The Number One Online Magazine for the:





AMSTRAD CPC / PLUS PCW PC NC & OTHERS

CPC OXYGEN Yearbook 2004

Computer Warrior!

Patrick Furlong looks back at the games featured from the epic comic story from Eagle.

The First Issues!

We look back on the first 10 issues of CPC Oxygen, and forward to it future plans.

Portable CPC?

The Amstrad NC100, is it a CPC and is it useful in today's world.

More!

150 pages of content, too much to list!

CONTENTS

Opening Pages

Review Zone

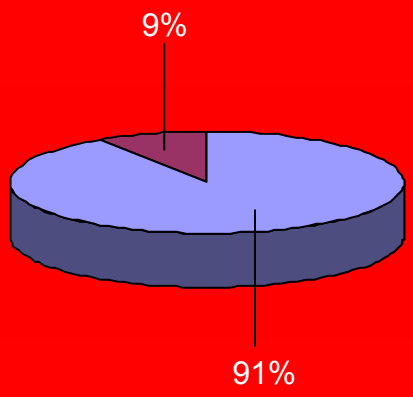
Technical

Action Zone

Your Vote!

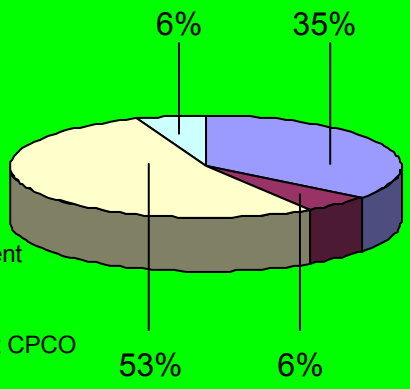
For the last few issues of CPCO we being asking for your vote on certain topics. Here are the results:

What do you think of the recent PDF version of the Firmware Guide?



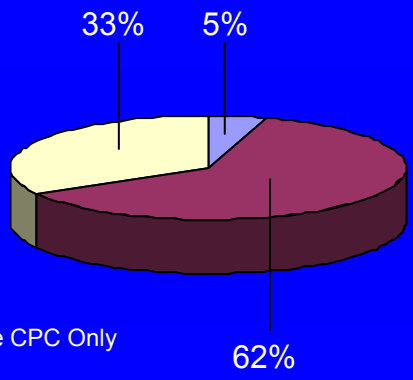
- Yes it's Great
- I Hate PDF

Will you be purchasing the CPC Yearbook CD-R when it's available?



- Yes, it going to be great
- Depends on price/content
- No, not at all
- Yes, but only to support CPCO

How much time should CPCO dedicate to other Amstrad but non-CPC articles?



- Equal amount of time
- One or two articles a month
- CPCO should be for the CPC Only

Words from The Editor!

This publication has had many delays (especially while doing the last 10 percent) due to many reasons which has resulted in the end of the title changing from 2003 to 2004, even more worrying for me, the pic to the left now seems way out of date. But after all the delays it's finally here and that what counts.

Those of you who are flicking through the paper edition will notice many articles from our biggest contributor, Patrick Furlong, a big thanks must go to him, partly for his patience in the slowness of this yearbook but mostly for his excellent articles, thanks Patrick. Also not forgetting the writers of articles written for the online edition of CPCO, republished here for all the see, on and offline. Thanks also to Sean McManus for the "B for Basic" article.

I want to give this award……. Wait a sec, wrong speech. I'll just finish up by saying that while we had our slow patch there are still many CPC (and others) articles that I can think of that haven't yet been written and I'm intend to write them, so do many of you reading this, I hope :-)

Oh almost forgot, you can find us at http://cpcoxygen.digi-alt.net and if you wish to email me, you can at john@digi-alt.net.

John Kavanagh

CD VERSION

At the time of going to press (always wanted to say that) the CD-ROM version of the CPC Oxygen Yearbook isn't yet finished, however it is near completion with only a few small things needing ironing out.

The CD-ROM will contain this publication in a number of formats, including an "easy on the printer" version. As well as the yearbook itself, there are also over 2000 disk images of great CPC games, a screen saver and so much more. More details and time of availability will be given on our website in the near future.

http://cpcoxygen.digi-alt.net

Amstrad Visions

While Amstrad is known as the producer of cheap n' cheerful electronics it still manages to produce a number of highly innovative products, although not as much as it did in the 80's and early 90's. Here we look at a number of its innovative computers from that great time period.

The Amstrad CPC

By far the most popular Amstrad by the readers of CPCO. It started life back in 1984 as the 464 with the 664 and 6128 following a year later. It was designed to compete against the already established Sinclair Spectrum and Commodore 64.

Despite been technically superior in every way to the Spectrum and overall better than the standard C64 it still only played third place in overall world sales (over two million units sold) of 8 bit micros. Unlike other 8-bit computers at the time, the CPC was marketed for both business and entertainment use.

In 1990 Amstrad produced the PLUS range along with the GX-4000 game console which flopped badly. While the PLUS didn't do as well as was hoped, it still managed to shift a modest number of units until it was quietly discontinued sometime in 1992 with remaining stock sold in many stores at bargain prices.

Commercial support for the CPC ended in the early 90's but even to this day it is still supported by a army of loyal fans. This is evident by this yearbook and the many active websites which are totally devoted to the computer.

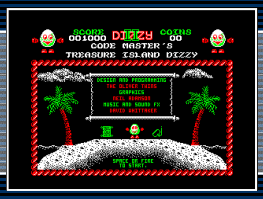
The Amstrad PCW

The Amstrad PCW was a dedicated word processor that could do a bit more, most notably the ability to use the CP/M operating system. First launched in 1985 as the 8256 with 256 KB RAM, followed by the 8512 with 512 KB. A more professional looking version came later in the form of the 9512, followed by the 9512+. What made

The CPC Range

Amstrad CPC 464

Year:	1984
CPU / Speed:	Zilog Z80 running at 4 MHz
RAM:	64 KB (16 KB used for screen memory)
ROM:	32 KB
Graphics Details:	16 KB screen memory, using 3 graphic modes which are 160 x 200 x 16 colours, 320 x 200 x 4 colours and 640 x 200 x 2 colours with text screen matrix of 20x25, 40x25 and 80x25 depending on screen mode. Colours used are chosen from any combination of 27 colours.
Sound:	3 channels, 8 octaves + 1 noise channel
Drives:	Cassette Datacorder
Operating System:	AMSDOS and CP/M (when used with a optional disc drive)
Interesting Facts:	The first computer from Amstrad and it came into direct competition with the Sinclair Spectrum and Commodore 64 which were both released two years earlier.



Nintendo has Mario, Sega has Sonic and the CPC has it own super hero as well in the shape of a egg! The Dizzy games were first created on the Amstrad CPC by the Oliver Twins in 1987 which lead to many sequels in adventures and arcade style games.

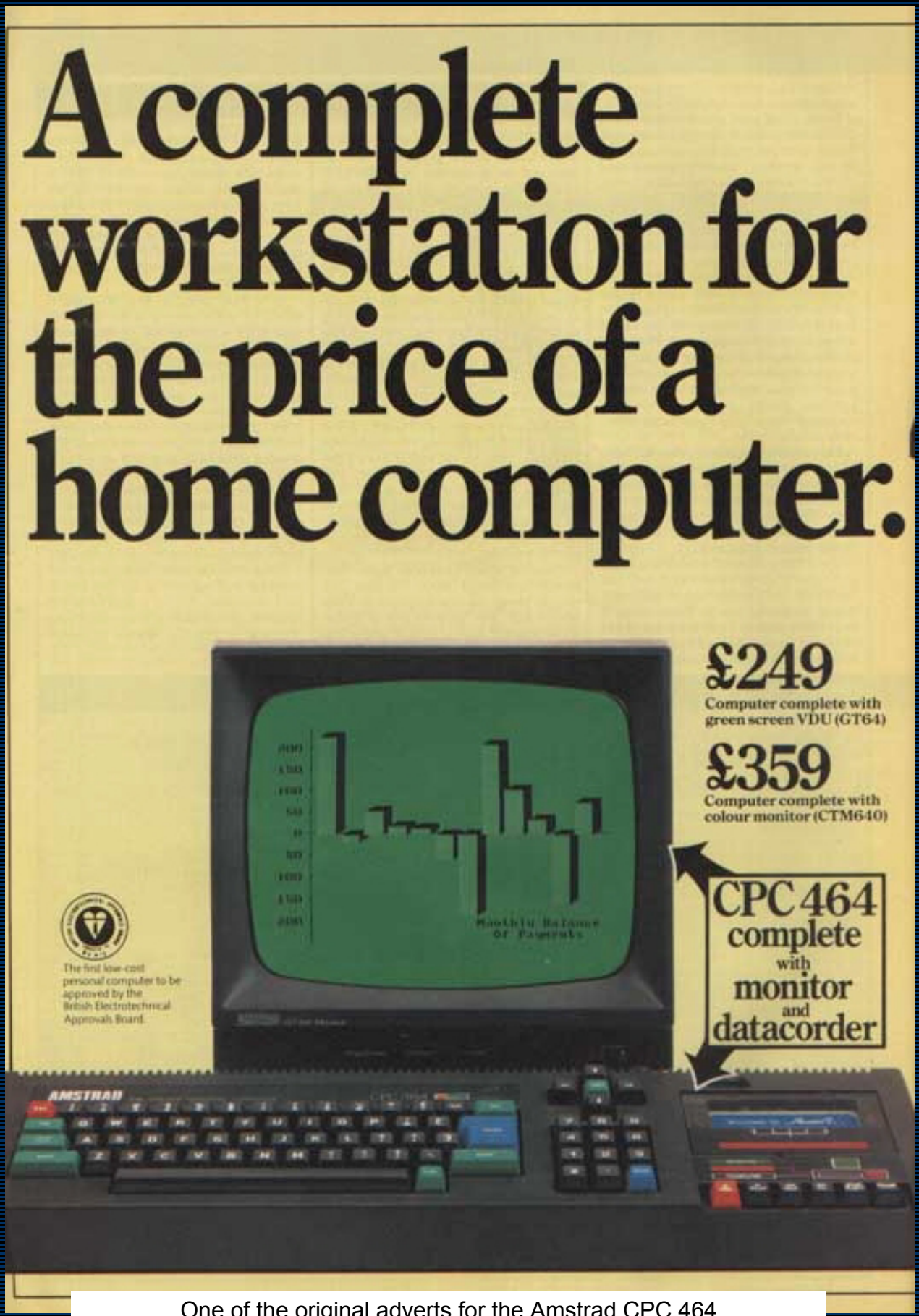
DIZZY

Amstrad CPC 664

Year:	1985
CPU / Speed:	Zilog Z80 running at 4 MHz
RAM:	64 KB
ROM:	32 KB
Graphics Details:	16 KB screen memory, using 3 graphic modes which are 160 x 200 x 16 colours, 320 x 200 x 4 colours and 640 x 200 x 2 colours with text screen matrix of 20x25, 40x25 and 80x25 depending on screen mode. Colours used are chosen from any combination of 27 colours.
Sound:	3 channels, 8 octaves + 1 noise channel
Drives:	3 Inch Disc Drive
Operating System:	AMSDOS and CP/M
Interesting Facts:	The 664 is the rarest CPC being in production for 6 months before been replaced by the CPC 6128. Apart form the disc drive, another advantage the 664 had over the 464 was version 1.1 of basic which had a number of extra commands.

Amstrad CPC 6128

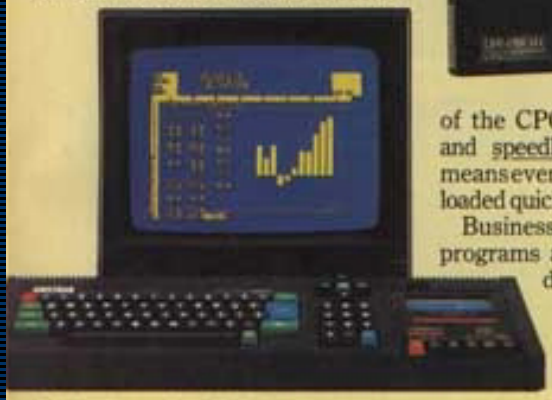
Year:	1985
CPU / Speed:	Zilog Z80 running at 4 MHz
RAM:	128 KB
ROM:	32 KB
Graphics Details:	16 KB screen memory, using 3 graphic modes which are 160 x 200 x 16 colours, 320 x 200 x 4 colours and 640 x 200 x 2 colours with text screen matrix of 20x25, 40x25 and 80x25 depending on screen mode. Colours used are chosen from any combination of 27 colours.
Sound:	3 channels, 8 octaves + 1 noise channel
Drives:	3 Inch Disc Drive
Operating System:	AMSDOS and CP/M
Interesting Facts:	There is no support for hardware sprites in the any of the original CPCs so they had to be programmed the hard way. However, the CRTC chip allows for amazing effects, some of which was thought to be impossible on 8-bit computers.



One of the original adverts for the Amstrad CPC 464

It's mouthwatering.

64K of RAM, 32K of ROM, a high resolution green screen VDU, integral cassette data recorder, typewriter style keyboard, numeric keypad and a very fast extended BASIC. All for £249. (The CPC464 is also available with a colour monitor instead of the green screen VDU for £359 complete).



CPC464 colour monitor (CTM640)

You'd be hard pressed to find a comparable computer at that price let alone the monitor and recorder.

And the CPC464 comes complete and ready-to-go. Just plug it in.

64K RAM (42K available).

The low cost but powerful CPC464 is equally at home in business and educational applications as it is running the household budget or playing games.

With 42K RAM available to BASIC, the opportunities for sophisticated and complex programming are considerable.

80 column text display.

The green screen VDU is purpose designed with a bright, crisp, 80 column text display that compares favourably with systems costing several times as much.

You can program up to 8 text windows and there's a graphics window, too.

The CPC464 has a typewriter style keyboard, large ENTRY key, sensibly positioned cursor keys, numeric keypad for fast data entry and a full 8-bit character set.

If you think that sounds impressive, wait until you hear the 3-voice, 7-octave stereo output through a hi-fi amplifier and speakers.

Amsoft. High quality software.

A wide range of programs is already available and we're expanding it rapidly. The software takes full advantage

of the CPC464's high specification and speedloading capability. Which means even complex programs can be loaded quickly.

Business applications, educational programs and arcade games are all designed to make maximum use of CPC464's impressive graphics, stereo sound and processing abilities.

Amstrad. User Club.

Members enjoy immediate benefits like the privilege card, Club binder, regular magazine, competitions for valuable prizes and contact with other Amstrad users.

Whether you're interested in serious commercial applications or simply a games fanatic, you'll want to join the Club.

CPC464. Unlimited scope for expansion.

We've thought of everything you're likely to need in the future. That's why there's a built-in parallel printer interface. A low cost optional disk drive system including CP/M* (with the option to access 3000 programs) and LOGO. Joystick port. And the virtually unlimited potential of the Z80 data bus with sideways ROM support.

Finally, a power supply and modulator (MP-1) allows you to connect your CPC464 green screen VDU system to a colour TV.

Optional disk drive DDI-1 including interface CP/M* and LOGO £199.95

Optional 80 column dot matrix printer DMP-1 operates at up to 50 characters per second. Combined with the CPC464, it offers a high performance text processing system for only £199.95.

BOOTS COMET Dixons
Menzies RUMBELOWS AND OTHER COMPUTER STORES

AMSTRAD

ONE GREAT IDEA AFTER ANOTHER

*Trade mark Digital Research

I'd like to know more about the new CPC464 complete computer system.
Please send literature right away.

NAME

ADDRESS

POSTCODE

To: Amstrad Consumer Electronics plc, Brentwood House, 169 King's Road, Brentwood, Essex CM14 4EF. Tel: Brentwood (0277) 228888.

the PCW so successful was its relative ease of use (at the time), especially its setup which was considered simplistic with only a power cable from the back of the computer and a connection from the monitor to the keyboard.

The last PCW was the PCW-16 in 1994, 16 standing for 16 MHz, not 16 bits which it could have been easily mistaken for with and GUI (Graphical User Interface). Support for CP/M seem to have been dropped in an attempt to make the PCW-16 to appear more of a word-processor and less of an computer. However it was possible with a bit of technical knowledge to set it up with CP/M to give it even greater flexibility than older slower PCWs.

The PC Compatibles

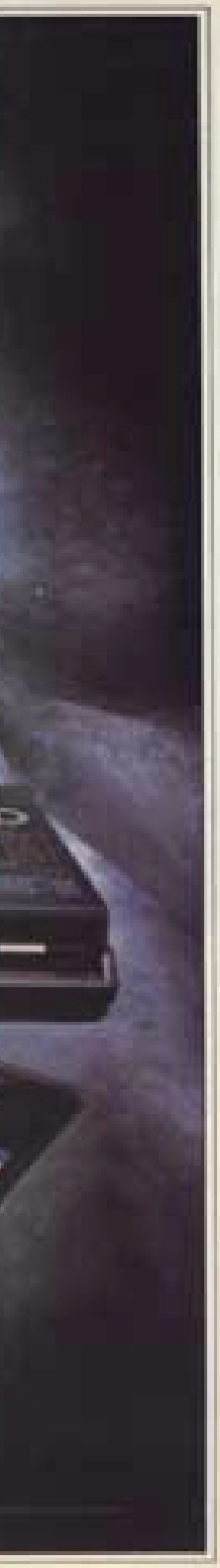
Amstrad greatest profits came from the PC market having been the first company to produce a sub Sterling £400 IBM compatible computer. That computer was the PC1512 (more info on next page) followed by the 1640 and later the 2386 series. Not only were they cheaper than other IBM clones but were also a lot more compatible to the IBM PC which was amazing considering the non standard way Amstrad build they PC. Things such as having one power supply for monitor and base unit smacks at cost cutting but it worked well in those systems. What was really interesting was that the system surpassed the expensive IBM with Amstrad own custom graphic modes along with other enhancements.

Amstrad continued making PC up to the late 90's but having lost their competitive edge to cheaper PC makers they soon pulled out of the computer business. Another factor to take into consideration was the lack of innovation in later models that made Amstrad computers more 'ordinary'.

