side by side to play games directly from a variety of emulation sites, as well as from original media like CD-ROMs. Content is recognized using the built-in Retro Platform Library database, which helps identify and automatically start the correct hardware configuration (Amiga, CDTV, CD32, C64, VIC 20, different PET/CBM models, etc.)

In addition to the hundreds of titles which come preinstalled with the software, tens of thousands of games and demo scene productions have been released by their authors for publication on preservation sites. Beyond one-click access to this precious legacy of the 1980s and 1990s, the packages aim to introduce users to the computer demo scene culture and to present other items of historical interest. The Premium Edition of Amiga Forever contains more than five hours of video footage on two DVDs.

Amiga Forever and C64 Forever passed official Windows 7 logo tests on both x86 and x64 systems. Prices start from $14.95 for C64 Forever, and $29.95 for Amiga Forever (downloadable Plus Editions, with discounts for upgrades and for bundles of both packages). The 2011 upgrades are free for all 2010 customers.
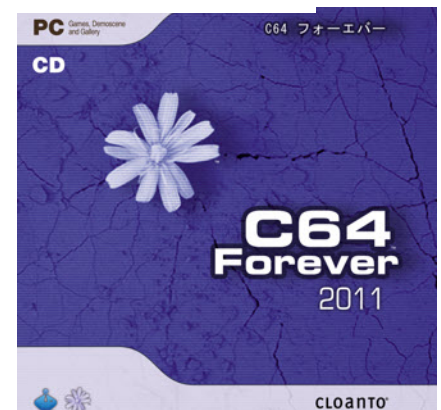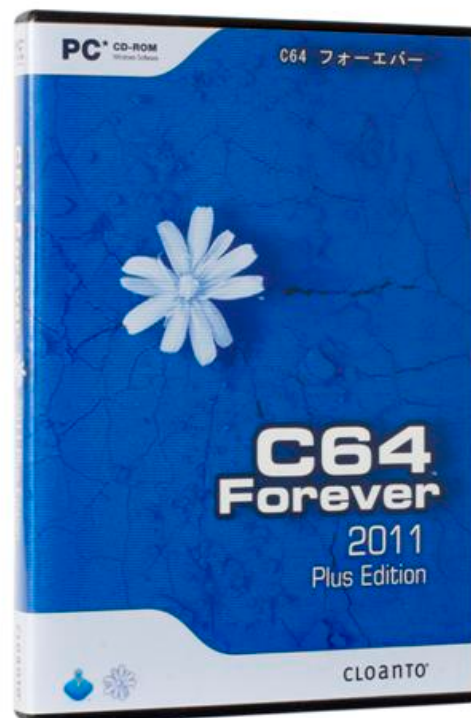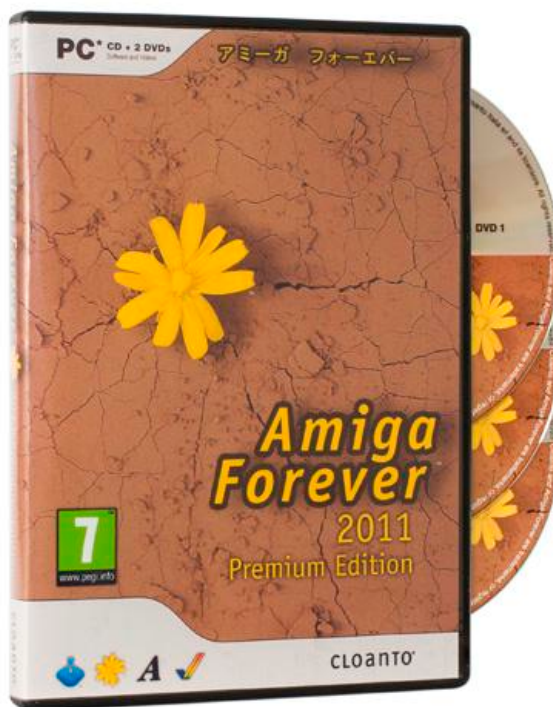
For more information:

Amiga Forever
http://www.amigaforever.com
http://www.facebook.com/AmigaForever

C64 Forever
http://www.c64forever.com
http://www.facebook.com/C64Forever

RetroPlatform Project
http://www.retroplatform.com

PLUS4: CLUB INFO 120 RELEASED

The sixth and last Club Info issue of 2010, the most popular C264 series disk-magazine, is here to warm your cold winter feet! Erich/Ultd packed the two disk sides with a lot of goodies. Enjoy.

Side A
Is filled with various German-language articles. There are tips and tricks, game descriptions, a user's guide to 3D Construction Kit etc.

Side B
Has recently released stuff as well as some new releases: Maths Battle, Hman +4 and Thunderbird Digi among others.

Download it now from here.
http://plus4world.powweb.com/dl/mags/club_info/club_info_120.zip

---

ATTITUDE DISK MAGAZINE

http://www.cactus.jawnet.pl/attitude/
ATTITUDE #11 WINTER EDITION IS HERE
Europe's #1 is back again! Attitude #11 delivered for your sole pleasure by the joint forces of TRIAD and Oxyron on December 12th, 2010 is now available for download. Get it and share your thoughts about the new edition with us!

---

dtv2ser 0.5 released

The dtv2ser device published in previous dtv2ser releases required the creation of a custom PCB with an ATmega8 and FT232 chip. Therefore, only a very limited number of users were capable of recreating the device.

To make the device accessible for a wider user base I have ported the firmware to the Arduino 2009 boards. These boards are easily available almost everywhere. The dtv2ser modification only requires minimal HW modifications and soldering skills.

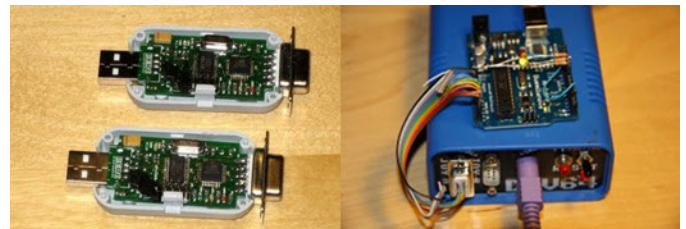The new release 0.5 contains this new firmware and detailed instructions how to build your own device. On the software side nothing has changed - only some minor bug fixes were applied.

The dtv2ser Arduino page:
http://lallafa.de/blog/dtv2ser/arduino/
The dtv2ser Homepage: http://lallafa.de/blog/dtv2ser/

Have Fun building your own dtv2ser device :)

1541 Ultimate FIRMWARE VERSION 2

http://www.1541ultimate.net/content/index.php

-83 new batch Ultimate-II's have been shipped.

-Firmware 2.0 FINAL is out, and available for download! Go to the download page to get it now and see what issues have been addressed. It is a major update, and highly recommended to install!

-All additional FPGAs were shipped to the assembly company. Hopefully they will release more completed boards before the end of the year. They said they would, but they couldn't tell me how many would be finished.

-First modified tape adapters shipped.

-It is now possible to order accessories for the 1541U-II. This is possible through the "Order Status" page. In case your unit has already been shipped, please create another order of 0 (zero) 1541U-II units, and add the accessories to that order.

-Tape connector boards have been received. One board clicks under the 1541-II case; the other holds the 2x6 pin tape connector and tape feed-through. The two boards connect to each other using a standard 10-pin flat cable. The part that clicks under the 1541-II case can also be used to power the cartridge for stand-alone mode (requires future software update).

-It's now possible with 2.0RC9 to capture tape data to a .TAP file!

-Payments have been processed until December 7, 2010.

-Payments for the new batch(es) are now accepted.

-Firmware 2.0 also runs on the Ultimate-I! (Of course, only the features that MK1-hardware supports will show.. no RTC, no USB, no C2N (tape), etc...)

Ultimate-II is open source! See the documentation on the download page on how to set up the build environment. This document (which is work in progress) also explains how the FPGA and software is structured.

BOH update14 released

The year could not end without BOH players getting one more nice gift: this 14th update ups the total number of missions to a whopping 50 (for many tens of hours of play), brings a few enhancements and makes life easier to who enjoys creating his own missions!

More specifically, this update:
* reworks the advanced auto mapper (the realistic mode has been replaced with the full screen view, which allows to see more; the controls have been made more comfortable);
* improves the video modes suggestions;
* allows terminating map lines with CR+LF;
* improves error logging;
* improves the graphics of blocks in theme "C64";
* adds 5 new missions;
* retouches slightly a few missions;
* retouches slightly the Spanish translation;
* updates the manuals;
* applies various other changes.

The update can be downloaded for free from the DOWN-LOADS page of the official website.
The demos have been updated to reflect the changes.

About BOH

BOH is a unique retro-flavoured puzzle-riddled maze game spiced up with thrilling action.
You move in claustrophobic, mysterious, dangerous battlefields searching for the Evil Masters, who throw countless enemies at you until you discover and face them in the final battle. Although your quests are made slightly less hard by the power-ups scattered all around, carrying out the missions demands lots of concentration and quick reflexes.

With BOH the fun never ends: new missions can be added anytime and you can even create your own! And, as if that was not enough, you can also customize entirely its audio-visual aspect!
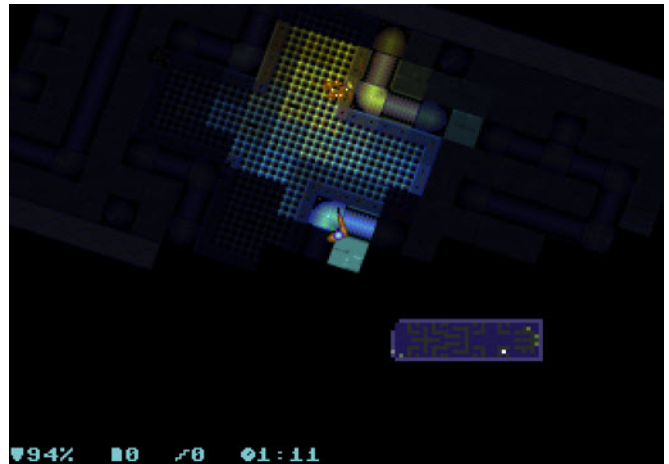BOH has been created with lots of passion and care to offer, with its old school style, a fascinatingly different video gaming experience.

Website: http://www.bohthegame.com
Facebook:
http://www.facebook.com/pages/BOH/99598418583
Twitter: http://twitter.com/bohthegame

MORE COMMODORE VIDEOS

For those who like Amiga computers and who could not get to the Amiwest Show 2010 on October 23-25, I've posted my entire video series of presentations and keynote speeches. Here they are --

Around the Amiwest Show 2010
http://blip.tv/file/4432511

Carmona at Amiwest Show 2010
http://blip.tv/file/4428812

Krueger at Amiwest Show 2010
http://blip.tv/file/4428423

Sadlik at Amiwest Show 2010
http://blip.tv/file/4424195

Sassenrath at Amiwest Show 2010 - part 2
http://blip.tv/file/4370435

Steve Solie at Amiwest Show 2010
http://blip.tv/file/4358295

Sassenrath at Amiwest Show 2010 - part 1
http://blip.tv/file/4347633

Dickinson, Hermans, Luck at Amiwest Show 2010 banquet
http://blip.tv/file/4341466

Ben Hermans and the Friedens at Amiwest Show 2010
http://blip.tv/file/4326295

Trevor Dickinson, Ben Hermans at Amiwest Show 2010
http://blip.tv/file/4320600

A brief description of what each video is about is at each link.
Truly,
Robert Bernardo
Fresno Commodore User Group
http://videocam.net.au/fcug
The Other Group of Amigoids
http://www.calweb.com/~rabel1/
Southern California Commodore & Amiga Network
http://www.sccaners.org

C16/PLUS4 SPECIAL EDITION

A free brand new C16/Plus4 special edition of Your Commodore is now available for download.

It includes every games review they published for the 264 series. Every "Our Comment" monthly feature. All the Data Statement news, all the type-in games and a review of both the C16 and Plus/4 themselves.

As always, all the scans have been straightened, reformatted and cleaned up, no yellowing of the pages to give a nice fresh original magazine feel. A brand new cover design and a new index page which I (hastily) designed for this retrospective look back of yet another classic British magazine.

PDF Format (31MB)
http://www.commodore16.com/baz/magazines/your_commodore-c16-plus4_retro_special.pdf

You will need Adobe Acrobat Reader to view this, which you can download HERE http://www.adobe.com/downloads/

ZIP Format
http://www.commodore16.com/baz/magazines/your_commodore-c16-plus4_retro_special.zip

The magazine in numbered .jpg images. These are high quality scans. This comes with CDisplay which is slightly similar to the Adobe Acrobat viewer.

You can help the scene by purchasing the full magazine scans on DVD of this and many other classic computer gaming magazines at the Zzap Zzuperstore
http://www.zzap64.co.uk/zzuperstore.html

You can now download the programmer's edition (see Commodore Free THIS ISSUE news feature **Your Commodore - C16/Plus4 Programmers Edition** for details)

Incidentally, this took a year to produce. The reason why it took so long besides the magazine being larger than usual and due to other commitments was because I accidentally lost an important backup setting me back half the work. Oops.

Meanwhile, enjoy the memories, and please leave your comments below :)
Sixteen Plus

Your Commodore - C16/Plus4 Programmers Edition

A free brand new C16/Plus4 special programmers edition of Your Commodore is now available for download. The issue includes every programming article that was featured in this long running magazine.

As always, all the scans have been straightened, reformatted and cleaned up, no yellowing of the pages blah blah blah of yet another classic British magazine.

This is only available for download in high-quality JPG images, it comes with CDisplay which is similar to the adobe acrobat if you wish to view it this way. You will need to download all 4 parts first so to open with WinRAR to extract the folders which is available free for download HERE
http://www.rarlab.com/download.htm

Download:
Part 1
http://www.commodore16.com/baz/magazines/your_commodore-c16-plus4_programmers_edition.part1.rar

Part 2
http://www.commodore16.com/baz/magazines/your_commodore-c16-plus4_programmers_edition.part2.rar
Part 3
http://www.commodore16.com/baz/magazines/your_commodore-c16-plus4_programmers_edition.part3.rar
Part 4
http://www.commodore16.com/baz/magazines/your_commodore-c16-plus4_programmers_edition.part4.rar

You can help the scene by purchasing the full magazine scans on DVD of this and many other classic computer gaming magazines at the Zzap Zzuperstore

Incidentally, this took a year to produce. The reason why it took so long besides the magazine being larger than usual and due to other commitments, was because I accidentally lost an important backup setting me back half the work. Oops.

Meanwhile, enjoy the articles, and please leave your comments below :)

Sixteen Plus

AresOne 2011 is available now
Firmware update for the AresOne 2010

Recently released is an AresOne 2010 Firmware-update.

Just download the ROM-file and put it on a USB-Stick or burn it on a CDR. Then power on your AresOne 2010 and press „DEL" as soon the AresOne Logo appears. Navigate to EZFlash like described in the main board manual. During the first reboot you will be asked if you want to setup the Bios (F1) or JUST use default settings F2 - **PLEASE**! Press F2! Ok if you now what your are doing you can also Press F1 : D

The new Version supports more CPU's and fixes some bugs. get it here: www.aresone.de

Time goes by and we have now the third reincarnation of the AresOne.

The AresOne 2011
Well the description on www.vesalia.de says still 2010 but just compare the specs and the price. We have now 2GB DDR3 Ram. The HD4200 is now a HD3000 but as some may now both are interesting because Bearsoft is working on a driver for those... In the meantime you can get Nouveau support by buying the Geforce 8400. Also we are investigating the possibility to add a sparkle Geforce 250GT.... cross your fingers.

The price is now below €250 for the base system.