END

Advert for the short lived Amstrad CPC 664

DISC DRIVE

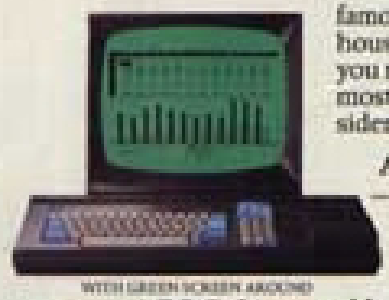


ENZIES · RUMBELOWS

THE LOW COST COMPUTER FOR HOME AND BUSINESS ·

WITH COLOUR MONITOR AROUND

· £449 ·



WITH GREEN SCREEN AROUND

· £339 ·

THE HOME COMPUTER THAT MEANS BUSINESS.

If you know anything about computers you'll know that disc drives are up to fifty times faster than cassette when you're loading and saving programs. In fact, a disc drive makes computing faster, more reliable, more efficient and more fun. But up till now the only way to gain these advantages for a home computer was to buy a separate disc drive attachment. Now Amstrad are pleased to announce the first complete home computer with built-in disc drive: The Amstrad CPC 664.

And when you buy a CPC 664 you'll find it's not just the disc drive that's built-in.

You'll get everything you need, including a monitor (green screen or full colour). We'll even give you a free CPM and Logo disc, so all you do is plug in and you're in business.

BUSINESS OR · PLEASURE ·

Although a disc drive will make games more fun (and there are loads of them to choose from) it also makes the CPC 664 a serious proposition for the business user.

There are accounting, word-processing, spread-sheet and data-base programs (to name but a few).

Amsoft Business Control, is a complete suite of programs for integrated sales invoicing, stock control and sales ledger for around £99. (Requires an additional FD-1 disc drive around £199 and DK-2 cable around £7).

The CPC 664 is also supplied with CP/M* to help make your business more efficient and effective by providing access to the famous range of CP/M* software.

Wordprocessing with Amsword can improve the productivity of everyone from unskilled typist to trained secretary. Around £25.95.

HIGH PERFORMANCE · LOW COST ·

The one thing you won't need a computer to work out is that the Amstrad CPC 664 represents outstanding value for money.

You only have to check the cost of buying all the elements separately (64K computer, disc-drive, monitor) to realise that the Amstrad package is very hard to beat.

With a green screen monitor the cost is just £339. With a full colour screen it costs £449. And after you've saved money on the price of the computer itself, you go on saving on the price of software.

There are hundreds of programs for business or pleasure available on disc (and cassette) to CPC 664 users. Many from Amsoft, others from other famous-name software houses. Few will cost you more than £49 and most will cost you considerably less.

AN EXPANDING · SYSTEM ·

There is a complete range of peripherals available to CPC 664 users which plug directly into the built-in interfaces.

These include a joystick, additional disc drive (to double your on-line storage) and the Amstrad DMP-1 dot-matrix printer. (There's also a cassette interface so that you can use CPC 464 programs on tape). And there are many more peripherals from Amstrad and other manufacturers which can be used to enhance the CPC 664.

AMSTRAD USER CLUB ·

Join the optional Amstrad User Club and we'll keep you informed with our monthly user magazine, and information on all software as it is introduced. Your membership details will be recorded on your personal club card, which entitles members to various privileges and offers.

Figure analysis made easy with Microspread. An easy to use spreadsheet with pull-down menus and a wide range of mathematical options. Around £40.

Please send me more information

Name

Address

Amstrad CPC 664

Amstrad, P.O. Box 462, Brentwood, Essex CM14 4EE

SPECTRUM · WH SMITH · WIGFALLS · AND GOOD INDEPENDENT COMPUTER STORES ·

CP/M stands for Control Program for Microcomputers and was programmed by Gary Kildall who set up Digital Research. It's regarded as one of the first micro-computer operating systems. Similar in use to DOS only more complicated and requiring less resources and only a 8-bit processor. In fact DOS was based on CP/M source code.

RFT / VEB Microelektronik Mühlhausen (DDR) KC Compact (Not an Amstrad Product)

Year:	1989
CPU / Speed:	UA 880 D (Z80), 8 Bit running at 4MHz
RAM:	64 KB
ROM:	32 KB
Graphics Details:	16 KB screen memory, using 3 graphic modes which are 160 x 200 x 16 colours, 320 x 200 x 4 colours and 640 x 200 x 2 colours with text screen matrix of 20x25, 40x25 and 80x25 depending on screen mode. Colours used are chosen from any combination of 27 colours.
Sound:	3 channels, 8 octaves + 1 noise channel
Drives:	Cassette Datacorder (external)
OS / Software:	KC-Compact V1.3, Basic 1.1
Interesting Facts:	The KC is featured here for been a clone of the Amstrad CPC, it is not manufactured by Amstrad. Made in 1989 in what was East Germany, production stopped a year later when the Berlin wall fell. Unlike the CPC, the KC can output video to a PAL TV as well as a RGB monitor.

Amstrad 464 PLUS

Year:	1990
CPU / Speed:	Zilog Z80 running at 4MHz
RAM:	64 KB
ROM:	128 KB
Graphics Details:	Software: 16 KB screen memory, using 3 graphic modes which are 160 x 200 x 16 colours, 320 x 200 x 4 colours and 640 x 200 x 2 colours with text screen matrix of 20x25, 40x25 and 80x25 depending on screen mode. Colours used are chosen from any combination of 27 colours. Cartridge Games: 16 sprites with 32 colours on screen from a palette of 4096.
Sound:	3 channels, 8 octaves + 1 noise channel
Drives:	Cartridge and Datacorder
Operating System:	AMSDOS and CP/M
Interesting Facts:	While the original CPC was marketed as a home and business tool, the PLUS range was marketed totally as a games machine to the horror of many serious users of the CPC. Also, the PLUS was nowhere as successful as the original CPCs.

There were only a handful of GX-4000/PLUS cartridges games, most of which were released by Ocean. Examples are Switchblade, Robocop 2, Wild Streets, Pro Tennis Tour, Plotting, Pang, Operation Thunderbolt, Bo Exit, Navy Seals, Klax, EPYX World of Sport, Crazy Cars II, Copter 271 and Batman the Movie. All of which are available on the CPC Oxygen yearbook CDROM.

Amstrad 6128 PLUS

Year:	1990
CPU / Speed:	Zilog Z80 running at 4 MHz
RAM:	128 KB
ROM:	128 KB
Graphics Details:	Software: 16 KB screen memory, using 3 graphic modes which are 160 x 200 x 16 colours, 320 x 200 x 4 colours and 640 x 200 x 2 colours with text screen matrix of 20x25, 40x25 and 80x25 depending on screen mode. Colours used are chosen from any combination of 27 colours. Cartridge Games: 16 sprites with 32 colours on screen from a palette of 4096.
Sound:	3 channels, 8 octaves + 1 noise channel
Drives:	Cartridge and 3 Inch Disc Drive
Operating System:	AMSDOS and CP/M
Interesting Facts:	Not really a interesting fact, more of a sad one when Amstrad decided not having any facilities for the connection of a datacorder. Although it was possible to add one if you were handy with a soldering iron.

Amstrad GX-4000 Game Console

Year:	1990
CPU / Speed:	Zilog Z80 running at 4MHz
RAM:	64 KB
ROM:	N/A
Graphics Details:	Cartridge Games: 16 sprites with 32 colours on screen from a palette of 4096.
Sound:	3 channels, 8 octaves + 1 noise channel
Drives:	Cartridge
Operating System:	N/A
Interesting Facts:	It flunked, badly. It was more powerful then the Sega Master System and Nintendo Entertainment System but released at a time when everybody wanted the powerful 16 bit Sega Megadrive!

The PCW Range

Amstrad PCW 8256

Year:	1985
CPU / Speed:	Zilog Z80 running at 4MHz
RAM:	256 KB
ROM:	48 KB
Graphics Details:	720 by 256 mono (90 by 32 characters)
Sound:	Beeper
Drives:	3 Inch Disc Drive
Operating System:	CP/M plus Mallard Basic build in
Interesting Facts:	The PCW was sold with an Amstrad printer and was not compatible with other printers without extra add-ons.

Amstrad PCW 8512

Year:	1986
CPU / Speed:	Zilog Z80 running at 4MHz
RAM:	512 KB
ROM:	48 KB
Graphics Details:	720 by 256 mono (90 by 32 characters)
Sound:	Beeper
Drives:	Two x 3 Inch Disc Drive
Operating System:	CP/M plus Mallard Basic build in
Interesting Facts:	The PCW 8256 / 8512 was not considered to have non-business styling while lead to the design of the PCW 9512.

The PCW range made computerised word processing affordable back in 1985 when before that, similar systems cost hundreds more. The PCW is similar to the CPC range in many respects suggesting that Amstrad build the PCW on recent experience of their first computer range.

Amstrad PCW 9512

Year:	1997
CPU / Speed:	Zilog Z80 running at 4MHz
RAM:	512 KB
ROM:	48KB
Graphics Details:	720 x 256 mono (90 x 35 characters)
Sound:	Beeper
Drives:	3 inch disk drive
Operating System:	CP/M
Interesting Facts:	The 9512 was replaced by the 9512+ which was the same computer apart from having a standard 3.5 inch drive.

Amstrad PCW 16

Year:	1994
CPU / Speed:	Zilog Z80 running at 16MHz
RAM:	1MB (2MB upgradeable)
ROM:	1MB
Graphics Details:	640 x 480 mono (no text mode since the OS is GUI based)
Sound:	Beeper
Drives:	3.5 inch disk drive
Operating System:	Roseanne and CP/M 2.2
Interesting Facts:	It was the last PCW. It was also the only PCW to have a GUI.

Amstrad made a large number of IBM compatible PCs. Here we list those that either had the most impact or that are unique in ways that sets them apart from other PCs of the period.

IBM Compatibles

Amstrad PC 1512

Year:	1986
CPU / Speed:	Intel 8086 running at 8MHz
RAM:	512 KB (upgradeable to 640 KB)
ROM:	16 KB
Graphics Details:	320 x 200 / 640 x 200 in 4 colours (CGA) and 640 x 200 in 16 colours. Text modes were 40 x 25 and 80 x 25.
Sound:	beeper
Drives:	5.25” Drive (1 or 2)
Operating System:	DOS
Interesting Facts:	The 1512 was twice as fast as the original IBM PC and selling for Sterling £399 at the time of launch it was up to four times cheaper then other established market leaders helping Amstrad to capture 25 percent of the European computer market. It was also more compatible than other IBM clones of the time period, despite been a very different computer in looks as well as technical design.

For more information on the Amstrad PC 1512 and PC 1640, check out www.amstrad1512.co.uk.

Amstrad PC 2386

Year:	1988
CPU / Speed:	Intel 80386 running at 20MHz
RAM:	4MB
ROM:	Unknown
Graphics Details:	MDA, CGA, Hercules, EGA, MCGA, VGA and enhanced VGA.
Sound:	Beeper
Drives:	3.5 inch disk drive (1.4 MB) and Hard Disk (usually 65 MB)
Operating System:	MS-DOS 4.00 and Microsoft Windows/386
Interesting Facts:	Amstrad often offer non standard graphics modes which were superior to the IBM standard which didn’t affect compatibility. However software which used the enhanced mode could only work on Amstrad PCs.

"The procesoor is an 80386 running at 20MHz; except during floppy disk I/O and the following 3 seconds, when it is 5MHz." - Page 3-32 of the Amstrad PC2386 User Manual. Amstrad Manuals during the 80's had a reputation for been hard to follow as the example above proves, along with spelling errors and all.

Amstrad Mega PC

Year:	1993
CPU / Speed:	Intel 386 SX running at 25MHz
RAM:	1MB
ROM:	64KB
Graphics Details:	VGA
Sound:	8-bit
Drives:	3.5" Disk Drive (1.44 MB)
Operating System:	DOS
Interesting Facts:	What made the Amstrad Mega PC unique is that it had a build in Sega Megadrive. The Mega PC (like other Amstrad PCs) was available with higher specs such Intel 486 processor, bigger RAM and Hard Drive.

Amstrad Integra

Year:	1996
CPU / Speed:	Pentium 100 MHz
RAM:	16MB
ROM:	N/A
Graphics Details:	SVGA
Sound:	16 bit
Drives:	3.5" Drive (1.44 MB) and Quad Speed CD-ROM drive, Hard Drive
Operating System:	Windows 95
Interesting Facts:	**NOTE: The above specs are for the top of the range model.** Forget Microsoft Media Centre, Amstrad did it years ago with the Integra, TV with internet, Video Capture, Video CD player, Video / Audio input, Fax and even an Answering machine. It could be argued that this was the last innovative Amstrad PC. By specs alone it seemed expensive at the time which resulted in it short market life.

Article Taken from April 2002 Edition of CPC Oxygen

Eighteen years! That's a hell of a statement. After all this time, CPCs are still working like there's no tomorrow, users are still programming them and generally having fun either by using them for small pet projects or to playing classic games saying to themselves, "They don't make it like that anymore!". Lets take a step or 18 back to where it all began.

In 1984 nobody took Amstrad seriously when they announced the CPC464. After all Amstrad was a small company that produced cheap and not so cheerful Hi-fi players. For the most part, the people were right as Amstrad had major headaches in designing the CPC. I remember reading somewhere about the almost comical antics that people at Amstrad had in trying to built a working computer. But for the life of me I can't remember a) where I'd read it and b) a few funny lines from the article, but I did remember thinking lot of it can't be true. Well anyway, back to this article.

Suffice to say that Amstrad, after a bit of employee shuffling managed to get a computer out, and to the surprise of everybody, quite a good computer too! It was of course, the Amstrad CPC 464.

It was launched on the 11th of April 1984 with a good selection of games and was bundled with either a colour or green tube monitor. It seems like a proper computer so the buying public thought, and they were right. Complete with it own monitor, built in datacorder and 64K of memory along with a really good, if strangely laid out keyboard (but at least it was QWERTY based which is more then said about other computers of the time period). Of course the computer sold like hot cakes.

Beyond 464

By 1985, the CPC had a good chunk of the market along with all the software you ever needed, almost. Magazines about the CPC started to appear on shelves (see right) and the most interesting topic in them were about the upcoming 664, a disc based CPC.

The 664 was basically a 464 with a 3 inch disk drive instead of a cassette datacorder. I never really understood why Amstrad went for a 3" drive instead of the standard 3.5" which would have surely boosted sales, although that was not the reason why there were not many 664's sold. The main reason is that just a few months after the launch of the 664, Amstrad confirmed that the 664 would be replaced with the 6128, carrying 128K of memory and a new restyled keyboard.

THE CPC 6128 HAD MORE MEMORY AND A DIFFERENT STYLE THAN THE 664

The 464 and 6128 continued to sell well, grabbing a big share of the European market. Software companies continue to produce video games, education titles as well as application / utility software.

The Five Stars

Magazines dedicated to the CPC must have a big part to play in getting people excited into purchasing those computers. From those glossy front covers, classic works of art along with bold statements make even those who are only slightly interested in computers to pull the magazines off the stand and actually get them interested in computers with a never-ending passion. The Five Stars in order of appearance is as follows:

Amstrad Computer User

November 1984 to May 1992

Computing with the Amstrad

January 1985 to December 1988

Amstrad Action

September 1985 to June 1995

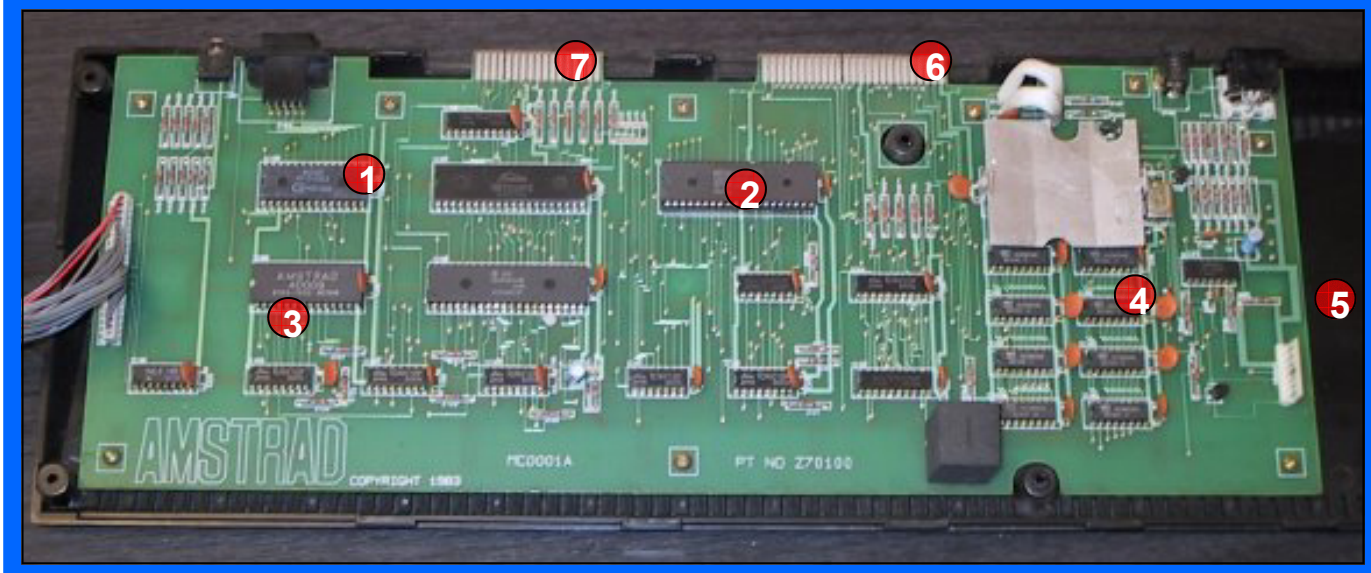
Amtix

October 1985 to April 1987

CPC Attack

June 1992 to November 1992

Technical Specifications of the CPC 464



1 Sound Chip

The CPC uses the AY8910 from General Instruments for sound output. Producing 3 sound channels + white noise.

2 The Processor

The 8-bit Zilog Z80A running at 4 MHz is the heart of the CPC. refer to past issues of CPCO for more information.

3 ROM

A 32K Rom supplies BASIC 1.0 and other computer functions.

RAM

4

The 464 has 64k (65536 bytes) of memory, 16k of that was used for the video display.

5 Datacorder

The CPC datacorder did have an advantage over units in other computers, been built-in and generally more reliable.

6 Expansion Port

The Expansion port in the CPC is very flexible, allowing up to 240 extra ROMS, extra RAM and other weird and wonderful devices.

Printer Connector

7

A general downer, CPC owners had to wait until 1990 for an 8-bit printer port, the original 464 only had a 7-bit port!

Graphics

8

640 x 200 x 2 Colours
320 x 200 x 4 colours
160 x 200 x 16 colours
Considered to be very good in 1984. The CPC was more flexible in terms of colour then other 8-bit computers of the time period, although it didn't had any hardware sprites for easy game development, that is until the Plus arrived in 1990.

Although the CPC was a success, it never competed with the sales of the cheaper Spectrum and Commodore 64/128 computers. Amstrad wanted to make even more money in the computer industry, they got the taste for success and was willing to bite the bullet to get even more.

The 'Other' Amstrads

In 1985 and 1986 alone, apart from the CPC, Amstrad produced the PC's (which were cheap but good clones of the IBM PC) and the PCW range. The PCW was very much like the CPC, sharing the same processor but with larger memory and better on screen text functions but with poorer graphic functions. So in other words, it was useless at games, but then again it was marketed as a word processor.

Although many other people who fell in love with the computer thought so much more of it, it had a real operating system so if a person wanted to, it could function like a proper computer (If using CP/M is a proper computer).

Thanks!

Our thanks to **Nicholas Campbell** for the use of his scanned magazine covers. Check out his web site for more cover scans at: http://users.durge.org/~nich/cpcmags/

FROM 1986 ONWARDS AMSTRAD SHOWED MORE INTEREST IN PC's THEN THE CPC. PC's LIKE THE 1986 AMSTRAD 1640SD (ABOVE) COMPETED WITH IBM RANGE OF PC's AND IN MOST CASES, WON.

18th BIRTHDAY

The PCW 8256 was launched in 1985 and the Amstrad PC, the 1512 in 1986, followed closely with the 1640 by which time Amstrad took a 25 percent control of the European PC market.

All of this left the CPC in the cold, with rumours going around that the CPC range was to be dropped. By this time Amstrad PLC had purchased Sinclair and started making weird spectrums which made the CPC world even more worried. But Amstrad hung onto the CPC up to 1990 when they released a new range of CPC's, the PLUS range

The PLUS

Amstrad introduced the plus range to fight off tough competition from the 16-bit computers of the time, mostly the Commodore Amiga. The Amiga was wildly becoming a hit in Europe, at the loss of the CPC and other 8-bit computers.

Amstrad answer was to make a slightly better 8-bit CPC! Everybody reading this probably know this was one of the worst errors Amstrad ever made and put the CPC into a spiral end to doom.

There was nothing amazingly wrong with the PLUS range, in fact there were fine machines (see box out), that is, fine machines using 8-bit processors. The world have moved on, and while 16-bit computers like the Atari ST did not make much of an impression against the CPC, the Amiga did. The Amiga was already a good computer and it would have taken another 16 bit based machine to compete with it.

Amstrad had limited success with the Plus range, no where as close as it did with the original CPC but they did shift a few units, mostly to followers of the CPC and families wanting a cheap computer for the kids.

The GX4000 game console was a complete flop, a powerful 8-bit console, way better than the NES and Sega Master System but then again, Sega was about to go 16-bit with the Mega Drive. Producing a few more games might have helped as well. I'll quote Alan Sugar (CEO of Amstrad) to show you to train of thought that helped the plus range failed before it began, effectively killing the CPC:

THE AMSTRAD NC-100, THE CLOSEST THING TO A PROTABLE CPC, THAT IS, UNTIL THE NC-200.

"The end user doesn't know whether it is 16-bit, 8-bit, or if it is working with gas or steam or with elastic bands!"

Alan Sugar, 1990

The Dark Years

By 1992 the CPC was considered long dead, the original CPC were been sold in car boot sales, the GX-4000 long forgotten by the software industry leaving only a few titles to play with. The Plus range was no longer been made.

Looking back it easy to see how short sighted Amstrad PLC were, but they weren't the only ones who believe the Plus would be successful, CPC followers did, including me. Even as a kid I still remember thinking that it may fail because of the Amiga, but also thinking that the Amiga at the same price of the CPC did not come with a monitor and I knew how awful it was to use a computer with a TV set for any sort of serious programming / work. Apparently a lot of people did not see it this way and went for the Amiga anyway. Amstrad seemed not to care much as it was still having success with it range of PC's.

Amstrad also focused it's attention to little notepad type computers like the NC-100 in 1992, which was the nearest thing to a portable CPC, at least that how Amstrad Action saw it.

Speaking of magazines, Amstrad Computer User, Amstrad's official magazine stopped printing May of that year. From the emptiness came CPC Attack. It was bright, new, colourful and a little childish in areas. But I liked it, it had a certain charm and did not go too far overboard on the childish side of things (or so I thought been a young teenager at the time).

In 1993, the only thing CPC you could get in any type of store was Amstrad Action as CPC Attack lived a short life of only 6 months.

No software to buy, no hardware, no add-ons. The death of the CPC seemed certain, if it was not already considered dead.

More Dark Years

With mainstream CPC magazines no longer in existence, a fanzine called WACCI kept the CPC world alive. By now, only hardcore enthusiasts and techies had interest, with neither large companies like Amstrad nor the general public wanting to know. WACCI continued to supply enthusiasts with wonderful articles…..

PC Total Control

1995, Microsoft released Windows 95 and helped springboard the PC to the general public, the type of public who would have never dreamed of owning a computer but who were by now purchasing PCs like any other consumer device (almost).

By now, the Amiga was pretty much dead in the water and Commodore was no more after the disaster which was the CD32 game console. Apple was having a hard time keeping its head above water, this was a time when Apple saw everything white and boxy, and not a curved transparent blue!

THE COMMODORE CD32, COST MILLIONS OF DOLLARS TO DEVELOP AND HELPED DROVE COMMODORE INTO BANKRUPTCY.

The CPC Plus Range

The Plus range had many improvement over the original CPC, the biggest been the built in game console with the following enhancements:

- 16 Hardware Sprites at 16 x 16 pixels, made up of 16 colours from a palette of 4096. Impressive when you take into consideration that each sprite palette is independent of the screen mode and can be magnified to double or quadruple size.
- Smooth hardware scrolling both horizontally and vertically. The original CPC only had limited vertical scroll.
- Cartridges can hold up to 128K for game data.

AMSTRAD GX-4000 GAME CONSOLE. KOOL LOOKING BUT A TOTAL FLOP.

Other improvements included analogue as well as digital joystick connectors and standard expansion sockets instead of edge connectors.
The printer port in the Plus is 8-bit instead of 7-bit.

THE AMSTRAD 6128 PLUS - ARGUABLY THE MOST BEAUTIFUL CPC IN TERMS OF DESIGN, ALTHOUGH LACK OF A CASSETTE RECORDER CONNECTION TURNED OLD 6128 OWNERS OFF.

However Amstrad did not put a connection for a cassette recorder in the 6128 Plus which was a big downer among CPC owners. All the Plus range used Basic 1.1 (except the GX-400 of course), the same as the old 6128's, having the year "1985" plastered all over the screen on start up can't have helped sales either.

Light @ the end of the tunnel!

The internet the way we know it today was becoming quite popular at this time. Web sites for anything imaginable were springing up all over the place and yes you guessed it, web sites devoted to the lovable CPC were popping up everywhere and best of all, former CPC users were typing "Amstrad CPC" into search engines. Amstrad had released the CPC ROM's for general use a year earlier and as a result it encouraged programmers to produce CPC emulators. The most popular at the time been CPCEMU, although work on this had begun as far back as 1991 as a Z80 emulator.

For the first time, the CPC seem to be making a come back, at least for the hardcore CPC fans. Ex-CPC users can now surf to their favourite CPC website, download an emulator plus a few games from a list of hundreds, games that cost money a few years earlier. It was a time covered in nostalgia, a time when CPC users felt that they lost something in the cold off-white PC world.

Beginning of a new Era

This nicely brings us up to the present, and the CPC is on the birth of a new generation with the upcoming CPC-NG. I'm sure you know a lot about this new baby and if you don't, then check the past issues of CPC Oxygen or better yet, go to the CPC-NG website at: www.cpcng.com

The Start of the Retro Wars

The CPC is not the only retro computer to be brought right into the 21st century, a Commodore remake is on the way and there's already a new Spectrum in the name of Sprinter, very much like old times so?

Both the Sprinter and Commodore One are similar in that both can act like retro 8-bit computers, after that the two computers are very different altogether.

The Sprinter is manufactured in Russia by Petersplus at prices starting from $299 while the 'One' is been developed in America and expected to cost around the $200 mark. The Sprinters runs at 21 MHz and can be slowed to 3.5 MHz through hardware while the Commodore One runs at 20MHz and can only be slowed with throttling software so the first point goes to the Russians.

The Sprinter

Information

Manufacture:	Peters Plus - Sprinter
Website:	www.petersplus.com
Price:	Starts at $115 US for the Sprinter SP2000 mainboard and goes up to $170 for the Sprinter.

Specifications

CPU:	Z84C15, running at 21Mhz or 3.5 Mhz, depending on mode.
Memory:	4MB, upgradable to 64MB
OS ROM:	256 KB
Video RAM:	256 KB
Screen Resolution:	320x256 256 colors, 640x256 16 colors
Video Output:	TV or CGA monitor
Sound:	16 bit Stereo

Although it is unfair the rate these computers as we don't have full specs, the Commodore One seems to be better in everything else.

First off there's the graphics of the 'One' with a max resolution of 1280 x 1024, this is coming into CPCNG territory, but of course as far as everything else is concerned, the CPCNG just blows both of them away (again a bit like old times). The Sprinter has improved since the Spectrum days but not by much with a resolution of 640 x 256 x 16 been mustard up with the tiny 256 KB of video memory.

Another cool feature of the Commodore One is that the OS will be stored on Compact Flash cards, bringing 512 MB of fairly fast OS access to the end user. The only downside been the current price of Flash Memory.

So far the Sprinter has not made no real impressions of the Spectrum community while the upcoming Commodore may not be such a easy push over. The reason is simple, while not anything close to the CPCNG, it does has one thing going for it, loyalty, the Commodore 64 was the best selling model of any home computer ever, in world terms, well beyond the CPC's user base.

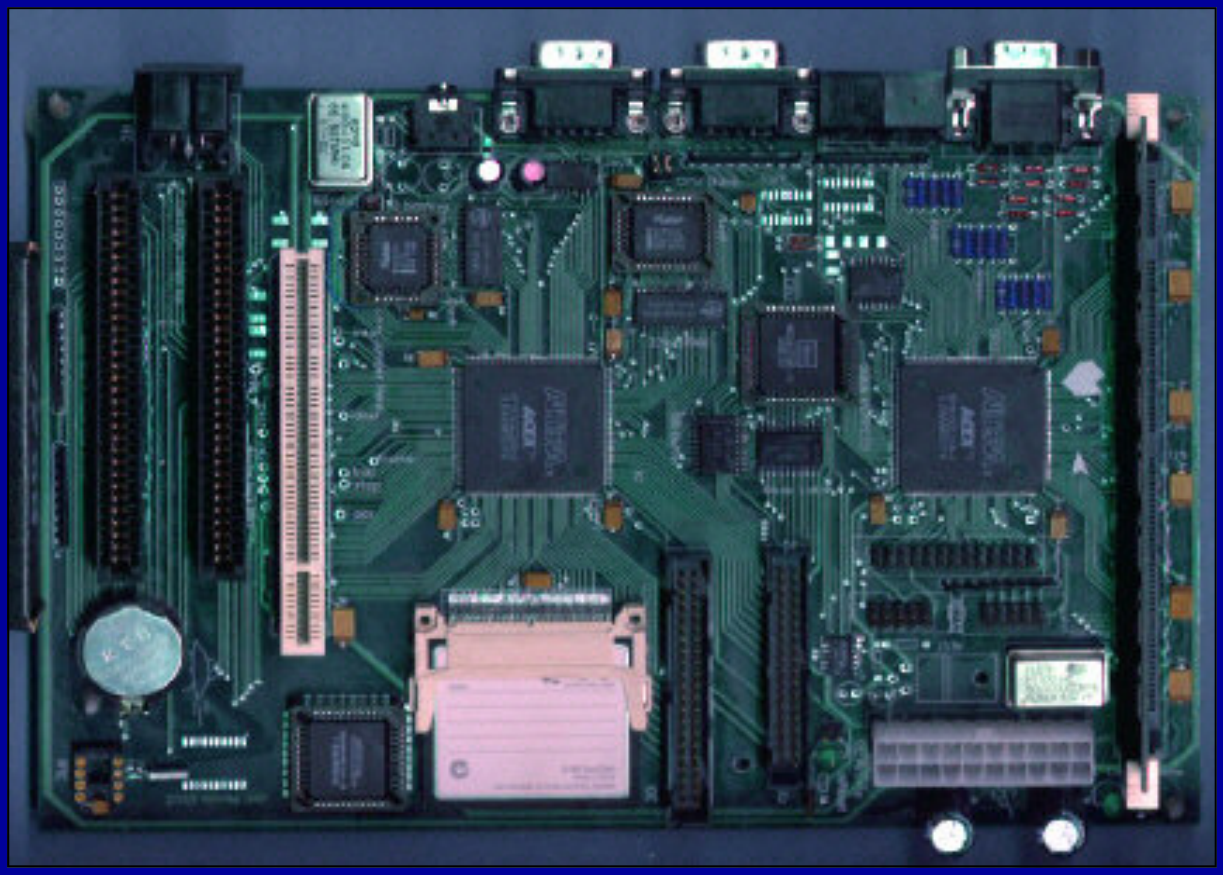
There is a Chinese saying, not that I'm Chinese, but it goes like this, "May you live in interesting times.", I think that about to happen.

The end of the end

Before I go, I leave you with another thought, in researching this 2 issue article I came across an old issue of Amstrad Action dated 1994 where it talks about the first 10 years of the CPC and the last paragraph was titled "The end of the end". I'm calling this "the end of the end" also because you can bet anything that somebody eight years from now will be writing about the history of the CPC. Goodnight.

END

The Commodore One



Specifications

CPU:	C1, 65c816. Fully compatible with the 6502 (the CPU of the Commodore 64). Capable of 24-bit address memory.
Speed:	20 MHz (approx) Software throttle is used to slow computer to Commodore 64 speeds.
System Bus:	Runs at 50Mhz (unusually faster then the CPU)
Secondary CPU:	6502 processor used for i/o support
Memory:	32 MB (hard wired) which 16MB used for RAM and the rest for the video controller.
Video RAM:	16 MB
Max Resolution:	1280 x 1024 with a choice of 256 colours out of a palette of 65,545 in regular and linear modes.
OS Storage:	Compact Flash (like the type used in digital cameras). Flash cards can go anywhere from 4MB to (currently) 512 MB, although the latter is very expensive. The Commodore One's OS will be stored on this.

NG GENERATION

"The most difficult part is to explain to people that it's not necessary to have a 1.5Ghz computer if you want to launch an office application"

Christophe Guelff

The NG Generation

Written by John Kavanagh

Over the past few months, since CPCO started in fact, the question that got asked more often then not is "What is this talk about a next generation CPC?". So without further ado, lets entertain:

The CPCNG project took off in May 2000 by Christophe Guelff after reading an article about the Zilog eZ80, a new 8-bit processor that is 100% compatible with the original Z80. From that Chris developed the idea of developing a cheap modern era CPC, namely the CPCNG (Colour Personal Computer - Next Generation).

It designed to be more of a consumer product, totally flexible, just plug in, turn on and play which is quite similar to the CPC in the 80's, only this time with modern 21st century technology. Christophe Guelff puts it perfect in a recent interview, "With CPCNG it's possible to do everything, go on the Internet, do office job, play game, do your homework if you're a student, listen to MP3, play DVD or look at a video files etc".

A very exciting statement indeed, this alone gives the impression of something very special, equalling the functions of a bulky PC and providing a more stable and cheaper alternative.

The Specifications

As I already mentioned, the heart of the machine is a Zilog eZ80 running at 50 MHz, processing 4 times more instructions then the original Z80 giving a total performance of over 200 times faster then the CPC.

This may not sound much next to a 2800 MHz Pentium 4 but unlike PC's the CPCNG will be greatly optimised to gain the maximum possible speeds from it's applications.

The CPU can access up to 16MB at a time but with the MMU (Memory Management Unit) this can be greatly increased. In fact, the CPCNG can be expanded to 1024 Megabytes which is bigger then some hard drives a few short years ago.

Graphics

The original aim was for the CPCNG to display a resolution of 1280 x 1024 in 32 bit colour, but technical problems has resulted in a final specification of 1024 x 768 in 24 bit colour. Still sounds good to me considering the fact that there's not much noticeable difference to the eye. The prototype version (see next page) will have reduced graphics to keep things simple.

The bad news is that the computer won't benefit from lower resolutions when rendering graphics, like the PS2, Game Cube etc.

Inspired by Amiga Fantasy Design - Damien Légiéde

Sound

The SPU (Sound Processing Unit) has 8 sound channels and can output sound in 5.1 surround. Along with large screen TV there's no reason why the CPCNG can't be used as a home cinema.

IDE Interface

The hard drive will be IDE based, suing the exact same standard as PC compatibles. This leads to one small problem, aren't IDE drives 16 bit?

Well... yes, and this leads to the problem, that with a 8-bit CPU, only half of the drive's capacity can be accessed (due to a 8-bit data bus). As usual, the innovation of the CPCNG team leads to a solution.

The solution is to buffer the high byte of the data word in an 8-bit latch, which is read / written to on an additional port number, problem solved.

CPC Compatibility

The CPC compatibility mode will be based on a CPC 6128 (not PLUS) and will run 90 percent of CPC software. It's difficult getting 100 percent compatibility due to some programs (mostly demos) requiring exact timing.

You can get more information about the CPC Compatibility Unit in this month's techie article, written by Christophe Guelff.