CBM-Command Version 2.0 Beta 1

This is the first beta release of CBM-Command version 2.0. CBM-Command has undergone a lot of changes since version 1.7, most of which consist of a thorough cleaning of the code by Greg King and new features added by Payton Byrd; with the space that Greg King freed up. Being the first public release after so many changes, please understand there will probably be some bugs.

New Features
* REL file copying between devices that support REL files.
* D71 disk image support - now you can create and write dual sided disk images with a 1571 drive
* Configurable Hot Keys - Don't like the default keys? You can redefine them yourself for version 2.0 using the Configuration program.

Changes
* Almost all code optimized for space savings to allow more features.
* Issuing a drive command is now done on the main screen instead of in a popup box
* When renaming a file, the directory remembers where you were so you don't have to scroll to the newly named file any longer.

* Changed disk image writes to reposition the buffer pointer in the drive instead of rolling the data in CBM-Command which freed up a buffer and some memory in the main program.
* Old keyboard shortcuts that required using the C= key now use the SHIFT key instead.

Known Issues
* Documentation is way out of date. New documentation is being written but is not yet ready for this release.
* Some of the default keys have changed; If you cannot find your old keyboard shortcut then go into the Configuration program and redefine the function as you need. These changes will be documented in the new version of the docs.

As always, you can download the latest version at http://cbmcommand.codeplex.com

---

RELEASED: NANAKO IN JAP. MONSTER CASTLE

Wood, the Portuguese developer who ported the Spectrum game UWOL to the C64 http://www.mojontwins.com/juegos_mojonos/uwol-quest-for-money-c64/ has released a conversion of an obscure Nanako game called "in classic Japanese monster castle." The game is released under the MojonTwin label, and a CPC and Spectrum 128+ versions are already available. …

The game consists of 25 levels you have to climb one level to another level in the castle and off course, there are some baddies. Nanako is a very original platform game with beautiful backdrop graphics and nice SID tunes.

http://noname.c64.org/csdb/release/download.php?id=120707

PLUS 4: 256K MEMORY EXPANSION

The Plus 4 hardware developer Levente Hársfalvi started his Plus 4 memory expansion project with the goal to use native development tools for his Plus 4 "activities". With the original 256K hack documents by Lux at hand, it became apparent that Levente had to piggy-back chips in the computer, which he didn't want to do anymore. …

After some pondering, brainstorming, etc. this is the product Levente came up with.

The Plus 4 memory expansion is designed in KiCad EDA, although it's Levente first attempt in about 15 years to etch a printed circuit board. As such, it should be considered a "prototype". The design is fully compatible to the original design of Hannes Lux & Christian Schäffner and has been tested successfully with Dream World (by Bionic/Synergy). http://plus4world.powweb.com/software/Dream_World

Benefits over the original design are:
*no piggy-back needed (installing a wire lead is still necessary).
*the need for custom chips eliminated (6529B)
*the memory bank register has a proper reset

*Levente is considering taking his design in to production, moreover the design is based on classic fundamentals and component are cheap and widely available.

For more information click here. http://www.facebook.com/album.php?aid=2092497&id=1458531880&l=3d34937604
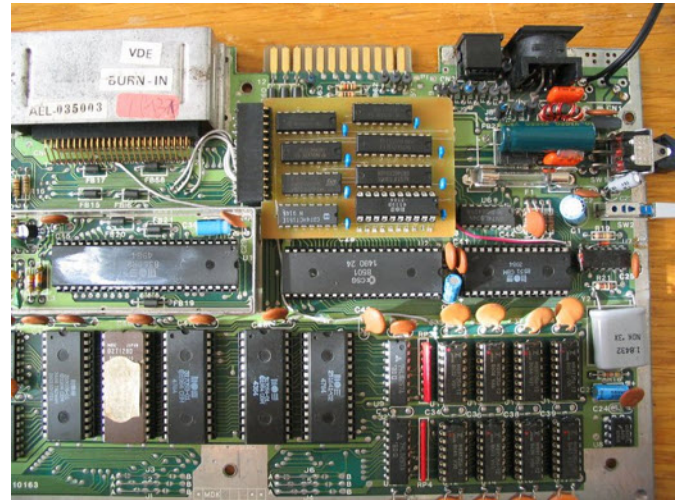
EWAY10 RELEASES 3 NEW C64 ADVENTURE GAMES

The German based C64 Publisher/Developer "Eway 10 Software" released 3 new adventure games, earlier announced in June. This time the games are translated into English. Nice touch, some of the translation is done by our own C64Scene.nl member: C64Addict (Hans).

The games are only released on tape, the games released are:
- Dark Catacombs (Dunkle Katakomben) is a classical RPG
- The Cave (Die Hohle)
- Astrolab

Eway10 is committed to develop new classical retro adventures for the C64 and keep the genre alive.
http://www.eway10.de/

VIC20 NEW GAME: THEATER OF WAR WW2 (beta)

Ghislain developer of the Vic20 game Realms of Quest has released the first beta of a new Vic20 game called Theatre of War WW2.

You play the role of a German general in the Eastern front that is outnumbered.

The computer AI great and the game simulates the USSR armies' dearth of quality officers because Stalin had purged most of the competent ones whose allegiance to pure doctrinaire communism was considered suspect.

Each side has a certain number of supply points that are generated per turn (which goes into reserve). This number goes down by 1 every time a unit is destroyed. Each battle begins with an empty map, and both sides start to place units (by using up supply/reserve points). A battle is won when a ground unit occupies the flag of the enemy.

The terrain also affects movement (can stop or slow movement) and can provide defence bonuses … the goal of the game is simple . "capture the flag."

http://www.c64scene.com/c64/wp-content/downloads/THEATR16.P00

Part 1: You can insert a copy-protected software disk, load the program, and when the point is reached where you are at the main level (or menu, or start), just press "S.M.A.R.T." and capture everything in the computer, reducing it to an instant load image.

[S.M.A.R.T. = System Memory And Register Trapping]

This does not "duplicate" the program to another disk; rather it images the operating program state to a distilled archive that will load again to that exact state in the blink of an eye. You may also adjust the S.M.A.R.T. capture features to allow precision captures of nearly all 64 programs. My personal archive contains more than 2000 captures and LLoyd Sponenburgh (LTK) has now created better than 6000 programs in his S.M.A.R.T. archives, and growing day by day. Soon, it is desired to have an entire C64 program biography (evolution) reduced to the S.M.A.R.T. instant load archiving format, all available for RA users.

(Disk Duplication (in most cases) can be done with FastCopy - Disk Duplicator Mode - F5 option from the FastCopy main menu. FastCopy automatically sends required 20x code to serial drives and unloads upon exit, returning any floppy drive to its normal state.)

Part 2: A CMD RAMLink is neither a DOS nor a Disk Drive; it is a RAM expansion device, limited to the hardware onboard. It does not provide anything more than extended Random Access Memory, with larger space. Even when paired with a parallel interface cable and a CMD drive, the performance is only "enhanced IEC serial" with emulation of CBM devices. This "emulation" falls short in many factors of true CBM compliance and causes many problems with programs. Other than a directory sorted storage lair, the CMD devices never got developed beyond rudimentary support. As for the Rear Admiral, all CBM support is fully available. With the RA, all of the work is done in the Host Adaptor, by swapping the entire programming and register space into (and out of) the Host Adaptor, before the computer knows what happened. The RA is, in a word, "Breathtaking" to see in action, since it would take your breath away upon seeing the speed it actually delivers without "SuperCPU" off-board assistance or other methods whereby the actual computer would be turned into a "dumb terminal" device. Unless you experience it, nothing more than a glorious myth would describe it. The RA DOS is the best part, exceeding the hardware, since it provides so many features that take the Commodore 64/128 into the true "programmer's realm" and beyond.

Rear Admiral performance is faster because it does all of the heavy-lifting for the computer within the Host Adaptor itself, allowing the computer to remain unhindered by required programming overheads placed on it when using RAM expansion devices (in any form). The Rear Admiral does not place any programming into the computer, the Host Adaptor does the work (as it should) without the computer ever noticing.

Evidence is in side-by-side benchmarking of the varying equipment. I still own several RamLink units as well as CMD Drives.

All with parallel cable configurations. These devices are still appropriate for many uses and should not be discounted.

Question - So it sounds like it just takes a snapshot like the Super Snapshot or Action Replay? That would be a nice feature, but basically doesn't have any better compatibility than the CMD HD or RamLink because you can't load copy protected programs. What about those with custom loaders? I guess those would not work either? Either way, I am glad to see something new being produced in 2010. It would be nice to see a video of it in action!

It does in deed support copy-protected software. S.M.A.R.T. is unlike Super Snapshot, Action Replay, or anything else for that matter.

Here is a more detailed (yet condensed) overview of the technology:

[S.M.A.R.T. = System Memory And Register Trapping]

What does the little blue button really do?

1) The system is prepared by allocating a centralized capture repository file and then placing the necessary SMART code into the Host Adaptor's on-board "shadow" RAM. The computer is then rebooted to a normal (non-RA) state. The Host Adaptor is now aware (transparently) and it waits for the S.M.A.R.T. button to be pressed. Now the user can load the target program (even copy-protected) from the original source (floppy) drive. At some point (user's discretion) when the program is loaded or running, the user presses the S.M.A.R.T. button to initiate the actual capture procedure. Once the button is pressed, the Host Adaptor "freezes" the entire computer - CPU activity, I/O, memory and all register states (as well as clock data), etc. At the same instant, all of the "frozen" information (system-wide) is buffered into the capture repository - a large REL "keyed access database" file with an indexed map of all captured information from the system. The system is then booted back to normal RA operation. The "image" file is available for instant access (via the SMART menu) and can be reverse applied (loaded back) into the system in about the same time it takes for the "ENTER" key to come back up from being pressed.

2) Since the system conditions change in a unique way for each individual program that is captured, the single repository is not the logical end of the capture process. The repository file is mapped large enough to handle the entire scope of the 64 yet, in most cases, very little of that entire scope is really changed. So, the SMART menu is the next step after capture. Adjustment "tweaks" (i.e.; user preferences) may be applied to the dynamic "playback" settings for the repository file. Once satisfied with the settings (defaults usually do the trick), the user may then create a permanent "archive" of the repository. This process will compare the "normal boot" state of the computer with the "target state" after capture. From this comparison, a vastly reduced map of the system necessary to load the captured program can be derived. That information is then placed in a new archival version of the repository (user named) and becomes the final capture image at a vastly re-

duced file size. The central repository file will now be used again for the next capture, in the same way it was previously, without needing to re-allocate. The SMART menu will also build autoloader front ends for any archive file. This makes it easier to load the archive by simply typing the name of the archive, instead of having to load it from the SMART menu each time.

3) The entire capture process requires approx. 1 second to perform. The "reload" process requires an approx. maximum of 350 milliseconds seconds.

A video? Not this month. Too many units to build.

Question - Thank you for the detailed explanation. I look forward to seeing one in action sometime. Maybe I'll see you at a Commodore conference sometime?

You're welcome, and correspondingly thanked for your questions.

Conference? That may be possible. I am currently involved in assisting a local (Detroit/Metro) area club (64/128/PET) in their revitalization efforts. They recently merged with another club (Amiga/Atari), causing change to their meeting place and schedules. I've had to wait for some of the dust to settle before jumping in. My goal is to provide a below-cost Rear Admiral System for their general equipment, which already includes some LTKs. However, I will need to recover a portion of my investments from the project before I can proceed.

You may be interested to know that I have somewhat more immediate plans to post additional information on the forum, and allow members to purchase Rear Admirals directly. This would (1) save money for member purchases and (2) generate funds for the forum, as I will be setting aside $10 from each unit sold (through a sponsored link) here for Lemon64.com.

Question - [Is it just copying the Lt. Kernal?]

I suppose it would be a nice job of "copying" had it been just a clone. However, the need for "just a clone" was nothing, as a goal. That lesson was learned (the hard way) many years ago after successfully "cloning" the CMD HD's interface logic. Although rather simplistic in design, it was still a laborious task to duplicate that device - the "cloning" of which has not been achieved elsewhere to this day. Upon first conception of the RA project, it seemed as if to be looking down the throat of a behemoth. The logic was so complex that it was amazing how they could fit so much into the 5C032 PLDs from that era. The project also required unorthodox methods (to be invented) in reverse engineering, just to get started. The CMD logic uses basic combinatorial methods, whereas the LTK uses both combinatorial and sequential logic, spanning two PLDs, with portions of the computer and other Host Adaptor components in between. After reverse engineering was finally completed, then came the task of improving upon the logic with "notions" dating back to the late 80's of my LTK experience. Eventually, the next version of the LTK emerged as the Rear Admiral HyperDrive. Now the floodgates are open for the batch of "surprises" to be discovered in the next release of RA DOS (7.4), that of which all registered owners will receive free of charge.

Complete owner's manuals are available to all registered RA users and accompanying documentation is sent with the product.

High-Quality detailed photography transmittal is currently suspended until an online-magazine finalizes a review of submitted information. This material also includes historical/retrospective data pertaining to the original LTK, as well as associated photography. There are delays pending this review due to minor language translations for international publication

---

UIEC CHENGE DEVICE PROGRAM

TO: Commodore Groups
FROM   JIM PETERS
Subject: uIEC device Number change

Old Jim here:
I added a couple of input lines, to the program someone posted some time back. I think it makes changing device number on the uIEC a little bit easier.   The keys listed inside : { } are to be used with the SHIFT key and [ ] are to be used with the CONTRL key.

```
 10 REM – INPUT DEVICE NUMBERS –
100 PRINT "{CLR/HOME}"
110 INPUT "OLD DEVICE NUMBER";OD
120 INPUT "NEW DEVICE NUMBER";ND
200 REM – CHANGE DEVICE NUMBER –
210 OPEN 15,OD,15
220 PRINT# 15,"U0>"+CHR$(ND)
230 CLOSE 15
300 REM – LOCK NEW NUMBER INTO EPROM
OR NOT ? –
310 INPUT "LOCK NEW NUMBER INTO EPROM
[REV]  Y/N  [OFF]";A$
```

```
320 IF A$ = "NO" THEN END
400 REM – LOCK NEW uIEC DEVICE NUMBER
INTO EPROM –
410 OPEN 15,ND,15
420 PRINT# 15,"XW"
430 CLOSE 15
```

Hope this makes changing your device number easier.
Jim Peters
Very nice. One fix I saw in there:
320 IF A$ = "N" THEN END
if Y/N was the input the program would never see NO... Also I would suggest:
320 IF A$ <> "Y" THEN END
that way if they miss-type N it does not change the eprom
Larry

ZTERM TERMINAL CLIENT FOR C64

Jeff Brown wrote a ZTerm C64 Terminal client as a CCGMS clone, in essence to support the swiftlink/turbo232/link232 high speed cartridges. You can download the application from here

http://www.c64scene.com/c64/wp-content/downloads/zterm.d64

Chameleon 64
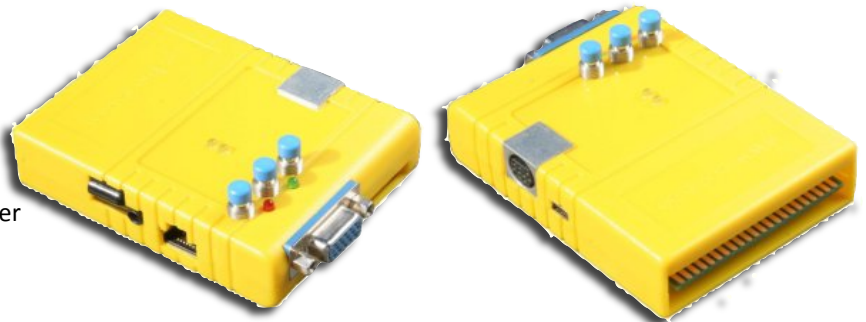### Flicker fixer, REU, Turbo Board and more for the Commodore 64

Chameleon 64 is an extremely user-friendly cartridge that can be used without opening the computer. It is just plugged into the expansion port of the C64. Everything a C64 user would expect from a multi-function cartridge is there; plus a lot more, as you can see in the features overview.

The Chameleon cartridge can also be operated as a stand-alone unit (since it's basically an FPGA computer like the C-One or the Minimig in a C64 cartridge case), replacing the computer, the floppy drive and the heavy power supply. When used as a stand-alone unit, a USB power adapter (available separately) or active USB hub can be used as a power source. Even a USB-enabled computer (or LCD TV, DVD players) can be used to power this nifty little cartridge.

Technical data in short:
- VGA-output with 60 Hz refresh rate or more
- Turbo function with full 6510 processor compatibility (incl. illegal opcodes)
- Cycle-exact REU (memory expansion) with 16 MB RAM
- MMC/SD card slot with MMC64 compatibility and 1541-emulation
- Connector for PS/2 mouse and PS/2 keyboard
- Freezer (compatible with Retro Replay)
- Bright yellow case with blue buttons
- Battery-backed real time clock
- IR receiver for CDTV remote control
- Stereo audio out for stereo SID emulation
- USB link to a PC for debugging and data transfer
- Connector for RR-Net (10 Mbit Ethernet)

(image shows prototype version)

The Chameleon 64 cartridge is under constant development. Since it is a re-configurable platform, hardware updates can be installed by updating the flash of the cartridge. This does not require expert knowledge, and it doesn't even require opening the cartridge. All you need is an internet-enabled PC (running Linux or Windows) with a USB port; this lets you perform the update via a USB connection. The special breakout-cable required for this connection is included with Chameleon 64

Orders are being taken from VESALIA online
http://www.vesalia.de/?V02b0f13545e4716715e59555c14475
b47071049485650100218 0b480b457212087f766c34323e756
e0c0a2c657f2e39291b39621d13383a27307f7b226a6a696545
1174647344655f41147a6f0b655c46562a7f411e1f7c46196b42
1a61471d55666

The unit is priced without Shipping costs at €199.

Individual's website is listed here
http://www.jschoenfeld.com/

the product information is available from here
http://www.jschoenfeld.com/products/chameleon_e.htm

or here in German language
http://www.jschoenfeld.com/products/chameleon.htm

sd2iec Firmware - Geos Bounty completed!

Commodore Bounty is proud to announce that the very first bounty has been completed!

The sd2iec-Geos support Bounty is complete and bounty contributions have been released. sd2iec-0.10.0 is confirmed to support all (non-cracked versions) Geos 2.0 64 and 128 D64, D81 and DNP images. Additionally (non-cracked versions) Wheels 64 and 128 v4.1-4.4 is confirmed working with D64, D81 and DNP images. These have been tested without SuperCPU and With SuperCPU turbo enabled.

sd2iec-0.10.0 is available to download as always from HERE.
http://www.sd2iec.de/

Commodore Bounty would like to thank Ingo Korb for such a fantastic job done. The results are everything envisioned and more!

http://commodorebounty.com/index.php?option=com_kunen a&Itemid=54&func=view&catid=8&id=143#145

A quick and dirty video demonstrating Wheels 128 v4.4 (aka Geos 4) running directly from a uIEC/CF (compact flash reader), on my Commodore 128 Tower with SuperCPU 128. In the future I will do other videos focusing more on specific uses and specific software. This video is simply proof of Wheels running from he uIEC. Geos and Wheels support to the sd2iec firmware is thanks to a completed bounty of Commodore Bounty and the efforts of programmer Ingo Korb.

http://www.youtube.com/watch?v=mPdOKuZcrNc

On the Edge: the Spectacular Rise and Fall of Commodore

To: Commodore Free
Subject: Commodore: A Company on the Edge - near release

It's been a long time in coming, but Brian Bagnall, author of the book, "On the Edge: the Spectacular Rise and Fall of Commodore", has confirmed that his new book, "Commodore: A Company on the Edge", is at the printer right now and will be shipped soon. In fact, at Amazon.com there is page that says that Variant Press will release the book on December 15. Just in time for Christmas!
To see the initial announcement, go to

http://www.amazon.com/Commodore-Company-Edge-Brian-Bagnall/dp/0973864966/ref=sr_ob_1?ie=UTF8&qid=1289706472&sr=8-1

   Think of the new book as a second edition of On the Edge. Here is what Brian had to say, "It was a long road and a lot of work to get the second edition into shape. There was so much new information from the 15 additional interviews and tons of pictures that I had to split the book into two parts. A Company on the Edge deals with the period up until Tramiel leaves and the second part (slated for 2012) will tell the post-Tramiel years with the Amiga. Each book deals with about a 10 year period, and a lot of players change in 1984 so it worked out to be a pretty good split."

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

----------------Original message----------------
From: Brian Bagnall
Date: 29-Nov-2010

Hi everyone.

I'm happy to report that my new book, Commodore: a Company on the Edge, is now available from www.VariantPress.com. This is a heavily reworked edition with about 40% new material. Before starting this project I made a list of everything that I would have liked to include in the first edition. The main areas I wanted to work on were: more interviews to fill in gaps in the story, reorder the material chronologically, and add more period photographs. It was very important to me to put all this material in the right order so readers could read a coherent, chronological retelling of Commodore history. This time you can relive the Commodore experience in real-time.

Originally, the goal was to keep the same page count as the first edition. That turned out to be overly optimistic. After 15 new interviews and a lot of new stories, plus numerous pictures, there was no way to make it all fit. As a result, the story has been split in two. The first book ends in 1984 when Jack Tramiel is forced out of the company and the next book will focus on the Amiga years and Commodore's eventual demise.

There are many new firsthand interviews:

Brian Dougherty – GEOS developer
John Feagans – PET developer
Andy Finkel – Commodore game developer
Bill Gardei – C65 systems engineer
Neil Harris – Commodore game developer and marketing Manfred Kapp – Commodore cofounder Dale Luck – Amiga developer Bill Mensch – 6502 and 65816 designer Dick Sanford – Chief Financial Officer of Commodore Bill Seiler – PET, VIC-20 systems engineer/surfer Kit Spencer – Commodore's marketing guru Yash Terakura – Commodore Japan engineer Michael Tomczyk – VIC product manager and marketing Plus additional fact-checking interviews with Chuck Peddle, Bil Herd and Bob Yannes. I did get a chance to talk with Jack Tramiel briefly but didn't get any new material from him unfortunately.

I don't want to ruin too many surprises, but I will mention a few things. First, thanks to John Feagans, an avid photographer, the book is populated with outstanding period photographs. One of the biggest surprises for me was finding out that current Nintendo CEO Satoru Iwata started his career programming VIC-20 games for Commodore Japan. The book digs deeper into actor William Shatner's association with Commodore and reveals some interesting titbits. There is also a firsthand account of the horrifying PET Jet disaster. Lots of new changes like that.

The book will be available in stores in about a month. If you need it for Christmas you can get it immediately from my publisher, Variant Press, which distributes signed copies from its website.
http://www.variantpress.com
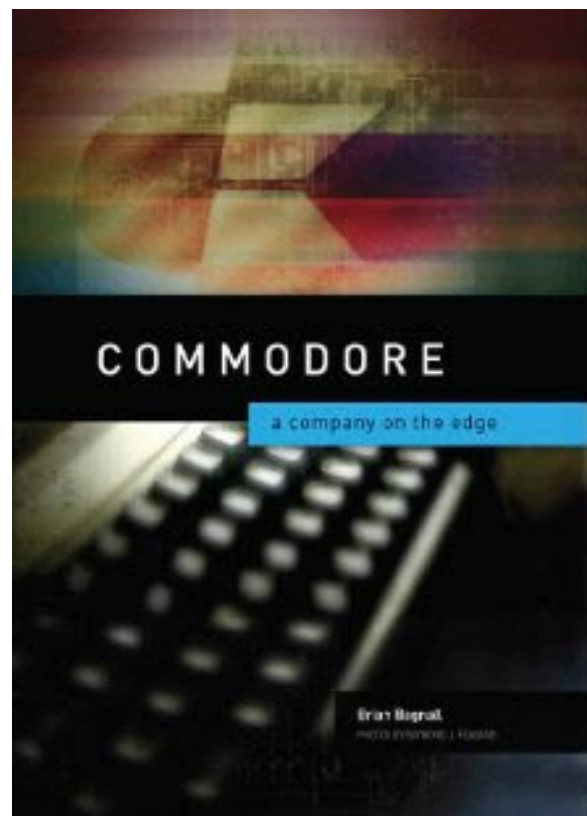
DiscreetFX Platform Shift

DiscreetFX's primary focus for quite a while has been developing visual effects products for video editing solutions on Windows. We would sometimes also make these effects available at a later date for Mac OS X. Potential Mac customers would call us concerned because certain products would ship much later for Mac OS X if ever. As 2010 winds down and ends we have noticed a softening in our Windows effects business. We did some research and found that while companies that sell integrated hardware/software solutions on Windows are doing fine software only vendors have noticed a great slow down.

Our research results have come back and this downward spiral will only continue and accelerate in the future. Microsoft continues to bleed many key executives at an alarming pace. What do they know that we don't? Since we wish to grow and expand DiscreetFX we plan to avoid this destitution and software sales decay. For some time we have already run the business side of DFX on Mac OS X and this has worked out well. Starting today we will change the focus of our software effects development from Windows to Mac OS X. Now the OS X versions of our software will come out first and later the Windows versions will ship if ever depending on the product. The first product this will affect is MusicFX.

DiscreetFX's Amiga business and Amiga.org will remain unchanged since this has always been the fun side of the business. This decision should strengthen DiscreetFX and help it grow. We are only reporting it here for transparency reasons and for full disclosure to our customers since some of them are members of this site. Of course old products are still fully supported.

If you have questions don't hesitate to ask. Our first mission is to help our great customer base. Exciting times are ahead like the Mac App Store. A lot of our regular customers have changed platforms from Windows to Mac and we will now join them. Expect to see other software developers follow suit. We are usually slightly ahead of the curb. If you remember we were one of the first to embrace selling software on Flash drives in 2006 and we completely moved to selling all our FX software on Flash drives in 2008. Apple just started selling Mac OS X and iLife 11 on Flash drives last week with the new MacBook Air.

Best regards
DiscreetFX Team

Visual FX company DiscreetFX LLC Inc. and a group of investors have submitted a proposal to Amiga Inc. requesting formal validation of the company's value. DiscreetFX hopes to secure funding to purchase Amiga Inc and it's Intellectual Property within the next year.

Currently DiscreetFX is hard at work completing their documentary titled "Oil Change". Oil Change is a film about the dangers of trans fatty acids in the style of "Super-Size Me". Plans to purchase Amiga Inc would proceed after the completion of the movie which is currently in pre-production.

Ultimately this transaction would depend on Amiga Inc's willingness to sell and price negations. DiscreetFX believes there is tremendous value in the Amiga brand and the Amiga Operating system.

DiscreetFX CEO stated. "Today we primarily develop our visual effects software on NewTek video editing solutions for Windows and Final Cut Studio for Mac OS X. But the Amiga has always held a special place in my heart since I developed my first product for the Amiga Video Toaster on it. I want to help bring the Amiga back to it's former glory in new IT markets."

If an agreement can be reached between Amiga Inc and DiscreetFX, the much speculated return of the Amiga brand could spark a unique, new chapter in the machine's ever fluctuating history.

Best regards,
DiscreetFX Partners

About DiscreetFX

DiscreetFX has been creating software products for the Amiga, video editing, computer generated graphics (CGI) industry since 1995. The Amiga computer defined and created the affordable video editing, computer graphics market with its birth in 1985. DiscreetFX creates Real-time transitions and effects seen on over 100 television programs including Blind Date, 5th Wheel, Shipmates and more! You can also see DiscreetFX software used on the following networks HBO, Showtime, Discovery Channel, PBS, Fox and more! Video Toaster, VT[5], VT[4]/[3]/[2], SpeedEDIT are registered trademarks of NewTek. Final Cut Studio and Mac OS X are registered trademarks of Apple Computer. Amiga is a registered trademark of Amiga Incorporated

http://www.discreetfx.com/News.html
http://www.discreetfx.com/

INTERVIEW WITH Alf Yngve

http://www.youtube.com/user/SEUCKgames

HISTORY OF SEUCK

Some people hate games based nothing more on the fact that it was written with SEUCK (the shoot em up construction kit) the program is a games creation package notably and fairly obviously from the title you will have guessed, it was designed to create shoot-em up style games. The package was created by Sensible software who wrote such notable titles as "parallax" and "Wizball" when the system first appeared it was said that Sensible had "given away there trade secrets" to create a games This was a system that required no programming knowledge to use, but still unfortunately needed an initial idea and some basic artistic talents. Some still doubt that it can produce anything at all never mind a decent game, Given the right idea and the skill and patience to sit down with a good design the package is more than a capable system, true it has limitations like it would only scroll vertically, although recent

updates have seen a horizontal version being released, if you want to find out more and download some games created in the package you could do no worse than to point your web browser here http://www.seuck.retrogaming64.com/index.html a very active site dedicated to all things SEUCK Not all SEUCK games are actually space games as you will find out.

Q. Please introduce yourself to our readers

Hi Commodore Free Readers!
I'm a Swedish game designer; I think I am mostly known for making SEUCK games, and I have done this since the 1990s.

Q, What profession do you currently work in

Well sadly I'm no longer in the games industry.

Q. How did you first come across the Commodore brand of machines

I Bought a used CBM64 from a friend.

Q. You have written a number of games using the SEUCK system can you explain the system to our readers

The Shoot'em-up Construction Kit is a very simple-to-use, fairly basic utility/interface for making games without programming skills. Everything is self contained so you have a sprite editor music and sound creator and of course a logic glue to put the all together. You can at a very basic level design characters and place them graphically on screen then set routes for them to follow and how to fire and when the screen moves etc. the original version only moved the screen upwards The best way to learn the utility is by trying it out.

Q. Why use SEUCK

Basically to save time, because you don't need to know how to program and also because you get fast results you can see any changes immediately on screen. If this sound like your way of working then SEUCK is the thing.

Q. Recently SEUCK has received a number of updates can you explain what the original system and its limitations and how the updates have been implemented

Now there are THREE versions of SEUCK available:
1. The classic version from Sensible Software, for making only vertically-scrolling games
2. Jon Wells' new version with horizontal scrolling instead of vertical
3. Gernot Frisch's SEUCK freeware program for making PC games with 16-bit graphics.

Q. Are there still limitations in the system you find frustrating

1. No power-ups.
2. Memory limits (but you can work around them if you're able to code a "multiload" system in addition to the original software.)
3. No music editor/option.

Q. Your recent game 1,000 KUNG-FU MANIACS! Has been enhanced though can you explain the enhanced process

You'd have to ask Richard Bayliss, the coder; I have no programming skills.

Q**. I know Richard Bayliss is a big SEUCK fan, did he provide any pointers to creating a great game**

Yes! He figured out what was possible with the system and what was not, and we collaborated closely on testing the power-up design.

Q 1,000 KUNG-FU MANIACS! is to be released commercially can you tell our reader where he would buy this product

Of course you can buy the finished version from http://www.psytronik.net

Q. Will the final version be a boxed version, and will there be other formats available for example on tape disk and as a digital download

From the website you can obtain the following versions
A Boxed version,

A tape version ,
A disk
and there is even a digital download in the form of a D64 file.

Q. So what is the process in designing a SUECK game, you start with an idea then what...

Well I Always start with an idea,
Then I work on a theme,
Of course you could do a puzzle or a simple story.