The Design

We have seen in the article a few designs on how the CPCNG may look, how close or how far they are from the finished machine remains to be seen. There have been a lot of talk about it been based on the Amiga fantasy design (top of page) which was a very popular design in it's time and is still pleasing to the eye in the 21st century.

While I imagine the "computer in a keyboard" be the most popular, the CPCNG will be released in different designs such as a tower unit, a video game console and even a DVD station just like the early 90's Philips CDI but far superior in every respect.

The CPCNG could also be made by other companies so it's quite possible to see many clone designs with various packages and options.

Final Note

We like to think that this issue of CPC Oxygen has wetted your appetite to learn even more about the CPCNG and we hoped the article has explained aspects of this wonderful machine in an easy

The Prototype

Before the finished computer there will be a prototype model which is currently been worked on. It's based on a simpler design and will be used primarily as a tool for team members to work on the finished computer.

The first noticeable limitation with be the graphics with only 128K VRAM. With this in mind the prototype can only display resolutions of:

512 x 256 in 8 bit colour
256 x 256 in 16 bit colour
1024 x 512 in only 4 colours

There will also be no audio of any kind.

The main components will be:

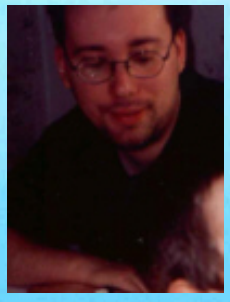
- **eZ80 CPU from Zilog**
- **128K EEPROM**
- **128K Fast SRAM for video and CPC-mode**
- **FPGA**
- **Sockets for large PC-style SDRAM cards**
- **Sockets for modem, keyboard / mouse and power**
- **Expansion sockets for new cards**

The finished prototype is expected to be finished by the end of the year, but this could slip to a later date. For more information check out Hans Summers web site at www.arnold6.com.

Could this be the look of the future? Based on the lines of the CPC 6128 but with a sleeker lowered keyboard, extra keys and a DVD drive?

The CPC-NG Team

(In no order at all! - Information correct at time of original article)



Christophe Guelff

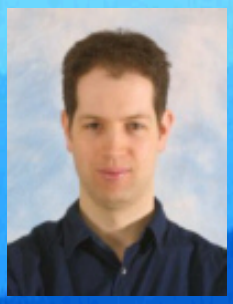
Nationality: French

GPU, SPU, Press / Communications (International), Coordination and Sponsoring.

Christophe Dupas

Nationality: French

GPU, SPU, Press / Communications (France), Software Development.



Hans Summers

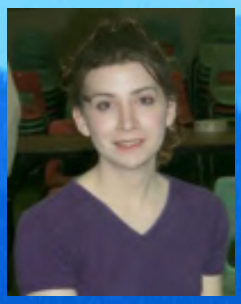
Nationality: British

FPGA and VHDL Design, Motherboard.

Jeri Elsworth

Nationality: American

Developer for the CommodoreOne.



Markus Buntru

Nationality: German

Preparation and Administration of cpcng.de, Press / Communications (Germany).

Ehong Hwangs

Nationality: American

PCB Design

Stefan Stumpferl

Nationality: German

Operating System, Assistant for cpcng.de.

Laurent Guedon

Nationality: French

Press / Communications (France) and Administrator of cpcng.com

No Photo!

Interview with Christophe Guelff

CPCO: How did the CPCNG project started and how did you got involved?

Christophe: The project starts at the end of may 2000. I decide to launch it after I've seen an article about ez80 in French press...

Christophe: But personally I was only interested in communication job, not really in hardware/software design

CPCO: Ah, I see. Why based it on the CPC? Were you a CPC user?

Christophe: Yes the CPC was my first computer and was the best I have ever seen.

Christophe: The CPC was a famous success in Europe. Our aim was to build a computer able to have success. If you build a DragonDataNG I'm not sure the success is really possible...

CPCO: I can imagine so! What your current job within CPCNG?

Christophe: Actually I work on the communication part. That means I write article (e.g. for the very good CPC Oxygen) for the professional press (in France, Hackerz Voice magazine is interested to have article about CPCNG in each issue), I contact sponsor, I work on the French translation of the website and I try, when it's possible, to give my opinions on the mailing list...

CPCO: That what I like hearing! Are you in contact with computer manufactures at the moment?

Christophe: Not for the moment. We need a full running computer before do that. The only contact we can have are actually financial contact..

CPCO: Is it difficult gaining financial support for the project?

Christophe: We start this job actually only. You know the most difficult is to explain to people that it's not necessary to have a 1.5Ghz computer if you want to launch an office application... It's difficult to explain too that it's possible to have fun with a computer (by programming for example). But I think next month we 'll be able to announce a good news about financial help for the project...

CPCO: Is office applications be the CPCNG primary use or will it be good at everything?

Christophe: With CPCNG it's possible to do everything : go on the Internet (the ez80 has network facilities), do office job, play game, do your homework if you're student, listen MP3, play DVD or look at a video file (AVI...) etc.

CPCO: So it equals the PC in flexibility? Will it have any advantage over the PC? Perhaps faster OS loading?

Christophe: But there will be a lots of advantages over the PC? I explain:

The CPCNG will be easy to program (with our NGBasic), CPCNG will have a stable hardware (no update, no card to add), CPCNG will be easy to use, CPCNG will be more stable than a PC under Windows (!), CPCNG will be faster than a PC (on CPCNG the OS and every program will be optimized), CPCNG will have a more exciting look'n'feel than a PC. If you want CPCNG is the computer for all the family. It's a complete solution, easy to use: you connect, it runs... It do what you want without problem. And everybody have the same computer (it's different with PCs)

CPCO: There be less need for software drivers then the PC?

Christophe: A good question... for e.g. the Plug'n'play is, in my opinion, a good concept but when you see how Microsoft include it in Windows, it's really crap. So we can imagine a PnP implementation for CPCNG. Actually Damien Legieda and me we are thinking about the possibility to have generic and global driver in standard for the CPCNG.

CPCO: I was hoping you say something like that! Above when you mentioned a more exciting look and feel, did you meant the OS or the computer looks?

Christophe: For the OS, you'll have two interface : a shell with NGbasic (like Amsdos on classic CPC) or a complete GUI. We are actually in discussion about the GUI concept.
For the computer's look, I think we need something compact like a computer-in-a-keyboard. That's my opinion. But this is, for the moment, an unofficial point of view... I know there is people (like Christophe Dupas, one of our best architecture designer) who think about the possibility to have several models of CPCNG (one per year for e.g.) : a computer-keyboard, a desktop, a CPCNG in tower (with expansion slot etc).

CPCO: Will the schematics be available for people to build the computer themselfs? Even if a major company decided to produce the CPCNG?

Christophe: Yes.

CPCO: How much electronics experience would a person need to build the computer? Could a person who only did basic soldering build it?

Christophe: Hum... I'm not sure the most important is the experience in electronics, but the money you have... Our aim is to purpose the most explicit documentation. Our schematic are easy to understand, we will purpose each source code etc. So a person with a big motivation is able to do it.

CPCO: Will it be expensive to build so?

Christophe: For a single person I think (I think Hans Summers is the best person for this answer). I think about the FPGA design price especially. For a company interested in the CPCNG, it's different.

CPCO: I'll be sure to ask Hans..... two more questions.... What can we expect over the coming months from the CPCNG project in the line of emulators, early prototype boards and software?

Christophe: An emulator is currently under development. The author Chris Dupas will try to finish it as soon as possible. As soon as the emulator is ready it's possible to start to write the first parts of the OS.

The FPGA is currently under development. I think it's possible to resolve this problem for the end of the summer.

If we work seriously I think it's possible to have first prototype board for the end of the year. For the GPU and SPU, they'll be developed by French student and the deadline is April 2003.

CPCO: Finally the last question, possibly the hardest or perhaps easier of all: What your dream for the CPCNG?

Christophe: I'm not able to see the future but what I hope is that the CPCNG community grows up and that the final CPCNG is ready for September - December 2003.

Christophe: And you ??

CPCO: Ok, now I'm the interviewee!!!

My dream is for the CPCNG to become the computer to beat the PC and to live a life where I don't have to curse Microsoft every time my PC crashes!

Christophe: It's the dream of every CPCNGineers ... :-)

END OF INTERVIEW

10 things we hate about the PC

10: They always require that games to be installed to the hard drive before playing, even if there's no real technical reason to do so.

9: Once Installed, PC's still request the CD-ROM. It had to believe but this is considered a anti-piracy tool by many software developers.

8: PCs look like a large off white brick, and that's been kind!

7: Most PC's act like one too! No matter how fast they get, you still end up waiting for things to happen, either due to software been less optimised each year or/and to the fact that Microsoft has released a new version of Windows.

6: The sexy looking PCs are always under powered, or if they are powerful n' sexy, they cost too much!

5: They crash, often. Ok, under Linux they don't, but Linux is not yet an operating system suitable for the average person.

4: They can be noisy at the best of times. Sitting hear on the PC I can hear the hard drive, the fan and my DVD drive and it all sounds like a ill washing machine.

3: And cause headaches the best of times, if you're lucky!

2: They totally un-optimised in every way. For example, accessing a floppy disks slows the computer to a halt, especially if the computer has trouble reading the disk.

1: Privacy, or to be more exact, the lack of it coming soon. See the “The year in review” article for more info.

To understand way. In the end, the machine acceptance depends on you, the public at large.

Will the world turn their noses up at the idea of a 8-bit? They didn't to the X-box been only a 32 bit compared to the PS2 and Game Cube. Bits has become less important then what they used to be and in the end people will judge the computer by visual and sound aspects.

When they see amazing games and fast applications on a computer that boots almost instantly, can play DVD's in 5.1 surround sound, all packaged in a nice design. Well then, how could anybody resist it.
For more information on future developments, check back with later issues of CPC Oxygen or visit the websites below. Also, if you have any specific questions, the CPCNG team will be happy to answer then in a future Q&A page. Send your questions to CPC Oxygen at cpcoxygen@hotmail.com

Coming Soon!

We only had a chance for one interview from the CPCNG team this month, we plan to interview more over the coming months. CPC Oxygen will be also support the CPCNG project by attempting to get popular British and Irish computer magazines interested in publishing articles about the project, thus creating extra public interest. Bye for now!

CONTACTS

www.cpcng.de

The Main CPCNG site, hosted in English, German and French.

www.arnold6.com

Hans Summers site for the prototype CPC.

www.cpcoxygen.pro.ie

CPC Oxygen web site where you can find articles about CPCNG dating back to the beginning of 2002.

CPC-NG Questions & Answers

Welcome to the Q&A page for the CPC-NG project where Christophe Guelf answer your questions.

What kind of OS support is planned for the CPC NG? Obviously AMSDOS in the compatibility mode, but for the non-compatibility mode, what can we expect? FreeBSD, Linux, FreeDOS?

Sigurd Urdahl

Hi Sigurd,

We are working on a new OS which takes ideas from two popular OS's called FutureOS (www.futureos.de), an OS for the CPC written by Stefan Stumpferl, and OS-X (a Z180 OS written by Francis Görmacker). The OS is currently under work and it will be like an Unix OS but more simple to use (you will use a great GUI), with a shell that will use NGBasic. Of course, CPCNG will be able to use other OS's as well. As you can see on linux.org, in the "hardware project" section, we have ask the Linux community to received help to port the famous OS onto CPCNG. One of our members, Belinda, is interested to port CP/M on the eZ80 and the CPCNG. There will be a specific version of FutureOS for CPCNG too. The problem is that it's necessary to see how the eZ80 run. Alls OS can't work on it.

Apparently you want to add two R4000 (32/64bits proc running at 50Mhz) on the board of the CPCNG. It's really exciting because the R4000 was in the...Dreamcast ! So, we can consider that in CPCNG, there will be two small Dreamcast, one for the video job, another for the sound job ? But do you think it's enough to have better games than PS, Nuon, PS2 or GC ? Thanks for your answer by advance and good luck for the future.

Syntax Error/Amiga coder

Hi SyntaxError,

I'm not sure that the Dreamcast used a R4000. Perhaps a SH... ? But the PSOne use a R3000 and the R4000 is used by SGI computers (like the "Indigo" station). The R4000 is a really powerful chip which will be able, for the 3D, to calculate polygons information. After the rest of the GPU we are designing, we will work on textures etc. Of course, it's not possible to have the same capacities than the latest X-Box (using an N-Vidia chip) or GC. But we hope to have graphics (2D or 3D) equal or better than the PSOne or the Nuon. After it's a problem of programming and you know... CPC programmers are the best for incredible things... ;)

Will there be an emulator or some sort of development package coming out before the computer so that people can get started on developing software for it? Web Browsers, Office Applications, Games etc.

John Kavanagh

Hi John,

Of course it's necessary to purpose to developers several tools to prepare softs, games before the official launch of

Where's the Techie Stuff?

Hungry for the technical stuff and more besides? Then check out CPC Oxygen online magazine at www.cpcoxygen.pro.ie where you find the latest news, opinions and articles on this wonderful future creation.

Here is a list of the online articles in order of issue:

Issue 2

General Introduction to the CPCNG

Issue 3

CPCNG sockets, GPU and SPU

Issue 4

The IDE interface and Memory Mapping System

The CPCNG. We are actually working on a eZ80/CPCNG emulator (a first release is available on our mailing list). The problem is that this emulator will never be optimized (it takes a long time to debug it etc.). So I think the best choice is to purpose to developers a kind of "CPCNG for developers" : a simple motherboard able to be connected on a PC display, using a PC keyboard with only developer's tools (Ngbasic and others programming languages) and a lots of doc (how run the video part, the sound part, the OS etc).

If you had to guess, when would you think the CPCNG first prototype be available to the public?

CPCO

What do you mean with "first prototype" ? The first prototype is "very simple" and do not include the advanced (and exciting) SPU and GPU. You'll find a presentation of it at www.arnold6.com (or www.hanssummers.com/computers/cpcng). For this prototype, I think as soon as the FPGA design is ready, it will be possible to work faster and I think it's possible (but not sure) to have them for the end of the year.... During the time of development of the FPGA, we are working on preparing the port of existing OS for the first prototype (futureos, www.futureos.de , and OS-X). If for you "first prototype" means first motherboard of the final CPCNG, I'm not able to give a specific date cos we are working during our free time and there is a lots of job to (for eg. the final advanced GPU and SPU will be ready only in april 2003...)...

What will the in-built programming language be like? will it be similar to Amstrad Basic or perhaps similar to Blitz 3D?

CPCO

We are thinking about a language like the Locomotive Basic - localized by using a system of tokens - with exciting possibility like include ASM code or launch external commands like RSX. To do 3D, I think it will be possible to have specific command. It's not really a problem...

Would the CPCNG be easy to change, for example, if a person wanted to make an in car MP3 player or a home DVD player with the CPCNG. Would something like that be easy to do and would the CPCNG be able to handle that?

CPCO

It would be possible to connect CPCNG to a TV so in theory play DVD on it would be possible. But to do this, we need an exciting and powerfull GPU. That's the reason why Damien Legieda and Chris Dupas purposed to add a powerfull Risc able to decode the DVD (for eg.).For an in car MP3 player... why not ? Schematics are free and you have the right to modify them but I don't see really the interest. Of course with the advanced SPU, playing MP3 will be very simple for CPCNG. I think the DVD player is more natural than the simple MP3 player... no ?

Issue 5

Second parts of the Memory Mapping System and the IDE Interface.

Issue 6

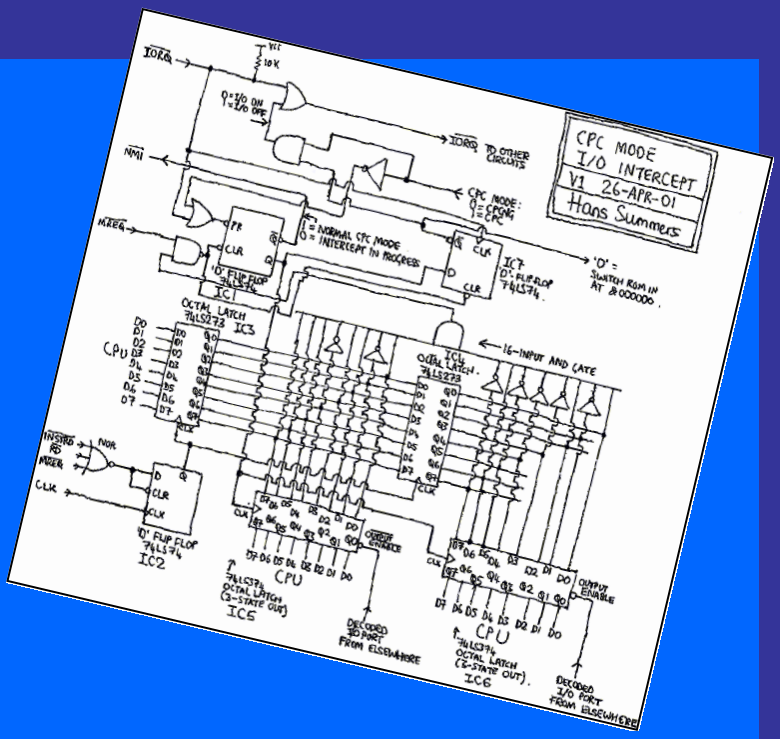
More details on the GPU, the Graphics Processing Unit

Issue 7

Details on the CRTC (Cathode Ray Tube Controller)

Issue 8

The CPC Compatibility Unit.



The Zilog eZ80 CPU

The eZ80 is probably one of the most interesting development of 8-bit Microprocessors in recent years.

While the eZ80 is being marketed as embedded webserver (see zilog's website at www.zilog.com), there's a far more interesting role to play as far as the CPC is concerned. The eZ80 is one hundred percent compatible with the original Z80 which as you know powers the CPC. This compatibility has let to the development of the CPC-NG computer.

THE EZ80 CPC - 50 TIMES FASTER THEN THE CPC'S Z80A AND THE MAIN PROCESSOR BEHIND THE CPCNG PROJECT

Before we jump ahead of ourselves let me explain some of the features on the eZ80.