Q. When did you start using SEUCK

Back in 1987 or 1988. Only after a few years did I start to send them to magazines. One editor at Zzap!64 (later renamed COMMODORE FORCE) encouraged me to seek a job in the games industry
which I did! And I worked in game design and QA (games testing) for many years.

Q. Did you receive any feedback about the games

Actually, no not that much to be honest.

Q. Do you have any favorite SEUCK games

I'm very pleased with how my game NUKENIN & THE RONIN turned out of course I am biased towards them.

Q. Is there a magic formula for creating a perfect SEUCK game

Sadly there are no perfect SEUCK games and no magic formula. But you have to test, improve and polish the game until it gets good. Attention to detail is crucial: one badly implemented feature can spoil the entire impression.

Q. Do you have any other plans for SEUCK games

More plans than I have time (unfortunately).
At some point in the future I will have to shift from C64 SEUCK to PC SEUCK development.

Q. Do you have a final comment for our readers

SEUCK is the computer game equivalent of Punk Rock: Do it yourself, it doesn't matter if you can only play three chords, and don't be afraid to express yourself.

COMMODORE FREE REVIEW

1000 kung-Fu maniacs!

This is a review of the Download disk image so I can't comment on the packaging and production.

Upon Starting the disk we have 5 options to select from the main menu

1. 1000 kung-Fu maniacs – the game
2. 1000 kung-Fu maniacs – the ending
3. stone age fighter
4. Ninja Blueberries – part 1
5. ninja blueberries – part 2

The music is very suitable for the main title screen very Oriental with gongs and oriental chimes. The in game music is written by Jon Wells.

Selecting option 1 then
We are loaded to the next menu screen again with suitable oriental style warbling music and lots of white noise (how can I describe a piece of oriental style music) well its oriental

themed

You have 3 options
F1 setup the music or SFX

F3 The story and instructions
F5 Lets fight

The story goes like this
Our here the master of awesome saw a child crying it seems that the Childs toy was stolen by the Blapsville boys, the child asks our hero to get the toy back

We then see our hero walking and skateboarding over various road layouts till he eventually comes to a grassed area with manic golfers firing golf balls at our hero.

It's quite a plain landscape, but this is more to do with the limitations of SEUCK rather than anything else. Our Hero is well animated although when I see a banana on the floor with the words TAKE PEEL, I do seem to earn points and loose energy and can't seem to bypass the banana without taking it. I am then given a skateboard to fight off Skater style Ninja enemies, not sure why Ninjas are on skateboards but hey it's a faster travel mode than walking. We pass various places along the way like Blapsville country club and every time you are hit, suitably fall on the floor; At the start of the game you have 6 Dan style shields for energy when these are depleted our hero's life ends. I am afraid I don't seem to be very good at this game and die rather quickly.

It may look basic; as most SEUCK games do, although some of the in game humour will keep you going back for more. Surprisingly it is very playable and does have the

"one more time factor", if you are a SEUCK games style fan this will definitely be one for your collection. If you have SEUCK then I am afraid the game will not convert you.

Who could load up a game with an option on a menu to see the ending without selecting this option? Ah well for this you need a 4 digit code and rather sadly I don't have to see the ending.

STONEAGE FIGHTER
Another game on the disk
This game you have to push boulders onto your enemies to kill them, again this is another SEUCK constructed game and is split into 2 parts, I presume the real skill in playing the game here is the learn all the movement patterns of the enemy, I suppose this is true of all SEUCK games. Sadly I am rubbish at this sort of thing and hence my efforts in the game were left rather lacking again. Not a bad game really it again has the "one more go feeling" and is a very well executed SEUCK game. Again the author is working within the limitations of SEUCK so you have to take this into consideration. Part 2 states you have to fight your way to the monster cave where a woman is held. Unfortunately I never made it to the cave. Quite a lot going on in the first part graphically and the sound effects are functional enough; you can only throw stone up the screen, which is driving me mad! All ads to the one more go though.

The title screen music is good however I am not sure how the tune fits in with the Caveman theme of the game.

NINJA BLUEBERRIES PART1
Nice intro music fairly Ninja like in sound, well it sounds oriental at least
Very well executed game, This could have been sold on its own

The in game sounds are just that again no music or extra enhancements and the usual 5 shields and you're dead. This looks more like a "traditional" SEUCK game you're moving up a scrolling screen shooting and or avoiding being shot, as with all these games it's very addictive; learning the pre-set moves will help enormously in this game. Very nicely animated sprites and explosions. The games copyrighted 1992 and was released as Public Domain, personally it's the first time I have seen the game, again I

don't seem to be that good, but that seems to be a theme with me and SEUCK games.

NINJA BLUEBERRIES PART1
How can we have part one without a part 2
Here the sequel to the Ninja game again copyrighted 1992, nice Title music although it seems to have lost the oriental twist a little. This games is set in a more perspective environment, although again is a traditional SEUK scrolling game Very well executed SEUCK game with lots happening on screen, here we see some of the familiar SEUCK glitches, although this is the software package rather than a fault in the game, I suppose less action would have helped, but the game is superb. I am actually better at this game but still can make it past stage 1 again the game was listed as PD

There you have it then
You seem to get quite a lot for your money, especially if you download the Free D64 image; if you're a SEUCK fan I suppose you will already have the PD versions but to have them as a packaged item.

The game is freely downloadable now from Psytronik Website

http://www.psytronik.net/main/index.php?option=com_content&view=article&id=88:kungfu&catid=34:commodore-64&Itemid=57

http://www.cmdweb.de/

Here you find information about the **SuperCPU**, **FD disk drive**, **HD series**, **JiffyDOS** and
RAMLink. To order, pick your closest **distributor**.
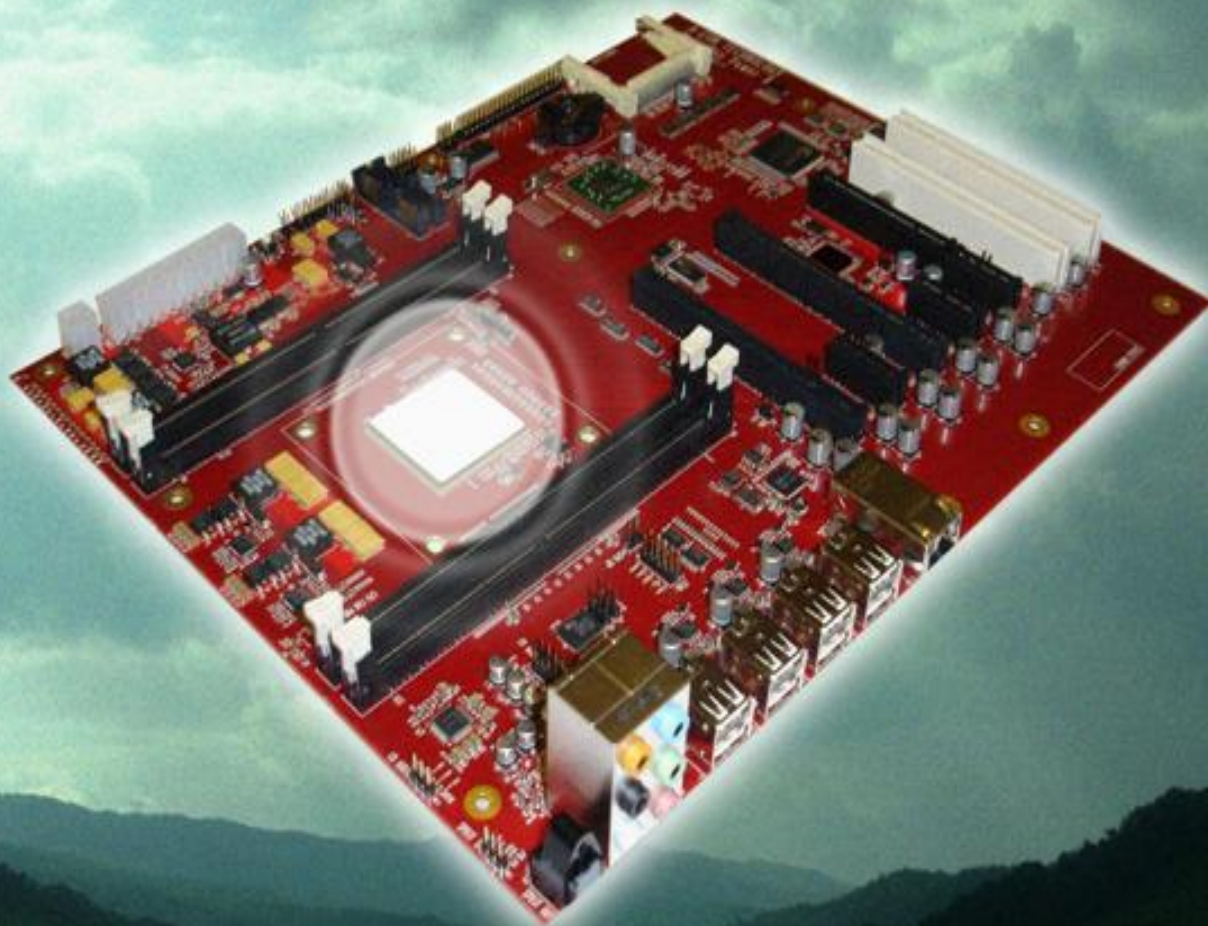Furthermore, you can visit our **About CMD** section or the **Download area**.

AmigaONE

X1000
I BELIEVE

How to build your own RGBI>S-Video Adaptor

By Allan Bairstow
www.commodorescene.org.uk

Foreword :

Many of you will be aware of my task to find a decent adaptor to convert the C128 80 full colour signal into one that will display on a modern SVGA LCD monitor, this has not happened yet but I think I am getting close. However, earlier this year, I finally managed to get part way there, with this S-Video adaptor which will allow you to use a modern LCD panel providing it has an S-Video input. The quality is very good, although not 100% correct, but to the untrained eye it looks perfectly fine.

I have now built many of these units for people and the feedback has been mixed with gratitude and a sense of relief that they no longer need their bulky, old and failing monitors. There are a couple of downsides to building these adaptors
(1) they are very fiddly to make,
(2) some of the parts have to be bought in bulk as the suppliers usually have a 'minimum order' policy,
(3) they are very time consuming to make.

Even if I built it correctly the first time it still takes me nearly four hours to build and test each one. Generally speaking, that is one a day.

So here we have a building guide for you to follow if you wish to attempt it yourself. I must warn you that you will need good soldering skill, a steady hand, good eyesight and plenty of patience.

All prices quoted are approximated from bulk buying and all the details were correct at time of writing.

Parts list : (Fig 1)
1x Case - Ice Blue 1591BT (Maplins N06GC) £4
1x A22 Board (eBay shop Y-PLUS usually have them) £30
1x Half a 9 pin serial cable (Maplins L66BT) £2
4x rubber feet (Maplins FW38) £1
4x circuit board stand offs (Maplins A01BJ) 32p
1x Grommet (Maplins) 10p
1x Breadboard section (Maplins) 50p
1x Small cable tie (not shown) 10p
6x Diodes 1N914 (eBay) 30p
3x Resistors 1k Ohm (eBay) 75p
1x Resistor 1.2K Ohm (eBay) 25p
1x 6pin connector housing (Farnells 151-6269 pack of 5)*
6x pins for connector housing (Farnells 151-6301 pack of 10)*
*I cant remember what these cost but there was a minimum order of £5
As you can see, before we even begin building the cost of parts alone is nearly £45 !

FIG1

Essential Tools :
Small vice (I used a 'quick grip')
Modeling drill with a cutting disc
Circuit tester / Multi Meter
Soldering Iron
De-solder device
High grade solder
Stepped core drill bit 4/6/8/10/12mm (see Fig 18)
Selection of small screwdrivers and craft knives
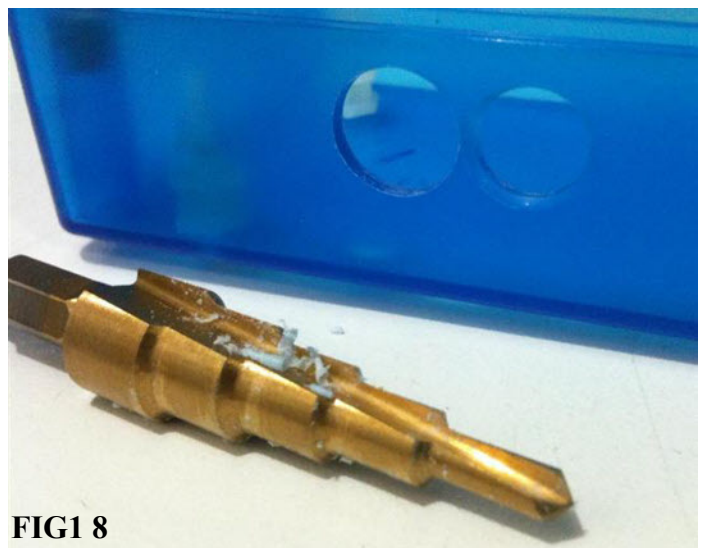
FIG1 8

Stage 1 – Modifying the A22 board
The A22 board has a 'composite out' connector
(Fig 3)

FIG3

that we don't need as it gets in the way when we try to house the board in its casing so it has to be removed. De-solder the two pins holding the composite socket to the board while carefully prizing the socket away from the board.
The removed socket (Fig 4)
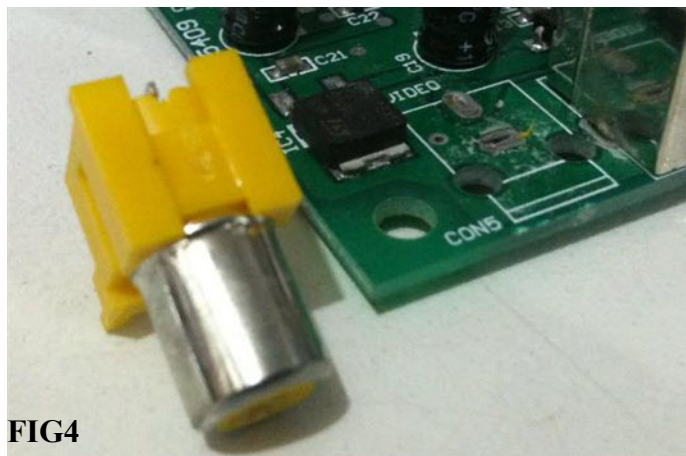
FIG4

can be discarded as it is no longer needed. History fact : early designs incorporated the Composite output but the quality of the signal was horrendous to say the least. Take it from me, you don't need it ! It also makes inserting/removing the A22 board a lot easier when you are building your device.