Embebbed webserver

To understand what an embedded webserver is you got to first understand traditional embedded devices. Not going into too much detail, traditional embedded devices communicates to a master computer using communications mechanisms that are both software and hardware proprietary. The master computer has a user interface that is tied to the embedded application so to modify the embedded devices requires reprogramming the system which can be quite costly. An embedded webserver contains dynamic web pages that control communications to and from an embedded device by TCP/IP. This as you can imagine reduces cost and simplify software installation and maintenance.

This is the basic idea of what an embedded webserver is. While this in the future could be beneficial to the CPC, this is not the main reason of why it is of a big interest in the CPC world for the last few months.

CPC New Generation

The big interest is that the eZ80 is 100 percent compatible with the original Z80 processor that was used in the CPC. What is ever most exciting is that work has begun on a new CPC that is exactly 50 times faster at processing data then any CPC that went before it. Check out http://www.cpcng.free.fr for more information on this amazing new computer.

The eZ80 operates at 50 MHz and can process 4 times more data per cycle then the Z80. So basically its similar to a Z80 running at 200MHz which is impressive by anyone standards considering how cheap the chip will be.

Other Advancements

As most techies know, the Z80A could only access 64KB at a time so computers like the CPC 6128 had to use software to switch between different banks of memory. In other words, while the 6128 had 128KB of memory as standard, the CPC could only see 64KB at a time. With the eZ80 its similar except that it now can access 16,384 KB (16MB) at a time and can even access more through the CPC build in Memory Management Unit (MMU), now that's impressive

It'll be interesting to see how this mighty 8-bit CPU gets on in a 32 / 64 bit world.

END

eZ80 Webserver

- 50 Mhz processor
- Multiple and accumulate engine
- 16 MB linear addressing and single cycle
- 3.3V / 5V tolerant2 DMA channels
- Universal ZiLOG Interface (selectable UART, IIC, SPI)
- 6 PRTs with prescalers
- 8KB SRAM
- 32-bit GPIO with interrupt support
- On-chip oscillator
- Optimised pipeline architecture
- External Flash support

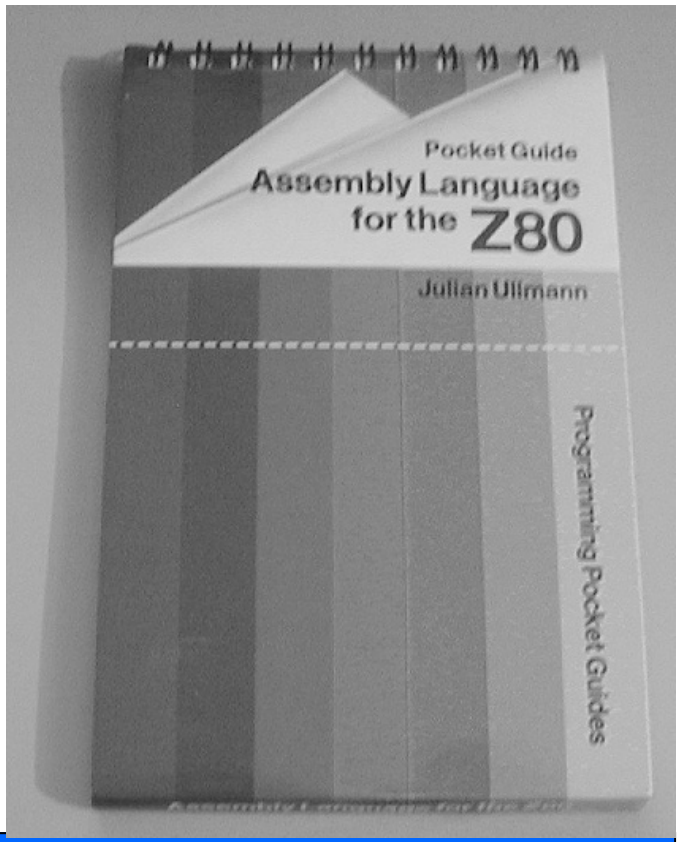

Blast from the past!!!
Get all your CPC information here
CPC/GX/Amstrad Pictures
Quiz
Technical Information
Member Area
Always Updated CPC News
Forum
Masive Music Rips Section
Big 3500+ Download Area
AMSTRAD.DK

Review Zone

Hardware

Software

Other Stuff



NC100 REVIEW

Amstrad NC

Undeniably cute comes to mind as John Kavanagh checks out the Amstrad NC-100 Notepad as he wonders about it usefulness in today's world.

On first impressions it's hard not to notice the beautiful design of the Notepad with its sharp lines and short travel keyboard along with little details that complement the computer. Originally launch in 1992 with a price tag of £199 in the UK it was targeted mainly for people wishing to write on the move along with a few other handy features.

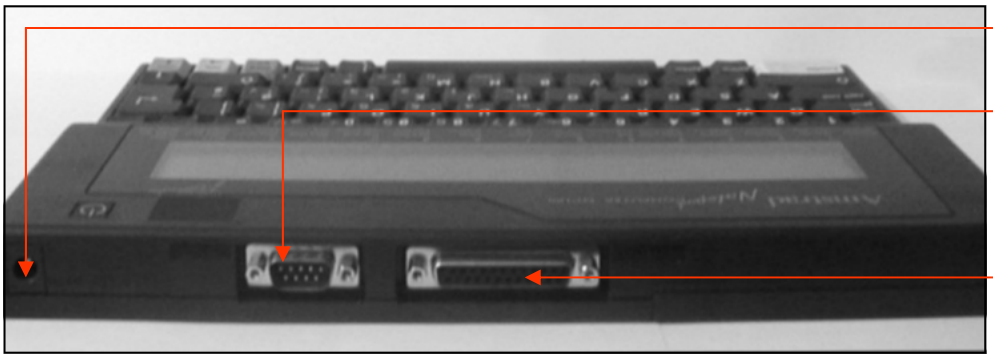
What a good job it does too, the keyboard, what can I say except that it the best keyboard I've ever used. The original review in Amstrad Action (Issue 87 December 1992) didn't overstated the facts when they said it was 'just about the best keyboard we've used'. So good that after a few minutes of typing with it and you'll be punching up words faster then Sean Connery in "Saving Forester". The only downfall is that it seems a bit noisy, less than the CPC or most PC keyboards but still way too high for use in a quite place like a library.

The build in word processor may seem a bit limited at first to many people who are used to modern day applications, but being based on Protext, the best word processor for the CPC it is a joy to use if given half a chance, more on that soon.

SIZE COMPARISION OF THE NC100 WITH THE CPC PLUS

100 Notepad

Specs of the NC



Speaker
6 Volts DC input.

Serial Port
Standard serial port, my favourite method for transferring files back and forth with a PC using a null modem cable. Although this is possible with the parallel port, the serial don't need any custom hardware.

Parallel Port
A standard 8-bit printer port, compatible with parallel printers of the time period, which mostly means dot-matrix. Don't expect modern printers to work.

Speaker
Surprisingly loud for such a tiny speaker. Beeper is only good for a few tones.

On/Off Button
May come in handy! Not really an off button as such, more of a standby mode. So it's useless in the event of a computer crash.

Screen
Good contrast for a 1992 computer, however it doesn't contains a backlight.

Reference Template
In keeping with Amstrad tradition of reference templates such as those on the 464 (not so useful) and the 6128 (useful), this is a must have. The yellow text means press the yellow key followed by the number for that function, while white text means press Control followed by number.

Keyboard
Standard QWERTY short travel keyboard, works really well, a joy to use in fact.

Year:	1992
CPU / Speed:	Zilog Z80 running at 6MHz
RAM:	64 KB
ROM:	256 KB
Graphics Details:	480 * 32 mono which works out as 80 x 8 characters.
Sound:	Beeper
Storage:	PCMCIA/Jeida SDRAM cards (up to 1MB)
Operating System:	N/A
Interesting Facts:	Size/Weight 295 mm * 210 mm * 22 mm / 930 g

First, lets talk about the build quality which is important for this type of device. Well for a start it's nowhere as strong as a modern day laptop but then again it weighs a lot less. That not to say that it is build badly but lifting it incorrectly, say from the bottom of the keyboard does cause it squeak a little under it's own weight. Similar to the sound of picking up a original CPC, or indeed a PLUS by it's lower keyboard. However with a little bit of care it should fall apart on you. The keyboard seems tough and all ports can take force when plugging in a cable without fear of damage. The screen seem fairly good as well as the pull down leg stand which only seem breakable with lack of care.

On turning on the computer, the screen is readable with good contrast achievable from the control at the right side of the computer. The image can seem slightly blurred if view from a high angle but the pull down legs help fix that problem and viewing it directly or at a lower angle does show a good sharp display. The only let down is the lack of a backlight, ok I know, backlights would ruin the battery life but still, it would have been useful in poor light conditions.

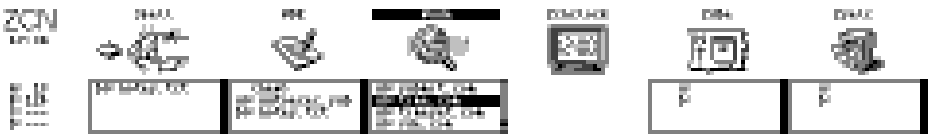
There are three options on screen at start-up, which are Word Processor, Calculator and Diary/ Address Book. All of which are easily accessible with a combination of the multi-coloured keys.

Function + Red gets you the word processor which we briefly mentioned above. Those who have used Protext on the CPC will feel right at home as it practically identical to use as the CPC version. In fact the NC-100 manual describes it as a special version of Protext by Arnor.

The word processor has a 48,000 word spell checker with the ability to add more words if needed, although this eats into available memory. Spell checking is right literally at hand by pressing Function and '=' and while it not the speediest by today standards for spell check, it does it in a reasonable enough time stopping on each spelling error displaying a list of options. It actually works quite well to the point of been enjoyable to use.

There's also word count, cut and paste, find and replace, keyboard macros, accented characters, case changing, un-delete and mail merge. Most of these features work well enough while some can be an awkward process, especially by far, the mail merge. If you want a head-ache, please do, use the NC mail merge function. It not that it difficult to use, well not after reading the manual, the problem is that it just plain messy.

On to the calculator function, big text calculator, accessible by the green text buttons on the keyboard which are hard enough to find. For some reason the eyes don't get drawn to the green keys making it difficult to use. The calculator isn't even scientific, the only advantage it has over a normal pocket calculator is that it can display 12 digits on screen instead of 8. You'll be better off to use BBC Basic to program a better calculator, more on that in a moment.



Now, the address book and dairy, it works, it not pretty but it works. I could say that it works reasonably well and in most cases it does, taking into consideration the limited memory available. It certainly better than some of those cheap personal organisers you can get in Argos.

The last paragraph reminds me, the NC-100 is a computer, not an or-ganiser. Some may disagree but think of it like this. A organiser, well, organises, had limited functionality and generally can't be used for other stuff. A computer on the other hand can be programmed to have many uses or to use software for certain tasks etc. Going by that alone, the NC-100 is a true, portable, user friendly computer. What makes the NC100 more than a oversized organiser is the build in BBC Basic programming language.

With BBC Basic (which can also work with Z80 assembly instructions) programs can be written to allow it to do many other things from simple games like Breakout to applications like PCB CAD (which is for simple circuit board design) and Accounts. To enter BBC basic is simple, just hold down Function and B where you can type commands and write little programs until the early hours. Check out **http://www.ncus.org.uk** for more programs which can be downloaded.

Even after the hours of Basic programming to be had you still not convinced that it's a computer, then try the CP/M operating system, the final prove that this is indeed more than an organiser. After all the NC-100 has a Z80 processor running at 6MHz. For more information on CP/M for the NC, check out **http://www.ibiblio.org/zcn.**

Getting back to even more very useful functions that are already build into the computer, namely the Alarm, time zone manager and diary manager. All are self explanatory and work well. Even when the computer is switched off,

providing that it has enough battery an alarm will sound and grows in volume until it stops when interrupted or after a short time period. When the computer is switch on, the diary reminder info or alarm info will appear. It's simple but yet a feature that works well.

The NC has about 30 hours of battery life using 4 AA batteries with one button battery for keeping data safe on flat batteries. This instantly makes the NC better than any modern laptop if you planning on typing to be your main use, when weight and battery life is of high importance not beats the NC for that. Well ok you could get a PDA or a Psion 5 but they don't have great keyboards. The nearest modern competition is the AlphaSmart Dana which is vastly superior, well it would have to be, it was made in 2003, but really it don't look the part, does it. Also you're unlikely to find one for pea nuts on eBay, like you would with the NC100. I'm not saying that the 1992 Amstrad is better, it not, but it is the one I would go for, providing that a few limitations are fixed first.

Firstly, the NC-100 is lacking in RAM, only 64 KB with less than 40K available for documents (less 40,000 words) so for any serious use a 1MB SRAM card is needed, giving room to type over a million characters. The second problem is getting your documents off the computer and onto a printer, direct connect with a old dot-matrix is fine but modern printers, not a chance. So the only way to solve this is to use a null modem cable to connect the NC to your PC and transfer files that way into a .txt format on the PC. Certainly messy and not for everybody. But the manual is good at explaining how it works anyways. So if the above don't seem much of a problem to you or if you just looking for another retro item to collect you could do a lot worse (Think Cambridge Z88) than an Amstrad NC-100.

CPCO RATING

8 out of 10

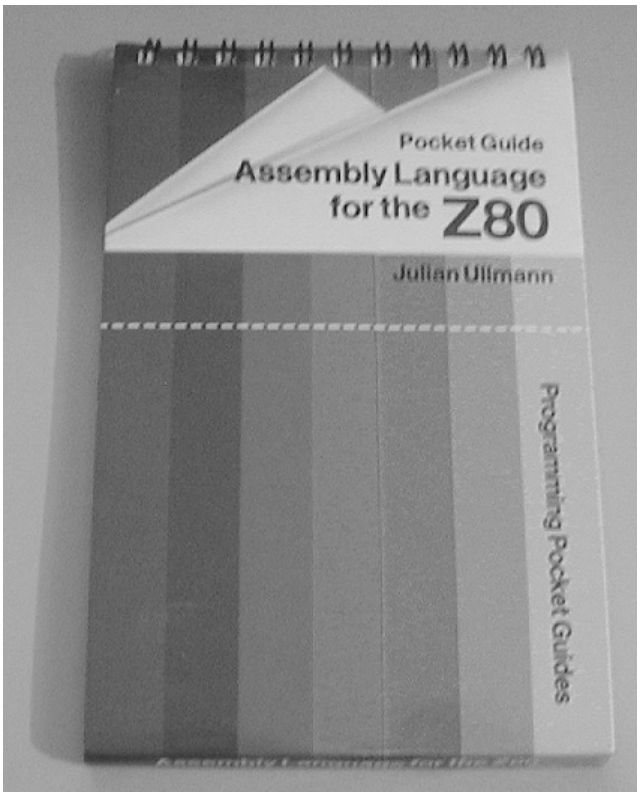
FINAL WORDS

Providing that you can live with the limitations, the NC-100 is a nice and still useful computer to have in your collection.

Pitman Pocket Guide: Assembly Language for the Z80 (Julian Ullmann—1984)

Written by John Kavanagh

This pocket guide provides a introduction to Z80 assembly programming as well as being a simple reference to fall back on. It's fairly small and is made to stand in an upright position for easy reading while typing, and that's where the problem starts (being small) which we'll talk about a little later.



AT LEAST IT GOT A NICE COLOURFUL COVER, AND WE DISPLAYED IT IN BLACK AND WHITE JUST TO GIVE YOU A REASON TO SEEK OUT A COPY!!!

The guide is not tied down to any one computer format therefore it would prove useful to Spectrum users who want to get into machine code as well as any other person wishing to program a Z80 based computer or device. Being not specifically tied to one format directly limits it appeal in many ways as it makes the type in programs quite boring being only for things like adding two numbers etc. For example, a program to 'Read a 2-digit decimal number and output it as a 2-digit hexadecimal number'. Ok, it's all a part of assembly programming but did the writer had to make it this boring? Worse still, the book can be confusing in parts making things a whole lot more complex then it needs be.

Getting back to it size, that's the main problem as the writer tries to cram so much into 60 mini pages which is a pity as there is some good writing talent there. It not all bad as there is wonderful content for those who want to brush up on a few things, for example... Interrupts. Also there are a few handy flowcharts to help explain things a little better, but these plus points still fall short of making this a great guide for want-to-be programmers.

Overall I would not recommend this book, and if you unfortunate to have gotten a copy, pass it on to a Spectrum user! It not a bad guide by any means but why have a average guide when you could have a great book?

CPCO RATING

5 out of 10

FINAL WORDS

Only slightly interesting at the very most. Try something different if you really serious about getting to grips with Z80 programming.

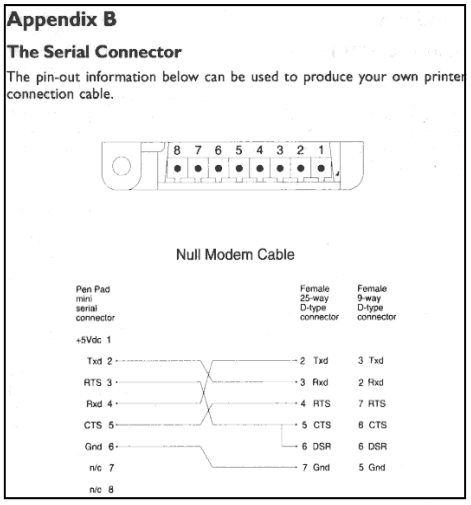
Amstrad PDA-600

John Kavanagh reviews the first PDA! The Amstrad Pen Pad PDA 600. Is it better then a filofax?

On first impressions the Pen Pad 600 is bigger then expected, in fact it felt gigantic next to modern PDAs. It's like the feeling you get when you take another look at a mobile phone from the 90's, seemed small and compact at the time but now it feels and looks like a heavy brick. The Amstrad PDA-600 does feels like that being 16 cm by 11.5 cm and over 2.5 cm thick, for those who don't like measurements, think filofax, only slightly smaller and of course, more fun.