Stage 2 – Modifying the Case
Inside the casing are four raised nodules
(see Fig 5)
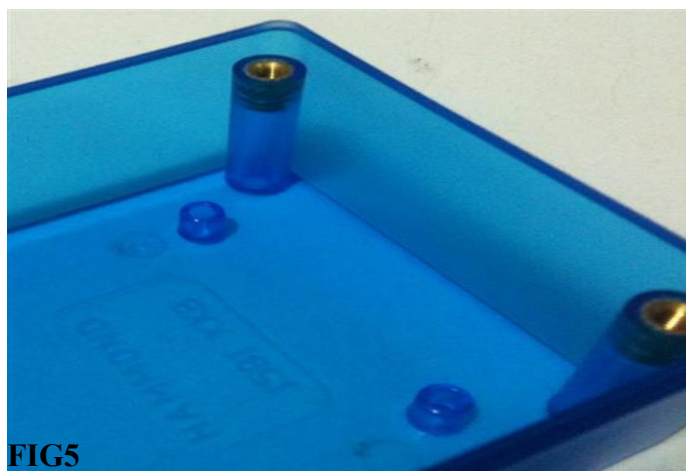
FIG5

which will need to be removed as they will foul the board if left in place. Using the modeling drill with a cutting disc, cut away the four nodules and trim any plastic away to leave the base nice and smooth
(Fig 6).
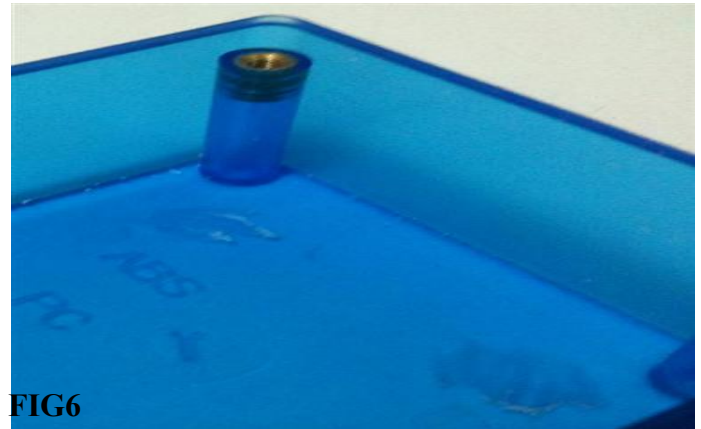
FIG6

Place the A22 board centrally inside the case with the socket side flush up against the case wall
(Fig 7)

FIG7

and mark the four board holes with a marker pen
(Figs 8, 9 & 10).

FIG8

FIG9

FIG10

Drill out the four marked holes with a small drill bit that is 'just' smaller than the threads on the board stand offs (Fig 11).

FIG11

do not cut them off. Refit the A22 board and push it home onto all four stand offs (Fig 14).

FIG14

Now screw in all four stand offs (Fig 12)

FIG12

We now need to mark three holes to cut, the main two at the front for the s-video socket and power socket (Fig 16)
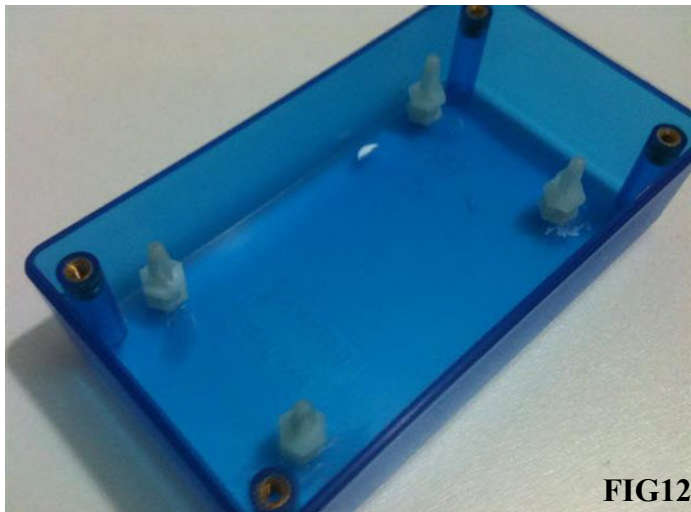
FIG16

and leave the protruding threads on the underside of the case (Fig 13)

FIG13

and one to the left hand side of the case for the RGBI cable to go through (Fig 17).
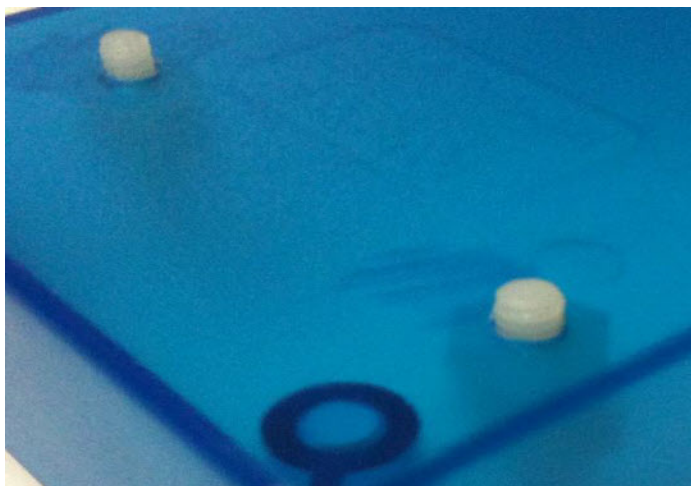
FIG17

VERY CAREFULLY – remove the A22 board.
Using the stepped core drill bit, drill the holes as follows: 12mm for the s-video, 10mm for the power socket
(Fig 18)

FIG18

and 8mm for the RGBI cable. Now cut out a 'V' section above the 8mm RGBI cable hole
(Fig 19)

FIG19

so that the grommet will slide into place
(Fig 20).

FIG20

Replace the A22 board and check the hole alignments
(Fig 21)

FIG21

and alter as required. Turn the case over and place the four rubber feet in position
(Fig 22).

FIG22

Stage 3 – Making the daughterboard
The daughterboard converts the RGBI signal into a standard RGB signal so we now need to turn this collection of parts
(Fig 23)
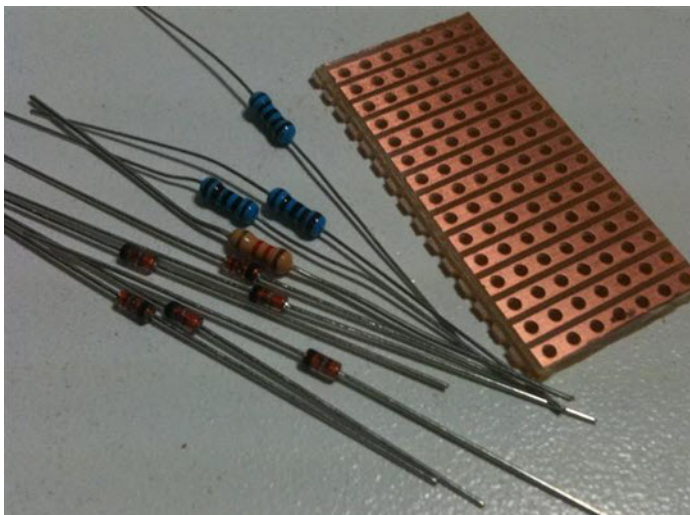
FIG23

into a daughterboard. This part is very important and is the second hardest part to achieve! I won't bore you with how to solder (you shouldn't be doing this if you can't solder), simply follow the wiring diagram
(Circuit 1).
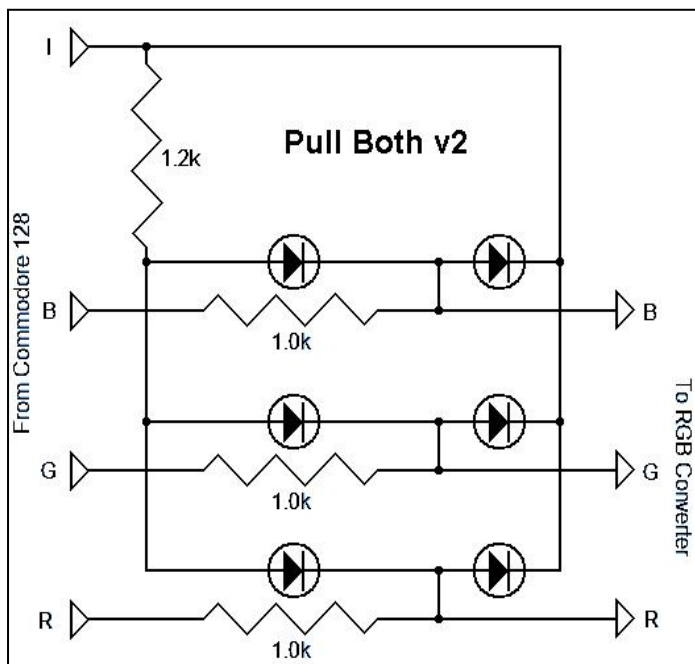
Circuit 1

The finished board should look something like this
(Fig 24).
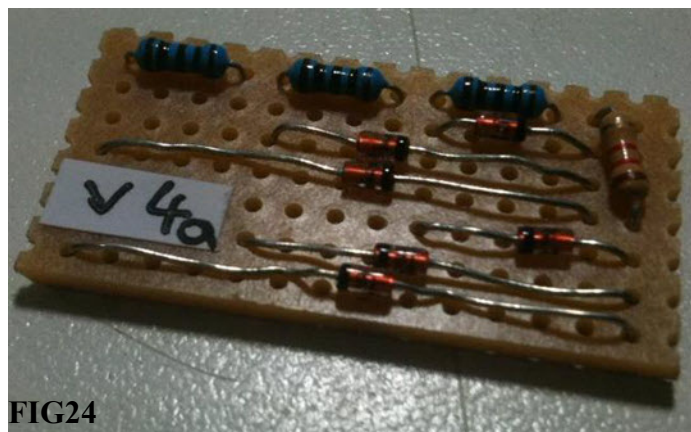
FIG24

Technical note :
Resistors can be fitted either way around but Diodes can't – make sure they are the correct way round before soldering them into place.

Stage 4 – Adapting the serial cable
Firstly, remove the grommet from the case and slide it down the cable and leave it there (Fig 29).

FIG29

Next you will need to remove the outer sheathing from the wires, make sure you expose about 6"
(Fig 30).

FIG30
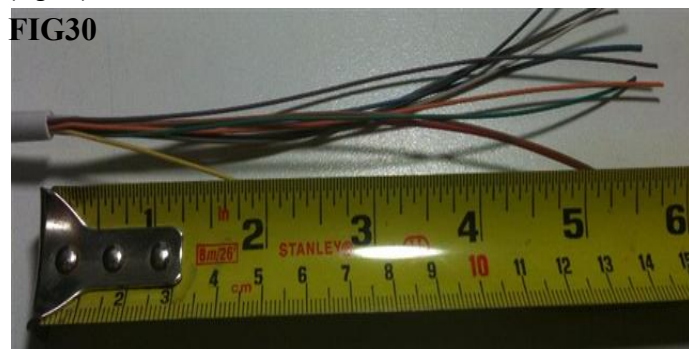
Using a multi meter check which colour wire is attached to each pin
(Fig 31).
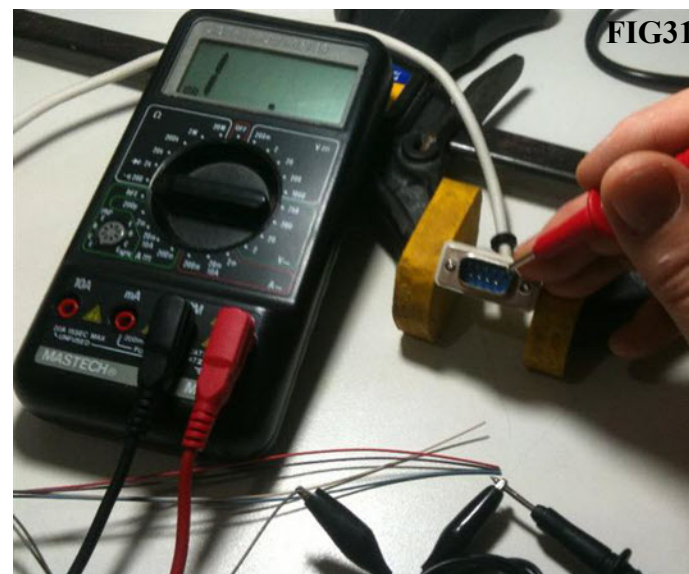
FIG31

If you look very closely at the pins inside the plug, you will see that they are numbered from 1 to 9. Make sure you write down the results as this is where most of my mistakes happened. You will notice that we don't need all ten wires, see below for details.

Serial Pin 1 – Ground
Serial Pin 2 – Ground
Serial Pin 3 – RED
Serial Pin 4 – GREEN
Serial Pin 5 – BLUE
Serial Pin 6 – Intensity
Serial Pin 7 – NOT NEEDED so remove it
Serial Pin 8 – Horizontal Sync
Serial Pin 9 – Vertical Sync
Serial Spare wire from shield casing – NOT NEEDED so remove it

We now need to crimp on the tiny connectors for the 6 pin block connector for the A22 board. I consider this bit to be the most difficult part as it is a very fiddly and delicate task to perform. Take your time and be careful when making these connections. Refer to the diagram
(Circuit 2)

Circuit2

for the pin connections on the A22 board.

1 – Vertical Sync
2 – Horizontal Sync
3 – Ground
4 – Blue
5 – Green
6 – Red

Wires 1&2 need to be twisted together and crimped into the same connector
(Fig 32) **Fig32**
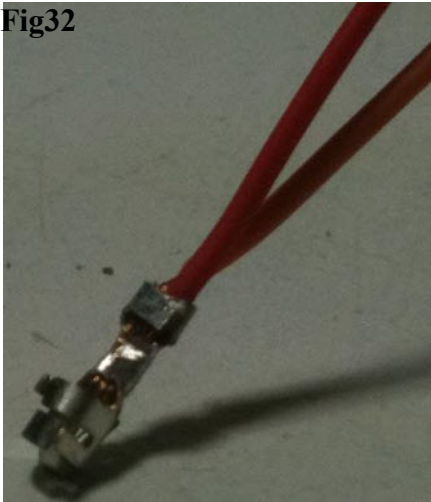
and slotted into the connector in the '3 - GND' position. When you have crimped all the wires and inserted them in the connector you should have three wires left over
(Fig 33),

Fig33

one is the Serial Pin 6 Intensity wire but the other two are not needed so they can be cut off and discarded.

DO NOT attach the connector to the board yet !
Stage 5 – Attaching the daughterboard
Using the cable you have now completed, we need to cut the RED/GREEN/BLUE signal wires in half and also cut the INTENSITY wire to the same length
(Fig 34).
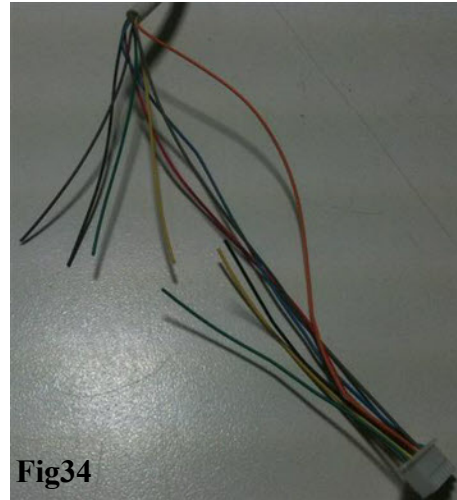
Fig34

These wires now need to be soldered onto the daughterboard
(Fig 35).
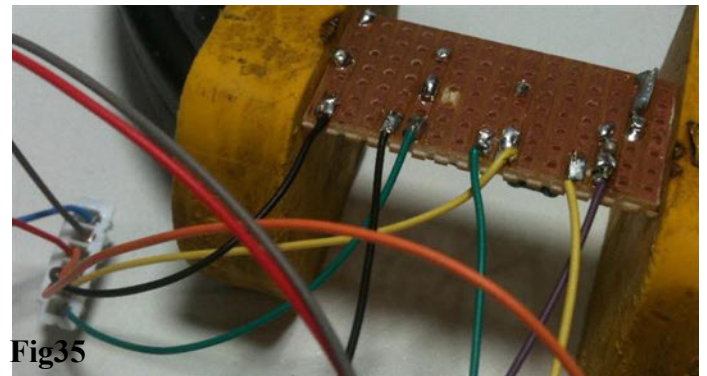
Fig35

Again, follow the circuit diagram to make sure you have the wires in the correct place.