Yes, fun as things do become better once you get used to the size and have a little play with it for a few minutes. Aimed for the business user at the time of launch it still never the less hard not to dive in and play, for example, click the note section which can be used as a jotter or scratch pad, fun. Basically in this mode you can use the supplied (total of 3 supplied) stylus like pen on paper to draw or write anything you want. There are 3 different line thickness and an eraser mode. Using it feels natural and a little additive, although the stylus feels a little thin you'll soon become accustomed to it. Adding new pages can be done with one click of the insert page icon at the top left of the screen, deleting is equally easy by just dragging the page to the left simulating removing a page from a filofax.

The number of pages that can be stored is quite limited, depending on the complexity of the drawing and can range from anything from about 2 percent to 10 percent of memory. Chunks of black takes up a lot less memory then load of lines over lapped horizontally, vertically and diagonally suggesting that there's some form of lossless compression in operation.


Appendix B

The Serial Connector

The pin-out information below can be used to produce your own printer connection cable.

8 7 6 5 4 3 2 1

Null Modem Cable

Pen Pad mini serial connector		Female 25-way D-type connector	Female 9-way D-type connector
+5Vdc 1			
Txd 2		2 Txd	3 Txd
RTS 3		3 Rxd	2 Rxd
Rxd 4		4 RTS	7 RTS
CTS 5		5 CTS	8 CTS
Gnd 6		6 DSR	6 DSR
n/c 7		7 Gnd	5 Gnd
n/c 8			

THE MANUAL IS GENERALLY INFORMATIVE , THAT IS, EXCEPT FOR PIECES LIKE THE ABOVE WHICH TELL YOU TO MAKE A CABLE IF YOU WANT TO TRANSFER FILES!

ADDING MORE RAM IS RECOMMENDED FOR SERIOUS USE.

Most of the other functions requires teaching it to recognize your handwriting. If on getting the PDA, it doesn't run through setup, it is advised to remove all batteries including the lithium battery for a few moments so that once you reinstall the batteries the PDA memory will be completely erased and will run through the setting up procedure. This is important as it allows you the set up the PDA with your own handwriting and settings.

It will run through a few menus asking to set language, date (which can be a real pain forwarding each month from 1993 to 2003!), stylus calibration, time and finally handwriting recognition. From there it run you through each letter of the alphabet in lower and upper case, and also numbers.

That the setup procedure finished, however from my use I notice the PDA getting confused with similar shape characters but this can be fixed later on with a keyboard icon on the bottom of the screen giving you a chance to re-train the PDA for those problematic characters.

The other four icons at the right of the screen are Address/Phone book, Diary, To-do list and Conversion. The Address/Phone book works like a filofax with the alphabet to the right of the screen which can be selected to bring up the page where it displays any surnames beginning with that letter. To add a name is easy by selecting a blank space with the stylus and writing the name into the 21 boxes that comes up. The big limitation of the PDA-600 is that each box has to contain only one character which can be very limiting to a lot of people.

The Dairy fairs better letting you view appointments in daily, weekly and monthly screens. In the daily view, it is split into two separate sections with the top being for entering data that will be turned to text (like in the Address /Phone book) and a bottom part for jotting down notes and drawings (like in the notes section). As you can imagine, this works very well indeed and is one of the most useful features of the unit.

The To-do list is just that, where you can enter tasks that must be completed at a certain date. The Conversion is for converting from different measurements (for example from miles to KM) with the option of defining your own conversions. Other features that can be accessed from the top of the screen is the Calculator, Anniversaries, World time, Calendar, Alarms (up to 10, either daily or once off) and a search function.

This gives the PDA all the functions you could ever require to get organized. From using it I can feel it is more of a handy tool then another useless gadget and can be of genuine use even to those who are technophobes. It also fairly fast in operation and there's not much waiting when it does it stuff, although the limited handwriting recognizing can be a turn off to many. Another limitation is the lack of memory especially if you intent on making a lot of notes. This can be easily fixed with a standard 2MB SRAM PCMCIA card.

While the screen is not great it is still very readable in normal lighting conditions. The battery consumption is only 12 hours or 3 months on standby but this can be expected considering the year it was made and that fact that it contains three processors! Even for a device dated 1993 it can give cheapo so called PDAs a run for its money and can easily be gotten on Ebay for a few Euro, and if you're a retro collector this is a must have, been the first PDA in the world and all!

Specifications

3 Zilog Z8S180 processors giving a speed of 14.3 MHz. 192 KB of memory with 128 KB accessible to the user. The other 64K is split 50/50 for screen memory and hand writing recognition. Main memory can be upgraded to 2MB with a standard SRAM PCMCIA card.

Screen resolution of 240x320 pixels in monochrome. No shades, either black or nothing. 1 mini serial port (RS-232C)

CPCO RATING

7 out of 10

FINAL WORDS

While not the best for hand writing recognition this is still a very useful PDA and being the first of it kind, a great collectors item. Well worth it considering the prices on Ebay at the moment.

WinAPE - Emulator

http://winape.emuunlim.com/
Reviewed by John Kavanagh

John Kavanagh reviews WinAPE, the Windows Amstrad Plus Emulator by Richard Wilson to see where it stands against the rest of the bunch.

A few months ago (near the birth of CPCO really) we reviewed the 'Arnold' emulator which scored a impressive 8 out of 10, commenting on it's layout, ability to load disc images from zip files and almost 100% emulation. We practically said it was one of the best emulators around, if not the best. So here we are, 8 months on, ready to review WinAPE to see how it compares.
On loading the emulator, you get the familiar Amstrad screen along with icons below and menus above. The first four icons are for the debugger, namely, play, pause, single step and step over. With those you have access to a whole load of debugging feature, while the 5th icon is for displaying information on the CRTC, Palette, Sprites etc. Yes, Sprites, see WinAPE primary aim is to be an Amstrad PLUS emulator unlike Arnold which covers all the CPC models including clones. So comparing WinAPE in the basis of an all round CPC emulator would be totally unfair, what we will do is compare WinAPE up against the PLUS features on other emulators (indeed, if they support the PLUS at all).

To PLUS or not to PLUS

The reason we emphasize reviewing this as a PLUS emulation is because there seem to be some attempt to emulate the original CPC, with options for green screen the selection of different CRTC types and a turn off "Enable Cartridge" option which along with selection CRTC types, crashes the computer. All of which can be forgiving as a PLUS emulator, but not as an all rounder CPC emulator.

Going Uphill

From there onwards things seem to improve, the emulator runs at a nice decent speed, just slightly slower then Arnold but more responsive (although you may get different results on your PC).

There's a nice speed display letting you know that you're running at the proper CPC speed with options to change it from anywhere from 5 percent to a 1000 percent. To play Burning Rubber at maximum would require a powerful PC, setting it to a more reasonable 250 percent brings new life to the game (just try not to blink!), great stuff!

There's also a FPS counter to see how you're PC holding up in the Frames Per Second department (anything above 25 FPS should be good). The counter usually aims for about 50 FPS, but increases at higher CPC speeds. Although the number of animation's remain the same in reality, it does looks smoother due to the emulator increased speed.

Let's Play

On testing a few games, applications and demos, WinAPE can easily equals Arnold. In fact, I haven't found any programs that would not run.

WinAPE allows good control over performance, giving options for 8-bit colour, half size display and sound output of the original PC speaker instead of the sound card. While it sounds terrible, it's a reminder of how superior the CPC was over the PC of the 1980's.

The keyboard can also be adjusted to suit the user taste, which proves very useful.

Assemble!

An exciting feature of WinAPE is it build in Assembler, allowing the easy creation of CPC software on your PC. It's fully fully featured Maxam compatible Z80 assembler Maxam compatible so it'll be familiar to users of what is considered, the ultimate assembler. There are also good instructions on the help file for getting started, but understandably, it not going to teach you to program!

Along with the debugger, it's the developers paradise.

Future Visions

Here are a few things that will appear in either the next version or later versions of WinAPE, in Richard Wilson own words.

"**1.** ZIP support for the next version of WinAPE.

2. A Disc Image manager. Basically a version of Para-DOS integrated into WinAPE. It will provide all the features of ParaDOS, working with DSK and other disc image formats. i.e. The ability to copy files to and from discs, format discs etc etc. It will also allow you to transfer files easily from DOS to and from disc images, a sector editor, hex and ASCII file editors.

Part of this functionality will be the ability to store files for the WinAPE assembler on a disc image, and write the output from the assembler directly to a disc image.

3. Maybe, but probably not in 2.0 Alpha six, a compiler integrated with the assembler. I haven't decided on the language yet, I might even invent my own.

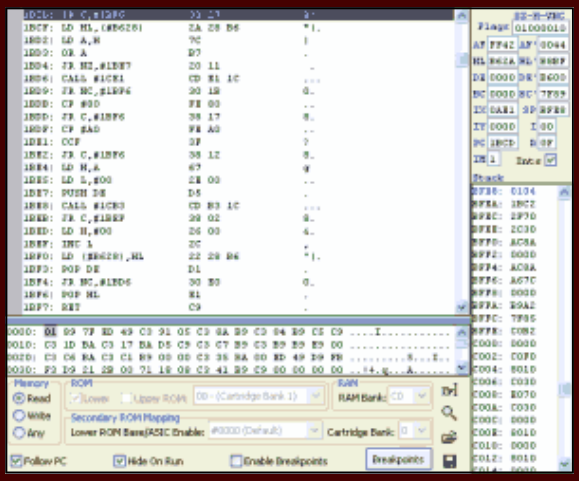
4. Reading and writing of WAV or VOC files for tape support.

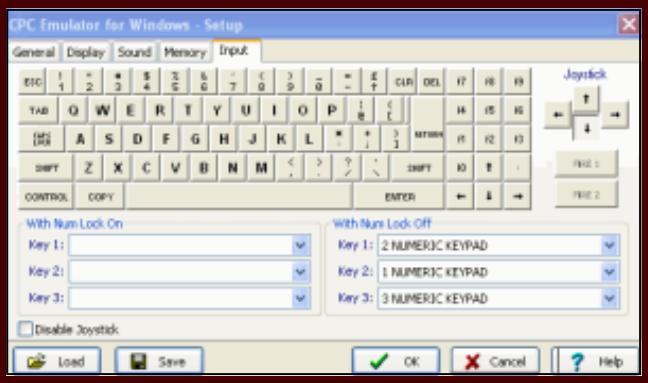
5. I plan on modifying the emulation core to allow switchable accuracy levels, and hence performance levels. It should be quite easy to provide a very fast simple emulation supporting only simple CRTC emulation and no split screen, and some modes allowing split-screen but no cycle level emulation or accurate monitor emulation. This way the user can decide on the level of accuracy required, and if they're only playing BombJack or similar they should be able to run it on a 486.

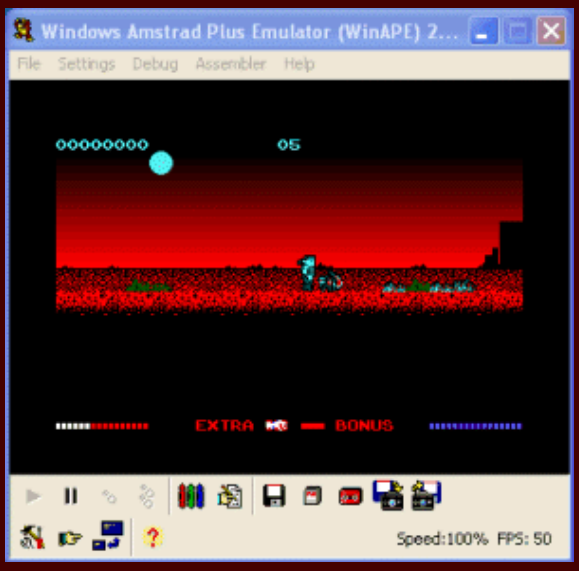
6. All the other features listed in the Feature Matrix which I haven't done yet!"

Richard Wilson

BURNING RUBBER AT 250 PERCENT!
NOTICE THE FPS COUNTER AT 117 FRAMES PER SECOND.







SWITCHBLADE PLAY PERFECT WITH CRISP SOUND AND RUNS AS SMOOTHLY AS IF PLAYED ON THE REAL THING.

The Best of the Best?

One thing that I found really irritating about WinAPE is it's lack of ability to deal with ZIP files, which in my opinion hurts it considerably in this review. While the next version of WinAPE will have support for this (see Future Visions), other CPC emulators already do.

The emulator is not for those who want an all in one CPC emulator solution as this only really supports the PLUS range.

However it equals and even betters other emulators in features such as saving/loading snapshots, tape images etc. Also let's not forget the debugger and assembler.

So should you download and use it? Of course you should, it's free! but let's imagine this for a second: If all emulators cost money, would I buy this? The answer is yes, for the cool speed controls and the build in assembler, but I'd also be buying a copy of another emulator to accompany it, mainly for support of all the CPC models.

Let's get back to reality for a second, not only it's free, it's has features that put other emulators to shame, so download it now at http://winape.emuunlim.com/ and have a go at Burning Rubber at 250 percent, you END

CPCO RATING

7 out of 10

FINAL WORDS

Not the best emulator but with the plans for future versions, it could well be the emulator of choice.

Arnold

Reviewed by John Kavanagh

Accuracy is the key as John Kavanagh reviews Arnold, a CPC emulator for Win32 based systems.

My first impressions of 'Arnold' were good, unzip the files to a folder and double clicked Arnold2.exe and a small window comes up with the start-up screen of a CPC, all very simple for the Windows PC generation. All five types of CPC's are covered plus the VEB Mikroelektronik KC Compact which is a CPC clone from Eastern Europe.

On loading a few games onto the emulator I notice a drop in performance, especially when a lot is happening within a game. The first sign of this is when the sound begins to distort badly. This is more noticeable on cartridge games like Burning Rubber. On checking the documents I soon realised that I was testing the emulator on a PC system which are lower then the minimum specs for Arnold, it recommends a Pentium 2 or higher!

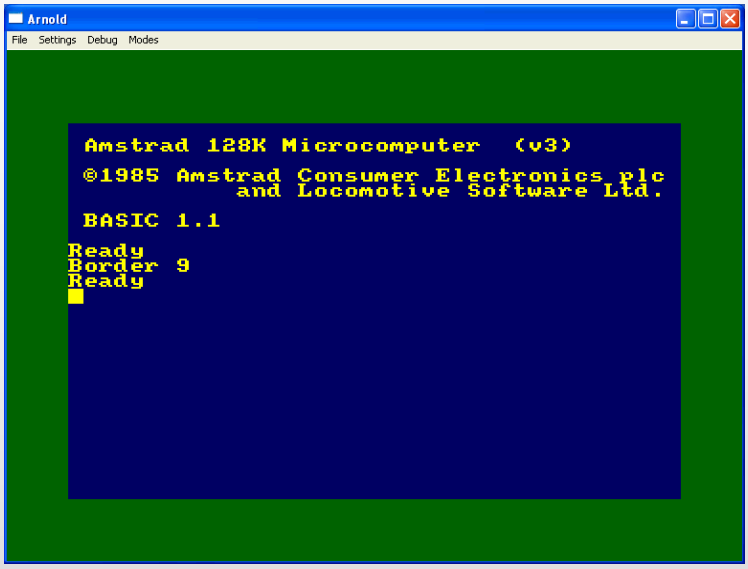
There are two main reasons for the need for such a powerful computer to run Arnold at full speed, not including the fact that it was written in C instead of Machine code. The first reason is that the emulator is made to be run from a Windows platform and not a DOS based system, this as you can imagine increases the overall processing horsepower needed to even emulated a modest system such as the CPC.

The second reason is due to that of accuracy. Arnold amazingly emulated and behave exactly as a real CPC should. For example you can choose which CRTC is used and the computer will look and work slightly differently when playing demos and demanding games. The emulator does not allow you to choose a CRTC for the plus and have an original CPC selected and vice versa. The same goes for other components such as the monitor. While you can select a black and white monitor for lets say an 1984 CPC464, you can't however choose a green screen for a Plus machine. This I find a little weird as there is nothing stopping a person from plugging a green screen monitor into an Amstrad Plus by just changing the monitor connection, a small project which we may cover at some point in CPCO.

- Emulator

ARNOLD

The accuracy of this emulator goes way beyond mind boggling with so much of the CPC/Plus emulated it's hard to know where to continue on. For a start, there's the full support undocumented features of the CPC+ ASIC and even undocumented Z80 opcodes and right down to the quirky-ness of the KC Compact. And that just the start! While there are a few things that are not emulated such as analogue joysticks and monitor brightness control, you still got to be impressed by what is emulated. Certainly one of the more accurate emulators around.



POSSIBILY ONE OF THE EASIST WAYS TO GET A CPC ON YOUR PC

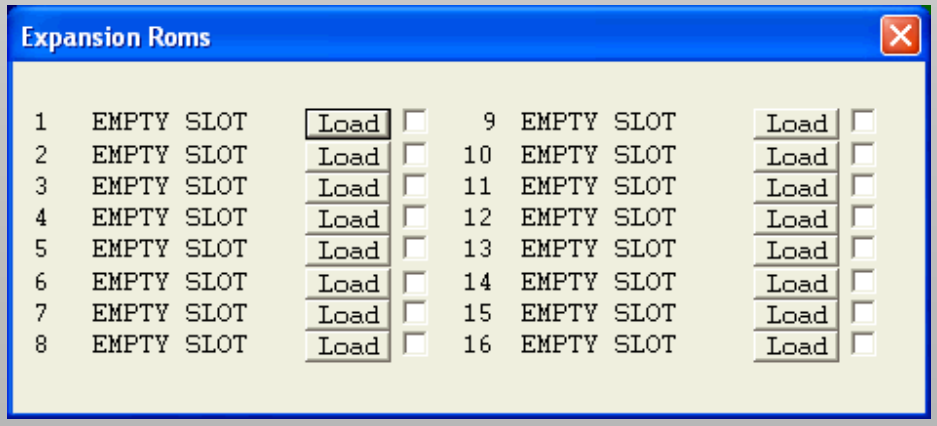
Of course emulating the CPC would be nothing without emulating it add-on components and Arnold does not fall short with full support for mice, digital joysticks, ram expansions such as the DK'tronics 64k or Dk'Tronics 256K Silicon Disc, Multiface 2 and many more. Arnold has full support for ROMS such as Maxam and Protext and up to 16 ROMS can be added at a time just like a real CPC.