Stage 6 – Final assembly
Your completed cable/daughterboard assembly can now be fitted into the casing
(Fig 36).
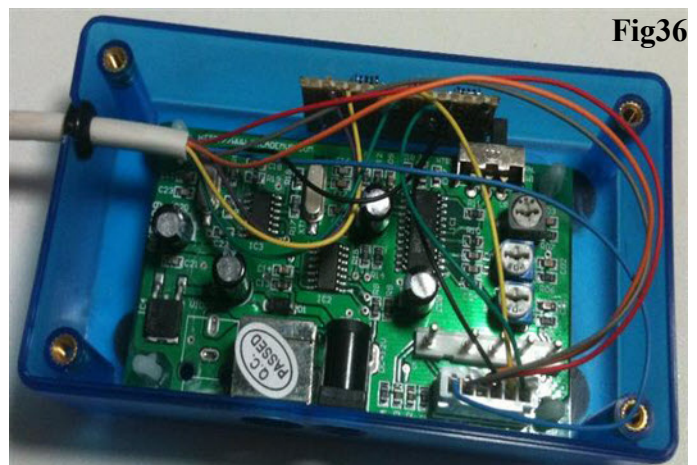
Fig36

Make sure the connector fits firmly and that all the pins are in place as sometimes they pop out of the connector when being connected. Put the grommet & cable in its location hole and fit a cable tie to the cable 'inside' the case so it doesn't accidently get pulled out. Place the daughterboard to the rear of the case and feed the wires around to make them tidy. The daughterboard does not need securing in place as the wires hold it well enough.

Important:
there are three gain screws on the A22 board (for RGB adjustment), these need to be set in the following manner – fully anti-clockwise and then clockwise about 1mm. The PAL/NTSC switch should be set to PAL.
When you have tested it and are satisfied that all is well, put on the cover and screw it down (fig 37) – job done.

Fig37

Powering the RGBI>S-Video
The A22 board can tolerate anything from 8v to 15v DC so anywhere between the two should be fine. I have been powering the units with a spare 9v 4amp DC from my CS-SuperPSU and it runs just fine.

Troubleshooting :
Building something like this for the first time is going to give you problems and I certainly was not immune to them in the early days. I found that the most common reasons why the device wasn't functioning were because the soldering on the daughterboard was 'touching' its neighbors, check that and rectify it. Another fault which took me ages to locate was that one of the tiny strands of wire had shorted out across its neighbor, again, check this and rectify it. Another problem could be the actual monitor itself. I have only been able to test it on two different monitors and both function okay. All the units that I have built for people work okay and no one has yet to report a problem with any monitors, however, that does not mean that this problem won't crop up in the future.

Acknowledgments & Reference :
This project has taken me all year to perfect and it could not have been done without the help of the following people : Nigel Parker (commodore free), Shaun Bebbington, Chris Syntichakis, Zetro, Robert Willie, Ray Carlsen, Bill Herd & Kevin Krausnick.
It has to be said that the final push came from Robert Willie and he details his project in full at http://sites.google.com/site/h2obsession/CBM/C128/rgbi-s-video .

I strongly recommend you go there to see earlier variants of this circuit design.

Details of the A22 board can be found at http://arcademvs.com/ARCADE_ACESSERIOR.htm .
I thank everyone for their words of encouragement and help throughout this entire process.

Pricing :
Now that you know the cost of parts and how difficult it is to build I hope you understand the pricing structure I have put in place. However, if I build one for you, it will be tested, 100% functioning and will come with a test program on 5.25" or 3.5" disk. Each unit is personalized to you with a unique label and serial number.

The prices include
the Mk2 RGBI>S-Video Adaptor (v4a),
a test disk (State 5.25" or 3.5"),
information sheet with proof of functioning.

CCCUK members £70 (some of proceeds going to club funds)
Non members £? - eBay auctions only
Postage & packaging - UK £7.50 (insured)
Postage & packaging - World £25 (insured)

Ordering :
Do not send any money 'up front'. Because, at best, I can only make possibly 2 or 3 of these a week so I won't take any money up front as I don't want people saying to me "I've paid you so where is it ?" every five minutes. The best way would be for you to send me an email (allan.bairstow@btinternet.com) and I will put you on the list for January/February 2011. First come first served basis will apply. When units are ready I will contact you to confirm you still require it and arrange payment.

Future developments :
The Mk3 will have a 'swappable' daughterboard to easily allow different daughterboard configurations to be used. I am currently testing different daughter boards on the Arcade RGBS CGA/EGA/YUV to 2 VGA converter which will 'hopefully' allow a solution to holy grail of a C128 80 Column Colour to SVGA/VGA problem.

Copyright:
All information in this article is given freely to the Commodore community and is not copyrighted in any way. November 2010.

COMMODORE PROGRAMMING
THE WRITE UPS FOR ISSUE: 45

How very cruel would I be, if I didn't come back to write up the explanations for the printed programs?

...Well, goodbye then...

OK, OK! Don't all cheer at once!
I do hope however, that by now some of my explanations won't be needed. Not by every-one anyway -just those with loads of money to spare who would gladly part with a small fortune for more tips on controlling ...err, interacting with their computer. So without much more waffle...

I start with GShape, & GShape1, pretty similar programs that attempt to show some of the c16/+4's drawing capability. Focusing on not just drawing the shape in the first place. Part of the shape is saved to memory, RAM in this case. Only for part of it to be redrawn on the screen elsewhere.

I can only guess that when commodore removed sprites from its capability, it was intended that this sort of routine would be used. The POKE command though is still very much in preference from the material I've read, as it seems to allow better scope for motion and quicker responsiveness.

Page 90, of The "Programmers reference guide for the commodore plus/4, Merten &Meyer" is referenced as being the place where these initial programs were obtained for our learning.
I made a reference also to page 45 of the same book, as a possibly good avenue as a way in to someday obtain a better understanding of the area being saved "Sshape", as well as to the rules of how much of the saved shape could be Got "GSHAPE" all at once. As illness takes its toll, this remains a mystery.

Lines 7 & 8; make altering the program easier. These being added by yours truly, again in the hopes of working out how to save different parts of the picture and put it elsewhere. Evidence of my starting this experiment already can be seen if you compare the program with the listing in the original manual.

Lines up to 40 draw the initial shape on screen.

45 to 55 Save an area of the picture (Actually, this takes one line.) The other two here yak on about it to -err- save me typing a lot here!!

60 Wait for any key to be pressed (This has nothing in the program to suggest it is what is required so, either assumes the user has read and understands the program, or relies on that time in games that I am sure you all remember, where a game crashes and you frantically press each key in the hopes that one of them will free it up. Cunning, very cunning!)

70 user has pressed a key, so draw the saved shape.

80 as 60 -"yawn!"

90 return to normal screen

200 A note for those who may wish to study and expand the program. The manual is available online.

flags1
If you're thinking where's the Union flag, The St. George, St. David, St. Andrew, St. Patrick etc. You may note due to the mostly emptiness of the screen that this program is unfinished. There is a good reason for this. As I was working through the book, something made me sidetrack to The flag of Japan, I think it was a film, but a I actually wrote this a while ago (necessary in case proof of copyright were ever needed, albeit in this case for the sake of good practice). For some reason what should have been nice and easy, perhaps the simplest to do became the hardest. And I cannot figure for the life in me why this should be. I left it down to manufacturing error in the absence of any professional guidance. -I will keep looking, and if any-one knows please let us know.

10 tell us what it is.

20 set background; main colours, though this takes a whole new meaning in the GRAPHICs screen. Will come to later.

30 An absolutely pointless command that survived all debugging. In truth is evidence that when I came to the stumbling block -that is to say the problems that as yet remain unknown regarding drawing of the flag of Japan. Is therefore evidence of varying experiments to try to rectify the problems that occurred.

40 set the required screen

45 This is supposed to say "goto 310", which was put in place so that when I was trying to troubleshoot the Japanese patriot emblem, I didn't have to wait for all the other flags to render. How the typo's got in, I do not know!

50 set the drawing color; first flag in The "An Introduction to BASIC part 1" book, and so I wonder if Prof. Andrew Colin had links? And again, if any-one knows where part 2 for the c16/+4 can be obtained Please, Please let me know!

60 draw a box with the relevant first colour.

70 change colour for second box and

80 draw it.

90 & 100 same again for last box.

You will notice that I had to draw extra, that the following box partly writes over. I found it necessary to do this, and you will see why if you experiment with the box numbers. Quite why this happens is anybody's guess.

Lines follow the same type of pattern throughout the program. The size of box being relevant to the required shape, and drawn in the appropriate colour of that nation. With as near a match as I could get to the respective nation. This being the third number in the COLOR command.

Having studied the rest of the lines, one will come to 310 in due course.
  If you've run the program already you will note the bottom corners are wrong.
  Try reinstating line 45 as mentioned.
change line 320, either put REM in front of it or delete it completely as it seems to be utterly pointless, though it is supposed to place the pc. "pointer cursor" which only the plus 4 reference seems to have the correct use for. See page61. For those who fume at books that leave it at that... the DRAW command can either be used with "from" co-ordinates, as well as the "TO" co-ordinates. With Locate on one line, the DRAW TO ... command is used.

<<<BREAK-THROUGH!!!>>>

It shows: If in doubt. With a long program, try a shorter program to fix the fault! I therefore include "Japanese3" because this is exactly what I did. And have learned that my error was in the GRAPHIC command. Numbers 1 & 2 do not allow for the multicolor function. (i.e. COLOR2,X OR COLOR3,Y). Strangely, everything is drawn much bigger. I had the size I wanted with "GRAPHIC1" but actually I can live with this. Change line40 to "GRAPHIC3,1" -where the x,1 clears any previous screen.

"Ah-ha!", it was in the manual all along, but being uneducated I didn't properly take in the cold hard terms. So for others who require good examples and sometimes quite lengthy explanations I have shared my humiliation/s here.

If you are using an emulator such as Yape. You may, if you wish, delete the lines 310 to 410 in Flags1. Then save it. Open Japanease3. Renumber lines as "RENUMBER310,10" Clear the screen completely (scnclr). then type list. edit out the "ready." prompt. Right click on the mouse, select copy. Now you're ready to go back to Flags1 and paste the program to append it. Once you've done this, check it works. And save it as "Flags2".

Hurray! This has saved me a lot of painstaking explanations, looking for ways to explain why certain lines had to be added, and trying to come up with reasons for "why this happened when..." etc. It proves: The solution is better than many answers!

There's still errors in the program! Can you work out why the middle looks too big in France & Italy's flag, and the cross is now off centre in this rendition of Swiss flag. These co-ordinates shall have to be altered in due course. First I will run

through the lines that we have appended to the main program.(as renumbered)

310 You may wish to change this to simply say Japan, to be more in line with the other titles. I now recommend putting other commands separate though, in this case COLOR gets her own line.

320 If you have corrected line40, in Flags... then you don't need this repeated. Either delete it to save memory, or place a REM before it to serve as a reminder of the exercise in trial and error.

330 "For those of you watching in black and white" this gives a dark purple drawing surface ("foreground"), and a black background.

340 again a debuggers checklist

350 The color is set as black to draw the box with. To look more professional, I changed 1 to 2 (from black to white) to correspond with the backdrop of the flag. If you are still unsure as to what is happening, you may experiment with this number. (but don't upset the patriots of the respective country!).

Note:
I have used flags as something easy to draw, to help the beginner to move off the beginning blocks and gain as much understanding as possible to how the c16/+4 graphics/drawing by co-ordinates etc. work.

360 special color2,x only available with "GRAPHIC3" or "GRAPHIC4". Needed here to fill box in white to go behind the following red circle.

370 draw box that is the outline of flag.

380 paint the area in the set colour.

390 set drawing colour to red. Note "color3,x,y" has not been used, we know this because there is no following paint command that refers to it.

400 draw the red circle/oval.

410 Paint in color1. i.e. "Paint1,x...". (To experiment change to paint3,x,... ...And why this happens is a complete mystery to me! - I got the screen rendered in a pink-ish color.)

lines420-450 You may wish to delete for the main program as this is covered later in lines9300+

9300+ Put here so you can have as many flags as you wish before ending the program. You don't have to wait for me to finish the program as tinkering with it yourself, such as the change in some of the co-ordinates that is now required. Looking for yourself now it is clearer what is going on may be the best way to learn!

9390 waits for key. Changes colors back to something writable.

9400 Gives main screen back.

Restore

As stated before these prgs were merely to test something. The actual proper use of the restore command is to reset the DATA Counter. So that when it gets to the end of the list; you can start again at the beginning. And with dummy-READ statement loops, that is the data is read as a variable, but nothing is done with it. You can actually move through the list to where you want to be. This is demonstrated by the changes between "restore" and "restore4". IF U=Still <> sure? THEN READ The following: ELSE Learning=0

If you began with this tutorial to learn Basic3.5, this may be a bit more complicated debugging than you may be used to here, especially if it is your first computer language.

10 starts with the beginning of The FOR..Next loop. The pass is set to 20.

20 This statement is READ which means that DATA is to be -err- Read! (The clue above is that the maximum allowed is set to a count of 20.)

30 As line20 states that the data statement is to be read, and set to the variable called A$, and as now the computer is told to PRINTA$, thus it shall Print the current data statement in that memory location that we have told Basic to call A$. Simple!-Yes?

40 Once this is done go to the next data statement. Only until the count of 20 is reached.

50 Once at the end of the count, Ask the Question,

55 Wait for answer via a key press. Check the result. If continue=No Then end there and then in an abrupt manner that will produce the line list -in case more debugging is required.
If the answer to "continue?" is Yes, or anything other than No, then get the next count in the FOR..Next loop.

If you've RUN the program already, you will know that after the second 'Y' press/ Yes request. The program spits out an "?Out of data error in 20" statement. Now, we see 20 is the READ statement. And as we know that as this line has worked for us in the past that surely it can't be wrong all of a sudden! ......And it isn't! The problem is the computer attempts to continue reading from the DATA list of statements, but these run out.

Before reading on, come up with and try your own answers for sorting this out. Note that simply adding more DATA will still eventually bring up the same error message.

I gave a big clue earlier. We see that there is a "Restore" statement already in the program. And you will know by now, that this sets the Data Counter back to the beginning of the list. ?Closer analysis of the program reveals that line 60 is never read by BASIC!

To make the program work. The easiest thing to do is to alter line 55. Take out everything after the ":", and after "RESTORE" on line 60 add ":GOTO10"

Finally change line10 to count to 39, if you want the whole list to be read and printed when running.
You may of course make your own lists, if you wish to use symbols or colours obtained via the keyboard etc. You will need to use "inverted commas".

RESTORE4

What is going on here? You might well ask.
If you RUN the program, You will find the first continue prompt will continue to read the full list of data statements. With the questions as REMarks at and after line1000, you may have worked out the answers for yourself. "Bravo!", if You did! you, no doubt, will also notice other things about this program that are different from the last. Numbers are placed after the words, and less importantly -here-, if the word is long, the number jumps out of alignment.

I'm not going to go through the full program again, as most of it is already covered. The main differences are as follows.

line 20 adds a colon ":", and a mathematical count setting the variable X to add 1 at each pass-through of this line.

30 as well as printing A$ to screen, we are now to have the count of 'X' printed using a comma ',' which has a specified number of spaces to place the next word after the last one. An alternative is the CHAR command. See prior issues of this excellent magazine.

We see that line 55 has already been changed as required, but line60 is not as listed in the Restore program above. Let's look at the new line.
60 IF X > (is greater than) 21 THEN perform the RESTORE command. (i.e. Reset the Data count)

65 continues the program, if yes was answered.

Now, the more astute among you will see that line 60 poses problems. If you want the computer to read out the full message every time, that is. So, how do we go about doing this? Again. Try to work this out for yourselves before continuing here.