Using the emulator is quite easy while still remaining with enough goodies to keep the more hardcore users happy. Inserting a disk image is easy, just click file, drive a, insert disk and select your file, you don't even have to unzip it first if you use .zip as the file type. There is also support for tape images such as .cdt and .tzk, and yes of course, there's support for plus carts. Every emulator would not be complete without snapshots and Arnold does not disappoint, with support for 64k and 128 snapshots using either snapshot version 2 or 3 version. You can also take screen grabs although I've found no short cut way of doing this as the menu systems seems to be lacking in shortcuts buttons.

The strongest part of this emulator, apart from the accuracy it processes is the debug system. From here you can get access to the system Registers, disassembly, memory dump and so much more. Just the thing when doing that little bit of late night programming.

While not a great piece of perfection this emulator certainly comes close with only a few minor things not working as they should, like the menu system for example. Also in some full screen modes the screen can becomes quite distorted and in some cases not very viewable. This all can be forgiven as this is not considered the finish product and it can be sure that those minor problems will be fixed in future releases. The speed issue is not really an issue since I was testing the emulator on an early pentium system which is well below the minimum specs. Taking all of this into consideration I fully recommend Arnold, especially for people who don't want to mess around with DOS based emulators which can be awkward to the newcomer.

END



JUST LIKE IN A REAL CPC YOU CAN ADD EXPANSION ROMS UP TO A MAXIUM OF 16 IN FACT

CPCO RATING

8 out of 10

FINAL WORDS

This is an emulator that nobody with even a passing interest in the CPC should remain without. Download now!

10 OF THE BEST:
AMSTRAD WEBSITES

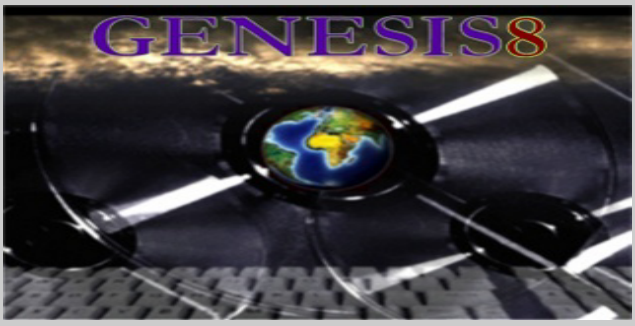
The following is our choice for the top 10 best Amstrad based websites. It was a hard choice and it was touch and go with which to decide upon. Although sites such as Amstrad.dk and CPC Game Zone stand out for all CPC tastes, others like the Unofficial Amstrad WWW Resource are only suited for techies so the top 10 might not be for everybody tastes.

The 10 were chosen overall for content and not layout / eye candy. Also we only selected from websites that contain English sections leaving out French, German and Spanish sites for a later issue.

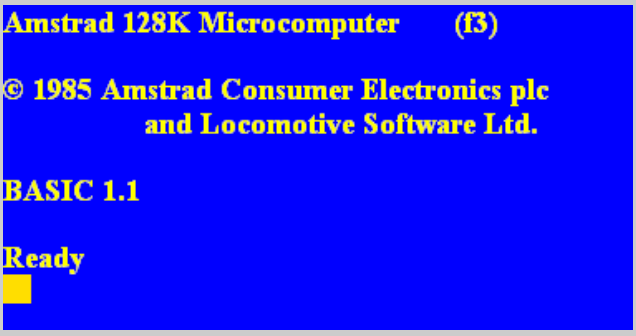
Amstrad DK
http://www.amstrad.dk

CPC Games Review
http://www.cpcgamereviews.com

Genesis 8
http://genesis8.free.fr



Le Site Amstrad
http://amstrad.cpc.free.fr/index_e.html



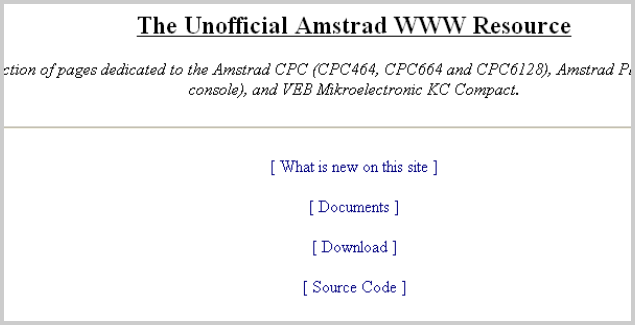
CPC OXYGEN
We didn't mention CPC Oxygen site at http://cpcoxygen.digi-alt.net cause that's for you to judge, not us. If there's anything you like to see added, just let us know and we can see what we do :-)

Ron's Amstrad PCW page
http://www.king27.freeserve.co.uk

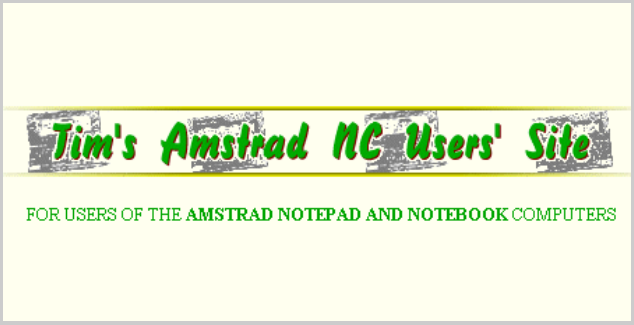
The Amstrad CPC Games Resource
http://tacgr.emuunlim.com

The CPC Zone
http://cpczone.emuunlim.com

The Unofficial Amstrad WWW Resource
http://andercheran.aiind.upv.es/amstrad



Tim's Amstrad NC Users Site
http://www.ncus.org.uk



Unoffical Amstrad Computer Site
http://web.ukonline.co.uk/cliff.lawson

ACTIONZONE

Contents

Game Reviews

Games Chart

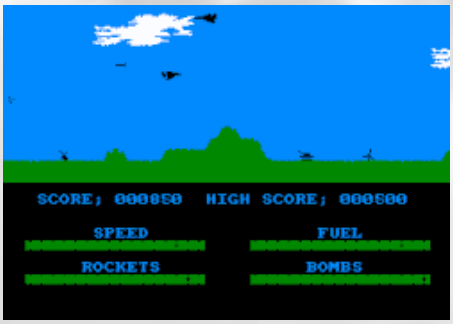
The monthly 'Games Chart' is basically a score board of the best game ratings from all the past issues of CPC Oxygen, with the highest scores at the top. This is decided by the overall rating from the reviewer. When two ratings are the same then the "Score System" is used to determine the positions. The score system is the sum of all the points awarded to a game which are Graphics, Sound, Lastability, Playability and Overall. If after all that the scores are still the same then it's up the editor to decide the games position. See the CPCO web site for the recent version.

NO	GAME	ISSUE NO	REVIEWER	OVERALL	SYS SC
01	The Sentinel (1987)	02	Matthew Johnson	10	47
02	Barbarian (1987)	05	Mark Hall	10	43.5
03	Rick Dangerous (1989)	10	Mark Hall	9.5	43.5
04	Continental Circus (1991)	10	Mark Hall	9.5	42.5
05	IK+ (1988)	06	Mark Hall	9	43
06	Renegade (1987)	07	Mark Hall	9	43
07	Trail Blazer (1986)	07	Mark Hall	9	43
08	Switchblade (1990)	10	Mark Hall	9	42
09	Operation Wolf (1988)	06	Mark Hall	9	39
10	Wec Le Mans (1988)	05	Mark Hall	9	39
11	Robocop (1989)	04	Mark Hall	8.5	42.5
12	Aliens (1986)	10	Mark Hall	8.5	42.5
13	Star Wars: Return of the Jedi (1988)	07	Mark Hall	8.5	41.5
14	Lotus Esprit Turbo (1990)	07	Mark Hall	8.5	36.5
15	Star Wars (1987)	06	Mark Hall	8	34
16	WWF Wrestle Mania (1991)	10	Mark Hall	7	34
17	Harrier Attack (1984)	3	John Kavanagh	6.5	33.5
18	Star Wars: Empire Strikes Back (1988)	6	Mark Hall	6	26
19	Outrun (1987)	6	Mark Hall	1	5

Harrier Attack!

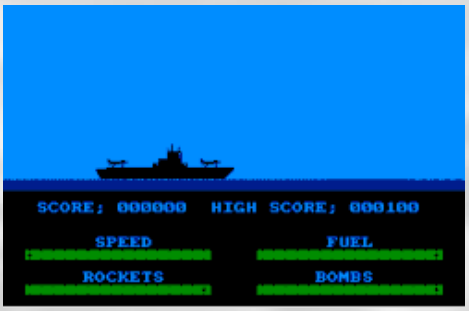
John Kavanagh gets into his flight suit as he reviews one of the first arcade jet fighter games for the CPC.

Being the very first game I ever played on the CPC, I have fond memories of playing this game, competing with other members of my family, going further, getting higher points, being amazed at the awesome graphics! Ok, I can imagine a few of you here wondering what this guy on? But you got to remember this was the 80's and the only other video games I saw at that time period was from an Atari 2600. Compared to the 2600 games, Harrier Attack was truly amazing, at least in my mind it was.



BE CAREFUL OF ENEMY PLANES HIDING BEHIND CLOUDS

Harrier Attack is basically a side scrolling, fighter plane, arcade style game. Your mission is simple, take off from your aircraft carrier and fly towards the enemy island where you encounter warships, fighter planes, tanks and other ground vehicles. Dodge the flak, don't get hit by the missiles, shoot the fighter planes and bomb a few tanks without crashing into a hill and you just might make it to the enemy base. Although this actually sounds hard, it's not. After a few turns you'll find yourself flying over the enemy base ready to bomb the crap out of it with
ease especially on level one. Later skill levels from two to five requires more



YES WE KNOW THAT THE PLANE IS OUT OF SCALE WITH THE SHIP. BUT SO ARE MOST GAMES FROM 1984?

WIMP OR WARRIOR? YOUR FIRST ENCOUNTER WITH AN ENEMY JET FIGHTER! WATCH OUT FOR DEADLY MISSILES

THE FIRST FEW SECONDS OF PLAY AND YOU'LL REALISE THE SIMPLE BUT FUN GAMEPLAY THE GAME HAS TO OFFER

manoeuvring as the Jet Harrier suffers more from flak fire which fills the sky. There are two types of weapons, bomb and missiles. It don't take a genius to know that the bombs usually used for the ground and sea targets with missiles being used to shoot down planes. Although playing the game, fighter planes could be bombed as easily as any ground target, but this only adds to the overall fun of the game.

That is basically what's the game is about…. fun, although short lived. The trouble is that after going through level one (which is quite easy), there not much to do after this, except land on your aircraft carrier and taking off again to blow the island up on a harder skill level. Boredom soon sets in and you'll bomb your aircraft carrier, complete a mission and funnily enough, end up with no carrier to land on and end up flying

The Controls

Using Joystick

Left and right slows and speeds up while up and down control height. Press fire to use Missiles and SPACE for bombs.

NOTE: To select joystick, when the games loads press J, and then 1 to 5 for skill level.

Using Keyboard

Take Off and Up	**.**
Down	**ENTER**
Fire Missiles	**0**
Drop Bombs	**SPACE**
Slower	**Z**
Faster	**X**
Eject	**ESC**

Other Formats

Commodore 64 (1984)

Sinclair Spectrum (1983)

straight out to sea before running out of fuel and crashing into the sea. Although a nice touch is the eject button which can earn you a 1,000 points if ejected just before loss of life.

Not the type of thing you get up to in the game unfortunately!

Graphically the game is primitive, with only text characters reshaped to make sprites, which explains why they are all just one colour, black! Nevertheless they get the job done and in fairness, nice and crisp to look at. The sound f/x are better, with the pitch of the harrier changes with speed and altitude, great for a game from 1984! Explosions are also spot on, although nothing amazing.

Compared to later games for the CPC, Harrier Attack fails in comparison by not using the full abilities of the computer, but then again, can you name a game that did in 1984? Harrier Attack was made for lesser computers and it shows by been largely, graphically speaking, out of date within a few months. None the less it's still a great game and one that worth your download time.

CPCO RATING

Graphics 6

Smooth and without garish colours which is amazing for a game of this time period. Every other game I could think of had garish colours, who needs them when you can have blue sky, green hills, bluer sea and errrr black sprites. That's the big let down, but what can you expect for rear-ranged characters.

Sound 8

Explosions, anti-aircraft fire, and the sound of the harrier as it flies by, what more do you need? The f/x are good but there just not enough of them. There also no background music but I think that's a blessing.

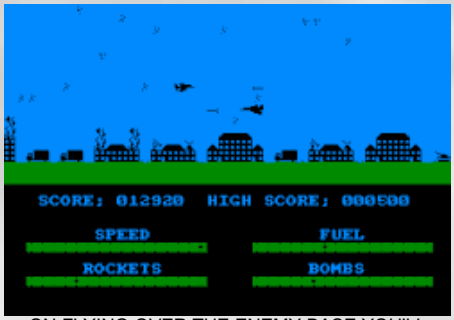
Gameplay 8

It has the type of gameplay that's easily to get into and quite fun. Although a few would say that the gamplay is too simplistic.

Lastability 5

This is where the game really looses out, there just not enough to keep you interested. But it's a nice game to go back to once in a while to pass a few minutes of time.

Overall 6.5



ON FLYING OVER THE ENEMY BASE YOU'LL ENCOUNTER A LOT OF ENEMY FLAK AND FIGHTER AIRCRAFT. JUST BE SURE TO DROP A FEW BOMBS

SURVIVE THE SURFACE TO AIR MISSILE AT THE DOCKS AND YOU HOME DRY, SO TO SPEAK, ALTHOUGH THIS IS NOT REALLY MUCH OF A CHALLENGE

AFTER FLYING ACROSS THE ISLAND YOU SHOULD BE HEADING BACK TO YOUR AIR CARRIER, THAT IS, IF YOU HAVEN'T BLOWN IT UP!

The Sentinel

Reviewed by: Matthew Johnson

In our first game review Matthew Johnson let us know the question.... what is The Sentinel?

Some games defy catagorization. This is one of the true original computer games of all times. Hardly a suprise since it is the brain child of Geoff Crammond who was also the designer of Stunt Car Racer (available on just about every home computer) and some critically

The first landscape with the sentinel.

acclaimed Grand Prix sims on the PC. The game is difficult to describe, but I'll try anyway.

In this game, you are a robot in a 3d landscape. Starting at a low point, you must gain enough energy and altitude

Having just transfered, you face the robot you came from. It and the boulder it stands on should be absorbed ASAP.

to make an assault on the Sentinel who is the overseer of the world. The Sentinel doesn't like energy imbalances such as yourself and if it spots you, will attempt to disperse your carefully collected energy until you are anhialated, or you move out the way... and that's where the twist comes in, you can't move, not in the usual sense.

The chequerboard landscape is littered with trees, which can be 'absorbed' to increase your energy. After the first level you may also come across sentries, which are like little sentinels and cause just as much havock, but are not as high up. In order to absorb something, you must be able to see the square it sits on, so just because you can see something doesn't mean you can get it. This also makes altitude important because you can only see the squares that are on the same level as you or below. Fortunately the same applies to the Sentinel and Sentries, so you can sometimes avoid their glare by hiding behind part of the landscape.

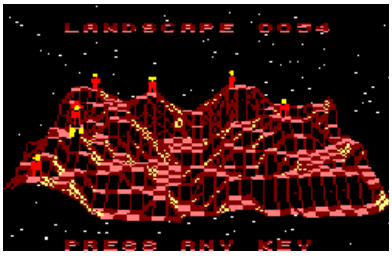
The game is played in 1st person perspective. You view the world through the robot's eyes. You can look up and down and around, but you cannot move from your current spot. As well as absorbing things, you can also create trees, boulders and other robots, each coasting energy. In order to progress around the landcape, you must create another robot and then 'transfer' to the new robot, leaving the old robot behind, which can hopefully be re-absorbed to get the energy back, unless the sentinel has spotted it. In order to move up in the landscape, you can create a boulder with a robot on top. When you transfer to this robot, you are able to see the squares of the next level up, and so can absorb more objects and create a robot on the higher level.

The sentinel is always situated on a

Do it... press a key!

pedistel on the highest point on the landscape. From there he slowly rotates around, looking for anything which is not a tree or a sentry. If he sees such a thing (this means you) he will deplete its energy one unit at a time and create trees at random around the landscape, thus dispersing the energy and restoring its sense of reality. This also applies to any robot or boulder you may have left behind, or even one you have just created, if it happens to be in its line of sight. When you are being absorbed, you need to quickly 'create and transfer' to another square. If you are totally stuck you can use the hyperspace feature but that will often put you in a

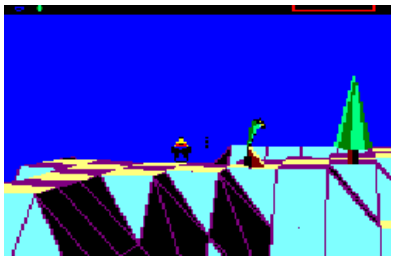
The Sentinel, conveniently facing the other way.



A level with 5 sentries.

Review (c) Matthew Johnson, Jan 2002. Screen shots prepared using CPCEMU, SNAPGRAB, CPC2TIF and IrfanView on a Pentium PC Using Dos and Win95. Email: noisycheese@yahoo.com.au
Please check out some of my music for free on mp3.com at www.mp3.com/cascadehush

A boulder, a tree and a rare meanie (bottom right).



You can just see a robot head in the middle distance but because its square is hidden you can't absorb it.

A sentry, about to be absorbed. Meanwhile a nearby tree is being dispersed by the Sentinel.

And when your done, you get a code for your next level.

worse position.

The ultimate aim is to get high enough to absorb the sentinel, take his place on the pedistal, and hyperspace off the landscape victorious. The amount of energy you have at the end determines how many levels forward you go. There are 10,000 levels, but you can go anything from 1 to around 25 levels forward each time. You access levels with the passcode you are given after succesfull completion of a level.