OK. Line20, you may realise just keeps going up, there is nothing to say that when we get to 21 X should represent 0 again. And perhaps we don't want it to? We might wish to keep an ongoing score, or count. Yet we clearly need the full message. because X is always greater than 21 after it has gone past the number, it will therefore always reset the DC on encountering that line! So, what do we do? There are a number of options, the best is: 1. link the restore to the variable J in The For..Next loop! but make sure the count is again for the correct number of DATA statements. If you want to wrestle with the program try 2. Add another variable 'Y', and when that reaches our number, reset both the number and restore the DC. (Data Counter). With the later you must be very careful. Again try your own.

25 Y=Y+1
60 IF Y>21 then Restore: Y=0
NB: This still misses words from the list, so try putting dummy words to the list, and alter the counts Y and J accordingly. This would be considered something of a 'spaghetti method' but that's not to say it won't ever be needed. - Still, we live in hope!

Code Converter

While I dedicated my last article to be named "Things I don't understand", with the work done purely in this write up; I name this effort as "Coming to an understanding". I would not have figured out Flags, had I not done the write up. This alone shows how important it is/can be to write the code out in plain speak. (I would have said 'plain English' but this may not be your mother-tongue) as well as in the computer language you use. I realised this then I thought... that is its own 'code converter'!
...Yes, John!

OK. My first effort, and real look at ML, with any real thought that it might, just might be possible for little me to have a go -one day. How nice that would be to develop some understanding of assembler, as it was always a mystery and wonderment in my early days with the sixteen.

Unfortunately, the one printed in the issue45 is incorrect. And I only realised this late in the day, clearly after the deadline for the magazine had passed. My sincerest apologies for any inconvenience caused. The corrected program has been submitted, and to the best of my knowledge is now making correct mathematical calculations. I only hope my error didn't put any-ones program out. I am, however sure, positive that any-one who is clever enough to understand and program in Assembler Language must surely have their own methods of checking the validity before using anything new.

The features of the program include:

Press   function
0-1   Will draw up a list of all the multiplications in decimal that total the numbers you require. You can choose to list everything from 0 to 65535, or be more selective.
        Pressing the RUN/STOP key at any time will return to the main menu.
        This function goes through all combinations, thus takes the same amount of time whatever number/s are requested.

2   Type in 00FF, this equals 255 in Decimal, and a row of 8 1s in Binary.

3   This time type in 255 to get 00FF etc. Note: These are just examples.

4   Binary. Type in a row of 1s. To get Hex=00FF, and DEC=255.
        The computer will allow for a base2 high byte to be typed in. And then it will convert the sum of both numbers in the above formats.

5   As RUN/STOP Key has other uses. It makes sense at least in the writing and testing stage to have an alternative way out to get into the program. With a couple of tweaks this could be '0', which is at the other side of the keyboard, so as to minimise a mispressed key.

The write up of the corrected version.

If this program is RENUMBERED by 10, it comes to 1530. A hundred and fifty three lines is less over facing to type than what it looked like before! I will Endeavour to make the listing in the original text listing given though I consider the program now as complete as I can make it. Often big jumps in numbers represent a new sub-menu, and can be very interesting to read. *Also, as there were rumors that the "editor" reads this in the bath! *EDITOR COMMENTS "yep True"* I don't want to get on the wrong side of him by making him have to get out, renumber the program before he can read it. -the thought itself is scary enough!

10 An unusual start. Assume the usual "copyright by John Fielden 2010", unless there is something already available that does the same thing. I am allowing public use of the program, usual public domain licence for all work published here, by all means those who have nice things to send, do. If you look through the magazine issues long enough, you will find my details.

20 as it says, home twice. Followed by shift home which equals clr/home key. This is to get rid of any sub-screens that may be in use, though you may prefer this as a separate variable, the program title is included. Also to prevent repetition, commonly needed page editing is given over to variables. A trick I learned while studying the Star Trader game. Available from www.commodore16.com or it was last time I looked.

25 -80 sets up main menu. Sometimes in programs I have done/seen this later or even last. It is best to leave space for it until you know how many options you are going to need.
30-35 important prog. variables.

100-130 part of menu, a space in program numbering is a hint that the absent minded programmer didn't know if he was going to need to dish out any more options.

135 The TRAP command, in this case says, in case of error Goto line 5010. Numbers ending in 5 tend more often than not to be an afterthought. Other numbers that end in anything other than 0 most likely are. We see in this case line5010 RESUME 30 sends the program back to the main menu. It may be worth noting that because the Trap command is put after the menu, run/stop can still stop the program while the menu is being printed, but only directly after typing RUN{return}. You may wish to put this higher up your programs in their final draft. But make sure you get the alternative way out sorted first!

140 wait for a key press.

150 set the variable X to equal the number of the key pressed.

155 specifies a meaning if 0 is pressed, and includes a special function. NOTE: It is always good programming practice to let the user know of such things -shows how "good" I am - dunnit!

160 The 'onxgoto' statement. This corresponds to the menu, and highlights where all the main sub-menus are. As pressing 0 or 1, sends prog. to 8000, let's see what is going on with the variable introduced at line155 "s1%". ...

A look indicates that some kind of on/off switch is being used, and may have something to do with the pause feature. Though as it has been awhile, I will have to test the prog. to find out what the difference is by having the switch on and off. ...

I still can't see any difference. It may be something I later decided against, or something I had in mind at the time that I didn't quite remember to put in when I came to it. It shows to do a synopsis, brainstorm etc. At least at the same time as writing the program. I must confess at this point that I've always seemed to have had to dive straight into the program. In childhood I had no thoughts for anything else, but developed bad habits -like this!- that way.

My third trick these days is to print the listing out on to paper. And study it in my own time lying down with it over my head. -I hear Buddhists do something like this when studying... Why not I? An option I didn't have in youth that has since been known to help -sometimes. Also leaving it for a while, then coming back to it when all the frustrations are gone out of my system.

170 if it isn't a listed number don't do anything, just go back and wait until a key is pressed that represents a relevant function.

180 key 2 menu.

190 - 200 Get number, give it a variable H$

210 An easy conversion and saved in memory reference to the corresponding Decimal.

220 make legality checks

230 print first conversion, if within range.

235 proceed to binary but remember this point.

Multi-menu
240 on return from binary, see if user needs another hex conversion.

250 wait for key

260 this shows itself to be a general purpose sub-menu. So, whatever X code is registered, that is where the program will go if Yes is answered.

280 or main menu if Y is not pressed.

300 Key 3 menu.

310-320 set up Dec variable.

330 get rid of unnecessary points, nb: the % symbol cannot be used with higher range numbers.

340 ensure choice is in range.

350 direct reverse of 210. This time the Hex equivalent is being shown of Dec, instead of vice-versa in210.

352 & 355 together determine the choice of the user and react accordingly. We see the 5500 is for binary. The choice determines whether the user chose to get one answer or find a number of them.

360 job done! Goto the communal sub-menu: "Multi-menu".

400 Key 4 menu: Binary input required!

402 Is there a hint of DATA & READ statements about!

405 set cursor' (Yes CHAR might work -but it has its limitations too!)

410 Do what you are told!

415 here's a suggestion "Use Poke command to print "X"'s prior to number." That is to go under the number rather than above/below it. -Well, what d'you think? I left it so far.

416 & 417 show the other idea.

420 start the required count using 'P' as variable name.

430 wait for a key.

440 ignore if out of range.

450 If in range set required variables to memory.
I consider programming would be easier if the computer had a shorter term memory that could be cleared separate to the long term memory!
...What did you say?

460 screen plotting.

470 start required count using 'K'.

475 This is one way to remove unwanted errors. For example, can be caused by a return key press that would mean the value already in memory being added. This would cause problems in cases of an accidental press. The extra loop may be avoided if you can find a way to make sure that the number selected is going into the right BIN-Array slot.

480 The number has to be put in memory and go to the right place on the screen. 'K' holds the number in sequence.

490 & 500 Printing does not happen without this sequence. (470-500)

510 start 'T' loop. This will be the mathematics so we can convert the BINs!

520 Read Data. See 570.

530 see notes. crucial sum for converting BIN to DEC.

540 Add current sum so far in line530 to variable C.

550 continue the sum.

560 restore: It would have been better, perhaps to put this at about line30ish. Because the closer to the communal start point a command that uses the same code is, then the less it needs to be repeated; thus avoiding spaghetti code and saving memory into the bargain! The reference at 402 uses other menus for its comparison. But the effect here is that the reset comes last which does not take account for some-one using the CONT command to restart the program. This could cause problems. Such errors as this which was spotted late in the day, and as I found this project already somewhat adventurous; I was quite jaded by the time I thought it was completed -so it got left, forgotten and printed -(there goes that analyst job!)

402 seems to be the surplus extra command.

570 On completing the above loop. The computer has reset already, and as it now has no order to READ these DATA statements so it doesn't! And we are still ok for the next time when we need the computer to re-read them.

580 confirm final Conversion to base10.

590 confirm base16 conversion.

595 This is a menu selection control switch, stating that if 4 was not pressed then we don't want the high byte formula. The line could've read if X<(is lower than) 4 then660. But if we are going to study Assembler someday, then we must learn something about conditions and switches. (I think)

600 Our switch being off the program asked the user whether they need to calculate the high byte. -Are you asking "What on earth is that?", as yet so am I!
610 wait for response.

615, 620 If No; reset everything that may be needed next time and see if another is needed.

625 If user is drunk, talking jibberish then go back and wait for a more coherent response!

630 if here, Yes was chosen to question@600. Show user choice: (Not essential) BY% Is switched to show second run through. And the sum converted now to decimal is held in C!. C is cleared so that the computer doesn't come up with an out

of range number. Byte is placed on screen as low byte. (hence being moved down screen, it just seemed apt.)

640 Print the low byte in binary.

650 second lot required! Must get more 0s and 1s for high byte.

660 This is the sum of low & high byte when added together in decimal.
with 665 the book that I found my correction from Iss45 in is also referred to.

670, 671, 675 print totals in DEC and HEX using the totaled variable holder C2.

680 reset the sum variables.

690, 700 wait for user to read data/take notice of the result/s.

710 Another surplus command because this is done in the next line. In case of debugging, because this line is repeated on the very next line. It would still not be apt. yet to just delete it. Running through the code we see 750 is referenced. Even if 710 had been referenced the following bug fix (in programming the term "solution" has come to mean something else). Change 710 REM.

750 Again make sure all variables to be discarded are reset. (the repetition in 680 may not be needed)

760 go to the menu that checks if another sum of the same type needs to be found.

1000 If you haven't yet seen a reference for this in another line, or you wish to practice your debugging skills. For now place STOP: REM ...(then let the line exist as is, until checking). -It is actually referred to at line260. And goes to Bin to hex & dec after clearing screen. I could not do this at 400, because of the hi+lo byte feature. -or, I could've used a switch, I suppose.

5000 Number 5 was pressed. It might be an idea to add a check to see if this button was meant to be pressed. -if so

5010 would be changed to 5400, and the line/s that refers to it would have to be changed accordingly. -In this case there's only line160. This error trap has already been covered.

5500 This Sub-menu is used in both main menu choices 2, and 3 for the conversion TO Base2 numbers. While on the subject, BASIC3.5 is ideal for this type of program because of the built in Functions, something that the makers seemed less keen to keep in later versions. (though I don't have the reference guides for BASIC7 etc. to hand, I find no evidence of it in AmigaBASIC for WB1.3. Though there is a hint of it in GFA. nb: Not a thorough search) -The point is I do not see how such things that make life easier can be dropped in a later version and yet still call that later version 'an improvement', which surely is the ideal ...or would be if profit didn't enter into the equation!

This sub-menu is the reverse of the sub-menu previously covered from 400. The program has to break down the singular DECimal figure. If the program reaches here from 'The Hex department' it will have already been converted to DECimal.

5510 A message for the user is always a good idea before calculations have to be done, especially if these are complicated, lengthy etc.

5512-5590 Break the sum down to its BINary components.

5600 I don't know an instance of why there would be, and in checking the program thus far there hasn't been an error. But a good rule of thumb is 'You never know!'. So for thoroughness a check in case anything goes wrong.

5605 & 5606 If this looks familiar, see 416 & 417. Should I have used a switch? ...for two print statements? err, NO! And you will see These numbers represent the amount of each base2 number in the array.

5610-5630 Print to screen the recorded results. i.e. The 0's & 1's.

5640 clear a line. (necessary?)

5650-5670 clear the used variables so that a future reference doesn't add up what is already there as this would result in a ridiculous outcome. (I would be sacked from even this voluntary venture, and forget any possibility of even a future analysts job!)

6000 go back to the menu it was sent here from.

8000 The special feature.
Because it was found that numbers higher than 255 would need the hi+lo byte system. And would probably have a varying amount of different combinations. It may be unnecessary to list them for Binary as well. So once the decimal is found. The user may then use this sub-menu to see all the possibilities that may be presented for that particular number. It takes a while though, and if you have an emu. it may well be worthwhile speeding it up for this process. -see notes.

8010, 8020 often it is easier to ask the question in a PRINT statement prior to the INPUT command, even though the latter has its own Question mark.

8030 more range controls.

8040 get the upper range within the max. allowed.

8050 this time the range check has to be more in-depth because not only do we have the maximum, we also have the new temporary minimum to consider because this number cannot be below the first one or the program won't work. Once correct we can continue.

9000 The Pause menu, in case the screen is running out of space, as the user may wish to make their own notes. The emu. may need to be slowed down for the key press to be realised.

9010 Uses variable set for cancelling any sub screens that also prints the program title name.

9020 remind user the range, as well as their currently chosen range.

9030,9040, 9050 denotes whether switch is on or off.

9060 Using a sub-screen here will allow the listed numbers to scroll, whilst keeping the main
information such as title. Hence a more professional look and feel.

9070 start search for codes that add up to the decimal numbers selected within the ranges.

9080 be open to a user key press.

9090 if 'P' is pressed send program to Pause menu.

9100 if 'Q' then exit here and return to main menu. As no maths are used with the ranges m1 and m2 do not have to be cleared.

<DEBUGGING TIP>
You may notice any key press sends the program to the pause menu. My first suspicion was that the problem was here it isn't. So I wrote an unnecessary bug fix. Can you see why it is unnecessary?
To correct this add   9085 ifa$<>"Q" ora$<>"P"then9110 do net send the program back to 9080 because this would cause it to freeze.

9110 use an alternative variable for decimal. This is done to avoid confusions etc. that would otherwise occur at some point in a program.
9120 Use part of the program already built to get relevant information for this process. (I saw Quantum Leap, film the other day: And looking at this reminded me of it). I will leave the reader to study the code as it provides the best explanation of -by now you should understand- what is going on.

9130 as I've taken this out, I'm not now sure what it does. Suffice to say I haven't missed it.

9190 Some say this line is unnecessary due to the following print command.

9200 tells user what is happening.

9210, 9220 A main loop with another loop inside it.

9226 see "9085"; the actual 'fault is here'. Why does this need to be here, what fault did I notice? As I can't remember I will have to look at it again. line10... (only kidding). If I wanted

any key to call for a pause I would've said that in the first place!

Q. Is lines 9080-9100 being actioned at all? -And if so, where?

A. REM the relevant lines and RUN the program to find out. Lines 9210 & 9220 show that we are in another loop separate from the last, but -

Q. Could these be more adjoined to save on typing repetitively, and to keep consistency.

A. is Yes. ...

Q. How? ...While getting the key is needed in each sub-program. The question of what key is pressed can be written into the actual pause menu. So 9090 & 9100 would go into the pause menu, while the line9226 here can then be copied to 9090 with the other line9100 not needed there.

9230 start to test the result of each combination to find numbers that add to those within the chosen range.

9240 remember CF is not the same as CF(x). The former replaced D. earlier. While the latter is a switch.

9250,9260 continue search by adding one to count. When inner count gets to end the outer count will add 1. And restart the inner count to its beginning.

9270 At first glane, I take this to mean that the first thing to do when the outer loop finishes is to see if there is another of the same type to find. Only if the switch is on and the main menu choice was 1.

9900 next number in the chosen range.

9910 when all loops are completed. Reset variable. IMPORTANT NOTE: The % in 'cf%' may be a typo. As I don't seem to remember seeing it anywhere else. (oops! ...Dear Microsoft, I am looking for a job...-sweeping floors!)

9920 All done! Back to main menu.

12000 The Pause menu.

12005 Let user know that nothing is going on at the moment.

12010 Wait for key press to decide what to do next when user is ready.

12020 if Q or E is pressed: Goto main menu.

12030 Delete the line that says "paused". And RETURN to whichever Sub-menu directed the program here.

12040 If keys don't match any of the choices listed, go back and wait until a key press does.

Notes:

VARIABLES USED

Line	Var	Description
20	XY$	screen plotting.
25	M1 & M2	Set range of allowed limits to convert.
	XL$	more screen plotting.
30	CF (-1)	see also 352 = control variable user choice dependent. For use in the menu@8000.
35	BY%	see595
150	X	remember user conversion choice.
155	s1%	Switch -of some kind.
200	H$	Hex.
210	D	Dec.
450	B%(P)	used to add up the binary to DECimal. In this case sort out where the 0s and 1s go in the array.
520	B	allows multiplication of user' selected 0s and 1s.
530	B%(T)	as B in line 520. Part of the calculation.
540	C	current BINary number chosen.
595,630	BY%	The default set in 35 is 0. Menu selection control.
660	C2	high + low Byte sum total.

(FUNCTIONS)

lines 210 & 350 refer to command types that are called "Functions". In User Manual (c16) pg.144 for DEC(variable$) which is classed as a "Numeric Function". And Hex$(Variable) on pg. 150 (referred to as "in the manual") is classed a "String Function". ...More to come on Functions. However these are limited, especially with screen editing. See line405. And "Lost Plot" coming soon.

(Emulator/s)

With most emulators, the speed may be increased. In Yape, there are two ways to get a faster program. And You can use them both at the same time.

Try 1. Increase memory setting. left click machine; in file list at top of Yape window. Then select "configuration", Select arrow in "RAM configuration:" And change to 256 KB...

 2. 'full throttle', "Alt+w" or left click; machine speed, No Limit or machine, "full throttle". This makes the emulator as fast as it can go.

There are other ways to increase speed. Such as to "Low compatibility" in configuration, compatibility menu.   ...But I haven't tried these and so "enter at your own risk!", and save back-ups regularly.

Happy Prog'ing!
John Fielden:

LISTING OF: CODECONVERTER.PRG
BY JOHN FIELDEN

```
10 REM *HEX TO DEC TO BIN AND BACK*
20 CC$="{home}{home}{clr} CODE
CONVERTER":XY$="{home}{down}{down}{down}{down}{down}{down}{down}{down}{down}{down}{down}{down}{down}{down}{down}"
25XL$="{home}{down}{down}{down}{down}{down}{down}{down}{down}{down}{down}":M1=0:M2=65535
30 SCNCLR:PRINTCC$;"{down}":CF=-1
35 D=0:C2=0:C1=0:C=0:FORT=0TO8:B%(T)=0:BY%=0:NEXTT
40 PRINT" 1 - ALL CODES FROM 0 TO 65535"
45 PRINT" (0 - TO SWITCH BETWEEN CONTINUOUS LIST)":PRINT
50 PRINT" 2 - HEX TO DEC & BIN."
60 PRINT" 3 - DEC TO HEX & BIN."
70 PRINT" 4 - BIN TO HEX & DEC."
80 PRINT" 5 - TO EXIT"
100 PRINT"{down} HEX=HEXADECIMAL, BASE 16"
110 PRINT" DEC=DECIMAL, BASE 10"
120 PRINT" BIN=BINARY, BASE 2"
130 PRINT
135 TRAP5010
140 GETKEY A$
150 X=VAL(A$)
155 IFA$="0"THENX=1:S1%=S1%+1:IFS1%>1THENS1%=0
160 ONXGOTO8000,180,300,400,5000
170 GOTO140
180 REM *HEXADECIMAL TO DECIMAL*
190 PRINT" TYPE HEX TO CONVERT:"
200 INPUTH$
210 D=DEC(H$)
220 IFDEC(H$)<0 OR DEC(H$)>65535 THEN200
230 PRINT"DEC: ";DEC(H$)
235 GOSUB5500
240 PRINT"{lgreen}ANOTHER Y/N?{black}"
250 GETKEYA$
260 IFA$="Y"THENONXGOTO9900,200,320,1000
280 GOTO30
300 REM *DECIMAL TO HEXADECIMAL*
310 PRINT" TYPE DEC TO CONVERT:"
320 INPUTD
330 D=INT(D)
340 IFD<0 OR D >65535 THENPRINT"RANGE: 0 - 65535 ON THE C=16/+4":GOTO320
350 PRINT"HEX: ";HEX$(D)
352 IFCF>-1THENGOTO5500
355 GOSUB5500
360 GOTO240
400 REM *BINARY TO HEX & DEC.*
402 RESTORE
405 PRINTXL$;"{down}"
410 PRINT" TYPE 8 DIGIT BASE 2"
415 REM *USE POKE TO ?X'S PRIOR TO NO.*
416 PRINT"{lred} 7 6 5 4 3 2 1 0 "
417 PRINT"{orange}128 64 32 16 8 4 2 1{black}"
420 FORP=0TO7
430 GETKEYA$
440 IFA$<"0" ORA$>"1"THEN430
450 IFA$="0" ORA$="1"THEN B%(P)=VAL(A$)
460 PRINTXY$
470 FORK=0TO7
475 IFP< KTHEN490
480 PRINTB%(K);
490 NEXTK
500 NEXT P
510 FORT=0TO7
520 READB
530 B%(T)=B%(T)*B
540 C=C+B%(T)
550 NEXTT
560 RESTORE
570 DATA128,64,32,16,8,4,2,1
580 PRINT: PRINT" TO DEC = ";C,
590 PRINT":TO HEX = ";HEX$(C)
595 IFBY%=1THEN660
600 PRINT"{down} *TIMES* BY HIGH BYTE? -Y/N?"
610 GETKEYA$
615 IFA$="N"THEN750
620 IFA$="Y"THEN630
625 GOTO610
630PRINTA$:BY%=1:PRINT"{down}{down}{down}{down}{down}{down}{down}{down}{down}{down}{down}":PRINT" LOW BYTE ":C1=C:C=0
640 FORJ=0TO7:PRINTB%(J);:NEXTJ:PRINT"{up}"
650 GOTO400
660 BY%=0:C2=C1+256*C :REM *BUG FIX.'EASY PROG'ING: COMMODORE16, STUART, I' ET-AL*
665 REM *PAGE 51*
670 PRINT"TOTAL: ";
671 PRINT"DEC ";C2;
675 PRINT" :HEX ";HEX$(C2)
680 C=0:C1=0:C2=0
690 PRINT"{grey3} PRESS KEY{black}"
700 GETKEYB$
710 D=0:C2=0:C1=0:C=0:FORT=0TO8:B%(T)=0:BY%=0:NEXTT
750 D=0:C2=0:C1=0:C=0:FORT=0TO8:B%(T)=0:BY%=0:NEXTT
760 GOTO240
1000 SCNCLR:GOTO400
5000 END
5010 RESUME30
5500 REM *FROM DEC TO BIN*
5510 PRINT" TO BINARY"
5512 IFD>255ANDCF>-1THEN CF%=1:GOTO5650
5515 IFD>255ANDX>1THENPRINT"MANY VARIATIONS{down}":GOTO5650
5520 IFD=>128THENBT%(7)=1:D=D-128
5530 IFD=>64THENBT%(6)=1:D=D-64
5540 IFD=>32THENBT%(5)=1:D=D-32
5550 IFD=>16THENBT%(4)=1:D=D-16
5560 IFD=> 8THENBT%(3)=1:D=D-8
5570 IFD=> 4THENBT%(2)=1:D=D-4
5580 IFD=> 2THENBT%(1)=1:D=D-2
5590 IFD= 1THENBT%(0)=1:D=D-1
5600 IFD<>0THEN PRINTD;" THERE HAS BEEN AN ERROR!"
5605 PRINT"{lred} 7 6 5 4 3 2 1 0 "
5606 PRINT"{orange}128 64 32 16 8 4 2 1{black}"
5610 FORK=7TO0STEP-1
5620 PRINTBT%(K);
5630 NEXT K
5640 PRINT
5650 FORKB=0TO7
5660 BT%(KB)=0
5670 NEXTKB
6000 RETURN
8000 REM *LIST LENGTH*
8010 PRINT" LIST FROM:"
8020 INPUT"START";M1
8030 IFM1<1 ANDM1>65535THEN8020
8040 INPUT"FINISH";M2
8050 IFM2<M1 OR M2>65535THEN8040
9000 REM *LIST 0 TO 65535 ..."PFFF!" *
9010 SCNCLR:PRINTCC$;"{down}"
9020 PRINT"MAX. LIST: 0 - 65535";" (";M1;"TO";M2;")"
9030 PRINTSPC(20);" (SWITCH = ";
9040 IFS1%=0THENPRINT"OFF";
9050 IFS1%=1THENPRINT"ON";
9060 PRINT")":PRINT:PRINTCHR$(27)+CHR$(84)
9070 FORCF=M1TOM2
9080 GETA$
9090 IFA$="P"THENGOSUB12000
9100 IFA$="Q"THEN30
9110 D=CF:PRINT"DEC=";CF;" ";
9120 GOSUB330
9130 REM IFCF%=0THEN9900
9190 REM *POSS. VARIATIONS IN DECIMAL*
9200 PRINT" *POSS. VARIATIONS IN DECIMAL*"
9210 FORL=0TO255
9220 FORL1=0TO255
9225 GETA$
9226 IF A$<>""THENGOSUB12000
9230 C0=L*L1
9240 IFC0=CFTHENPRINTL;"*";L1;"=";C0,CF
9250 NEXTL1
9260 NEXTL
9270 IFX=1ANDS1%=1THEN240
9900 NEXT CF
9910 CF%=0:GOSUB12000
9920 GOTO30
12000 REM *PAUSE ON*
12005 PRINT"PAUSED";L;L1;
12010 GETKEYA$
12020 IFA$="Q"ORA$="E"THEN30
12030 IFA$="P"ORA$=" "THENPRINTCHR$(27)+CHR$(68);:RETURN
12040 GOTO12010
```

1,000 KUNG-FU MANIACS

brand new release from the mind of game design master Alf Yngve is now available for the Commodore 64. **1,000 Kung-Fu Maniacs** is a sideways scrolling action game featuring multiple levels of kick-ass Ninja ACTION! Take control of the **Master of Awesome** on his quest to find a stolen toy robot in the Ninja riddled city of Blapsville. Fight your way through various locations including the Blapsville Ninja Country Club, Lawyer Street in downtown Blapsville, Clown Street - where you will find the dreaded Mime University ('sworn to evil'), McTerror Restaurant (do you want fries with that?), Miller Street (watch out for Ninjas hiding in the shadows) and TV Street (where Pox Studios is located) - and prepare yourself for an epic Boss showdown at the end of the game!

** GAME STORY **

You've heard of Karate, Kickboxing, Judo, Ultimate Fighting, Soccer Fan. But few have heard of the **ART OF AWESOME** ... Or its supreme champion, the **MASTER OF AWESOME**. We could go into the details of awesome, and how the master gained his rank by defeating the **LORD OF THE BLAND** in an epic showdown ... But frankly there is not enough time for all that. So let's get on with the plot. Ahem!

One sunny day, the Master of Awesome was on a walkabout, in his quest for the world's most dangerous banana peel, when he came upon a crying child ... *"Waaah!! The mean boys from Blapsville stole my toy robot!! Can you please, pleease help me get it back mister!"* And the Master of Awesome, being devoted to helping the weak and downtrodden, said to the kid: *"Don't be sad, kid, I'll bring back your toy, even if I have to search the entire city to find it. What's your name by the way?"* *"I'm Melvin. Melvin Podznofsky."* And so, the Master of Awesome walks into the city of Blapsville to find Melvin's toy robot.

BLAPSVILLE, POP. 10,001, home of the World's highest per-capita concentration of Ninjas and evil martial art schools, about 94%. Undeterred by the "VAGRANTS ARE SHOT ON SIGHT" signpost, the Master of Awesome crossed the city limits and entered Blapsville county. He had barely finished asking a group of Ninjas: *"Excuse me, have you by any chance seen a toy ro ..."* Then he got attacked. Thus began a day of violence, pain, misery, suffering and of course, destruction which makes a smashing computer game ... Or does it?

** GAME CREDITS **

Game design, GFX + SFX by Alf 'Master of Awesome' Yngve.Music by Jon 'Master of Audio' Wells.Enhancements by Richard 'Master of Magic' Bayliss.Packaging by Jason 'Master of the Universe' Mackenzie.

** GAME FEATURES **

Awesome weapon power-ups!
Multiple levels to battle through!
Fantastic soundtracks by Jon Wells!
Detailed background graphics!
Wacky Ninja battle action FUN!
Mega-Boss Battle!

** NOW AVAILABLE AS A FREE DOWNLOAD THE COMMODORE 64! **
(Psytronik tape and disk versions available SOON!)

CONTROLS
Use Joystick in PORT 2 only.

Controlling the Master of Awesome as he fights his way through the city of Blapsville, you have the following attacks and moves:
Default left-facing punch: Just hit FIRE.
Right-facing punch: Press FIRE while moving right. Good timing is required or just hammer the fire button.
Jump or skip: Move UP.
Dropkick: Move UP and LEFT or RIGHT and press FIRE.
Crouch Kick: Move DOWN.
Downward left kick: Move DOWN and LEFT and press FIRE.
Downward right kick: Move DOWN and RIGHT and press FIRE.
Upward punch: Move UP and press FIRE.

BONUS ITEMS
For every 10,000 points scored, you earn an extra life. You also gain one of the following bonus items until a life has been lost:

1. AWESOME SHURIKEN - Can be thrown in any direction.
2. FEEBLE SHURIKEN - Can be thrown left or right.
3. PRICELESS MING VASE - You skim huge vases to knock your enemy over.
4. TAI-CHI HEARTBURN - A giant fireball will give the enemies a good toasting.
5. THE WORLD'S MOST DANGEROUS BANANA PEEL: The thrown peel slides very slowly but trips all enemies in its path.

CREDITS
Game design, gfx + sfx by ALF 'Master of Awesome' YNGVE.
Music by Jon 'Master of Audio' Wells.
Enhancements by Richard 'Master of Magic' Bayliss.
Packaging by Jason 'Master of the Universe' Mackenzie.
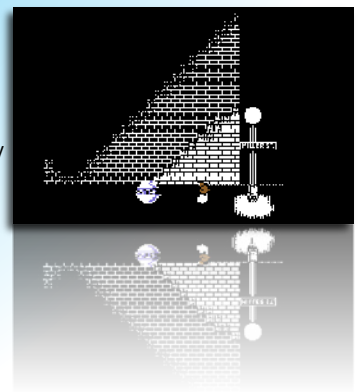
Issue 47
January 2011

Editor
Nigel Parker

Spell Checking
Peter Badrick

Text & HTML Conversion
Paul Davis

D64 Disk Image
Al Jackson

PDF Design
Nigel Parker

Website
www.commodorefree.com
Email Address
commodorefree@commodorefree.com

Submissions
Articles are always wanted for the magazine. Contact
us for details .We can't pay you for your efforts but you are safe in the knowledge that
you have passed on details that will interest other Commodore enthusiasts.