This is a strategy game requiring careful thought with no time to waste. The overall pace of the game is slow, due to the intesive graphics processing, but this adds to the tention. You constatantly evaluate every move, sometimes sacrificing energy for altitude. The sense of paranoia can be quite real as you hear the sentinel rotating in the distance. There are times when your energy will be despirately low and times when you will come across a bountious grove. The Sentries will be your bane, but if you can get up to their level your energy will soar. Get trapped in a low valley and it will probably be game over. But when it all comes together victory is sweet.

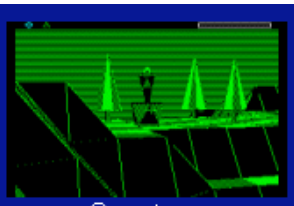
There are few games which have such a long lifespan. As the game gets harder the challenge can almost break you. But you will be back if you are really looking for something to stretch you whit and will. Simple but never easy, this game is a classic!

The Controls

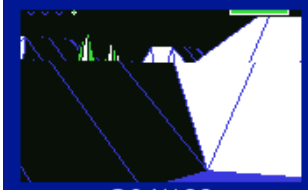
Pan left	**S**
Pan right	D
Pan up	**L**
Pan down	,
Create Tree	**T**
Create Boulder	B
Create Robot	**R**
Sights on/off	SPACE
Hyperspace	**H**
U-turn	U
Absorb	**A**
Transfer	Q

Other Formats

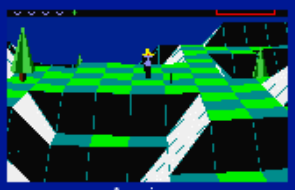
Amstrad CPC 1987
Spectrum 1987
C64/128 1986
Atari ST 1987
Amiga 1988
PC 1989 (Remake in 1998)



Spectrum



C64/128



Amiga

PC (Sentinel Returns)

CPCO RATING

Graphics **10**

Superb 3D graphics for an 8 bit. The graphics for the CPC version were even better than the C64.

Sound **7**

Nothing spectacular, but they do the job.

Gameplay **10**

Original and compelling. Easy to pick up, hard to master.

Lastability **10**

Still playing after all these years. I heard of a guy who played through the 10,000 levels twice (not necessarily on a CPC).

Overall **10**

Reviewed by Mark Hall

I first remember my introduction to Robocop the film round about May 1987, and like everyone else of that time was starved of decent sci-fi action. What a sigh of relief then when this little gem hit the big screens.

Before long my favourite computer magazine (AMSTRAD ACTION) had made an announcement that Robocop was due to be released on the CPC 464 and a disk version was to follow for the 6128. Within what felt like a lifetime I had received a copy through mail order, and little did I know at the time. But this game was about to change my gaming world forever.

The disk came in a big box with a plethora of screen shots on the back that looked just amazing, probably taken from the arcade machine I thought. Upon loading the game I was presented with a musical opening theme taken straight out of the film albeit in Amstrad stereo (foot tapping and all). To my amazement speech was presented as well

ROBOCOP! The errr... Man himself.... Robocop. Half man - all cop, or something like that!

The first level is played in the streets where Robocop basically got to punch holes in everything that moves.

which sounded like a digital recording of Robocop himself, which set the game up just nicely.
The game engine was of a side scrolling nature. This was also excellent as most Amstrad games of that time suffered when it came to left and right scrolling movement on screen (Descent programmers me thinks). I was totally convinced that this game had got potential to be an absolute classic and give the 8-bit world the shake up it needed.

The game takes you on a roller coaster ride through many if not all of the scenes from the movie. You start off in the suburban streets of Robo-

This is where you got to shoot the man without hitting the woman, easier if you've seen the movie.

-cop land against what seems like a formidable amount of enemy, and to make things that little more difficult no weapons in hand. After a short walk you pick up a machine gun and fire is a plenty as thug after thug eats metal.

The game also has 3D sections included one of which has been taken from the film where you find yourself in a hostage situation against (not the brightest bloke I've ever met). The game is also littered with bonus puzzle sections where you have to match a criminal face to the one stored on your computer within a set time limit.

The levels in the game get more difficult as you progress but not overly to a point where it just gets ridiculous. The weapons in the game keep on getting better and better. The later levels find you firing balls of fire that just wipe out anything standing in its way, which you should reserve as this weapon is later needed for the ED 209 units.

All in all this game has everything, 2D / 3D state of the art graphics for an Amstrad, a thumping great soundtrack to accompany the action as you waste waves after waves of enemy. The only gripe I have about this game are the lack of sound effects during action and an occasional slowdown in the heat of battle that unfortunately prevent it from getting a 10/10.

Other Formats

Amiga (1988)

PC (DOS) (1989)

C64/128 (1988)

ZX Spectrum (1988)

CPCO RATING

Graphics 10

Some of the best graphics of the time, which is really saying something as competition in this department was good at the time.

Sound 8

What sound! Music and speech is all you get. One of the best soundtracks I've heard on the CPC for the time.

Gameplay 9

Excels in every department, game play never stops (must get that next weapon).

Lastability 7

Once you have beat the bad guys coming back and doing the whole thing over again is never fun or so I thought.

Overall 8.5

WEC

I first remember this game when I was on holiday in South Wales, a small arcade called Joe's place was at the resort we stayed in. On walking into the arcade while the old lady was busy playing Bingo next door I noticed someone sitting in a round shaped object and turning a steering wheel left and right frantically.

I Quickly walked towards this strange contraption peering to see the screen and to find out what all the commotion was about. My eyes were quickly filled with amazement, I had found the ultimate driving game.

Once the guy had left the arcade machine I leapt inwards and inserted my money. The race was on. It all looked superb and the cockpit I was sitting in would actually shake when moving to the left or the right, on impact with another car or roadside object the car would either spin or throw you violently through the air losing precious time. Boy this was heaven. Coin after coin went in one after the other until I finally beat the game.

On returning from our holiday break I purchased a copy of the arcade phenomenon (Wec-Le mans) for the Amstrad CPC 6128 and proceeded to load the disk version of the game. Loading screen is good I thought and then came the title screen bellowing out the Le mans soundtrack that sounded not to dissimilar to the arcade version.

Upon choosing the controls for the game I waited with fingers shaking as most people who used keyboards to play games in those days did (Joysticks were crap). And low and behold the race was starting, Oh

LE MANS

Reviewed by Mark Hall

my god I thought what's happened to the graphics, I was sure my Amstrad had come with a colour monitor but everything was in green. Never mind I thought lets see how it plays.

To this day I'm convinced that this was the best ever arcade conversion of a racing game and pretty much run everything else close as well, Although the colour was mostly green the graphics held together well and were very fast moving. The collision dectition was of the best I'd ever seen on the Amstrad and there was absolutely no slowdown whatsoever.

This game was as playable in every way as it's big brother sitting in the arcade. Magical moments like the car spinning out of control and high speed impacts would send your car spinning through the air. The only downside being no steering wheel or turn able cockpit included with my £15 pound purchase, oh well.

The game itself only has one track. One does not really notice this throughout the duration of the race. On Later levels the traffic increases and the computer controlled cars crash into each other blocking your path ahead and lowering your ever

decreasing time limit, quick and accurate reactions are needed here to quickly hurtle through the pile of cars gathered at the side of the race track.

The sound and music are great albeit the constant whining of the engine can be a bit annoying. These are just little niggles as the whole thing really does do it's job and adds to the play and look of the whole experience. The graphics actually do work giving the games player a true sense of speed and at no time whilst playing the game did I experience slowdown.

TACGR compared this game to the Spectrum version and although I agree that playability is similar the Amstrad version along with graphics and sound is more complete and a much better all round conversion. Just check out the screenshots to see for your self.

The only gripe that might hinder this game is once you've passed the chequered flag will you want to go back and re live the Le mans 24 hr experience all over again. Well I can safely say yes. All in all probably the best racing game the Amstrad has ever seen.

Other Formats

C64/128 (1988)

ZX Spectrum (1988)

Arcade (1986)

CPCO RATING

Graphics	8
Sound	6
Gameplay	9
Lastability	7
Overall	9

BARBARIAN

Reviewed by Mark Hall

In a dark and grim land a long time ago lived the Barbarian a blood thirsty warrior like race, where each and every sunrise will see them training for survival until darkness falls.

Bodies scattered throughout the plane and those still living praying night will never come. Severed arms and legs along with the occasional head, lye roasting in the hot and torrid Sun. Night after night battle commences in a fight for survival. Body's driven to their physical edge and no remorse for the living or the dead.

Those that survive under extreme circumstances are chosen by the Warrior lord an evil being that scours the land for more Barbarian prey. A sick minded individual who pits warrior against warrior in a dungeon like cave. A fight until the death where heads of many a brave warrior have met a steel cold blade.

But this time he's messed with the wrong Barbarian. The most powerful and fearful Barbarian in all the land, who for years has evaded capture from Drax the evil warrior lord, Drax has now kidnapped his girl and given the Barbarian an ultimatum. Turn up and fight 6 of my finest warriors or the girl dies.

With bated breath and sword in hand the warrior makes his way to Drax's dungeon in a bid to dethrone Drax and drag his girl back to their cave. The game starts with you in control of the mighty warrior with broad Sword in hand. All you have to do is fight six of drax's best warriors then confront the madman himself. Now that sounds easy I here you say. This game is 2D in design and uses the Amstrads graphics chip to perfection. There are 4 different levels you can choose from, one sees you fighting it out in a forest and another actually in Drax's throne room. The game-play is the best I've ever experienced from this style of 2D fighting.

The character can roll forwards backwards and jump. Not to mention the sheer amount of combat moves available, hey you can even head-but. All these moves are great but you haven't seen anything until you see your first victims head roll, yes roll I said. You have to time it to perfection though and after a while you'll be chopping off heads just like Bruce Lee.

Sound and music is also top, there are humorous musical touches throughout the game. The one that bears to mind is when your opponents head gets chopped off a little green goblin walks across the screen and takes a huge kick at the blood soaked head, Brilliant stuff.

All in all this game is a fighting legend and anyone who missed this game has missed out on a true piece of gaming history. Thank you Palace software.

CPCO RATING

Graphics	9
Sound	8
Gameplay	9
Lastability	7
Overall	10

Other Formats

C64/128 (1987)

ZX Spectrum (1987)

Out Run

Reviewed by Mark Hall

You're cool, the engine's hot, the girl's gorgeous, a tank full of gas and an open road...the rest is up to you..!

This was just a small amount of the even greater trash that was splashed all over the cover as I handed over my £15 pounds of hard earned cash, totally brainwashed into thinking this could be as good as the arcade.

I've played outrun on nearly every format out there and most have remained faithful to the Arcade version. The speccy, the C64 and Amiga while not great still had the playability needed for you to have one more go.

So as an Amstrad fan and someone who knows quite a bit about hardware, what! the hell happened here SEGA.

I'll tell you what happened, the programmers working on the Amstrad version were pants and had ruined what could have been a very good game for us Amstrad owners.

It's now quite safe to say that this was the worst conversion ever to appear on a home micro system. The graphics were an absolute insult and only the car reminded you that this was supposed to be outrun.

Sound was also on the road to nowhere. And instead of coding music into the game they bundled a cassette with the arcade music on to piss us punters off even more.

All in all the worst conversion ever and one to forget about.

Please take into consideration that this review is just my own opinion.

ARCADE SCREENSHOTS

CPCO RATING

Graphics	1
Sound	1
Gameplay	1
Lastability	1
Overall	1

OPERATION WOLF

Reviewed by Mark Hall

Reviewed with Amstrad Light Gun

The noise, the noise, can't take anymore, Please help me. Please God get me out of here. At the same time the mother of all mortar shells rips him wide open. Mortar after mortar shells pound the land and life from beneath the unfortunate soldier's feet.

These are just a few of the cries heard, as bodies lay piled across the battle field. The injured ones driven mad by the constant rattle of machine gun fire flying overhead, not to mention the unthinkable pain caused through the loss of limbs.

Bullets fired illuminate like laser fire against a dark blood drenched sky, on their way to an unsuspected target. And rarely miss.
This is what happens when you mess with a team of crack commandoes known only as OPERATION WOLF.

Operation Wolf way back in 1988 was voted Arcade game of the year. And anyone who has played the Arcade version will tell you what a fabulous game this was and still is 14 years on. Games such as Time crisis and Virtua-Cop owe a lot to the likes of Operation wolf.

The No.1 Arcade game of 1988 was released later that year on the trusty old CPC, the graphics excelled in every department and most had mistaken the screenshots for 16-bit versions. OCEAN quickly decided to use the Amstrad graphics to promote the 8-bit and 16-bit versions and before you could say how much, had sold an absolute shed load.

So the graphics are good and faithful to the arcade conversion, but what about the playability and sound.

Just like being in the Arcades, absolutely fantastic and the sound is very much identical to its big brother. Playability is where it really excels and constantly has you on your toes as wave after wave of enemy soldier's tanks and helicopters try to piss you off to the max. And on the odd occasion are very successful.

The game scrolls sideways and simulates you walking through enemy infested territory.

There are six levels in all ranging from ammunition sites, prison camps and an army airbase. The locations have to be seen to be believed and the constant bombardment of action is just right. For example a guy will run onto the screen and throw a grenade or a knife at you from a near or afar, this creates a 3D appearance of it sticking in your chest and energy quickly depletes.

You can combat this by simply shooting the knives or grenades before they impact. You can also pick up ammo and energy by simply shooting icons before they descend off screen.

Another is when two guys jump out at you and your natural reaction is to shoot them in the body as this is the biggest area thus easiest place to hit, but these guys wear bullet proof vests and are armed to the teeth. Bang, bang your dead next time shoot us in the head.

Overall this is one of the best arcade conversions ever made for the Amstrad CPC range of computers and if you are lucky enough to own the disk version the whole experience feels a little more complete.

ARCADE SCREENSHOTS

SPECTRUM 128K

CPCO RATING

Graphics	8
Sound	6
Gameplay	9
Lastability	7
Overall	9

IK+

Reviewed by Mark Hall

First came the Way of the exploding fist who's legend will forever live on in all our Amstrad hearts, a game based around all style martial artists who if caught you on the chin with a spinning round house kick or head butt would safely see you on your way to meet the spirit in the sky, no stopping to say goodbye on the way.

IK+ sparked off a plethora of fighting games and the fighting genre we all know and love today, these type of games swamped the Amstrad games market in the late 80's with one after the other arriving on shelves, some good some downright ugly. The one that sticks out the most was International Karate.

All these fighting games released fired the imagination for a certain programmer called archer MacLean who decided to follow on where International karate had left off, unaware at the time that IK+ was to become the best fighting game ever released for the home computers.

IK+ presented a unique first time ever three fighters on screen at any one time. Bonus rounds where included for going two rounds with your opponents. This involved a sort of endurance test where you deflected bouncing balls armed only with your fists of fury and trusty old shield.

The programmers not content with balls bouncing all around you mixed with constant fighting. They decided to make things that little more difficult and vary the speed of the bouncing balls that would start off slow and increase in speed and even change direction.

The cover of this game says "Blink and you'll end up flat on your back" and this is exactly what happens throughout your playing experience of IK+ unless you quickly learn the ways of the wise teacher Mr. Miyagi (only joking).

The action is very fast, there are over 20 moves and combinations you can muster and execute at lightning pace, my favourite being the jumping double split kick and seeing the back flip is awesome to. The game pits you against an array of opponents who for some unknown reason all look the same and are only distinguished by the colour of their karate suits. during play you start with a white belt and slowly progress to black. There is no official ending to this game it just goes on and on until you can't.

The graphics and animation in the game are fabulous and the backdrops are superb, for example the sun shines over the sea and the water ripples to create a wonderful affect.

All the stops were pulled out on IK+ and this really shows when listening to the sound track that accompanies play. The track plays for 10 minutes never repeating until fully finished and in my opinion one of Rob Hubbard's finest if not the best. This game is an example of what an 8-bit machine can do when given a bit of TLC, and let's face it this game wasn't even pushing the CPC.

IK+ will always go down in my book as being one of the best combat outings of all time. Thanks Archer.

Other Formats

Commodre C64/128 (1987)

Spectrum (1988)

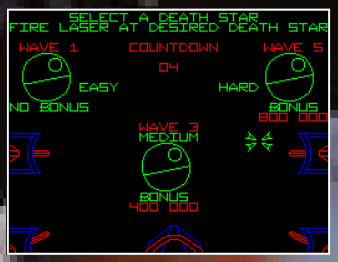
Amiga (1988)

CPCO RATING

Graphics	8
Sound	9
Gameplay	9
Lastability	8
Overall	9

STAR WARS

Reviewed by Mark Hall



A long time ago in a 3D vector galaxy far, far away entered Star Wars the arcade game. This was the first attempt at recreating the film of the 70's and gave people the chance to jump into an X-wing and go one on one with the empire.

The Arcade game all over the world received high acclaim and droves of people flocked to cue and have their chance to become Luke Skywalker and relive the trench scene.

It wasn't long before the home micro systems such as the BBC, Toshiba, Spectrum and Atari XL saw the release of the arcade smash. But these versions failed to impress and suffered mass amounts of slowdown especially on the old speccy.

Born out of greed and a lack of originality "Domark" decided to release Star Wars onto the new flavour of a 8-bit computers, amongst these were the C64 and Amstrad computers and Amiga and Atari ST to follow.

The Amstrad version of Star Wars as far as I can remember was given a good review from Computer and Video Games, but received criticism from some Amstrad magazines. I can guarantee you that if you were to crank up your CPC now and play Star Wars that after about 5-10 minutes you'd probably laugh at the sound and graphics, but in the 80's these were phenomenal and the playability was fantastic if somewhat short lived.

The game was almost identical to the arcade version in the graphics department but lacked the classic Star Wars theme during play thus lost some of the feel of being there in C3PO world.

This game at the time had everything and although on rails gave one of the best roller coaster rides of the Star Wars universe in 8-bit history and managed to immerse you the games-player from the stress of the real world can't be bad.

All in all a classic not to be missed, but as the sands of time continue to decrease and years flash rapidly by, and then after recently playing Rogue Squadron II on the Nintendo game-cube, sadly become a distant and almost forgotten memory in a galaxy far, far away.

Other Formats

Commodore 64/128 (1987)

PC DOS (1989)

CPCO RATING

Graphics	7
Sound	5
Gameplay	9
Lastability	5
Overall	8

STAR WARS
EMPIRE STRIKES BACK

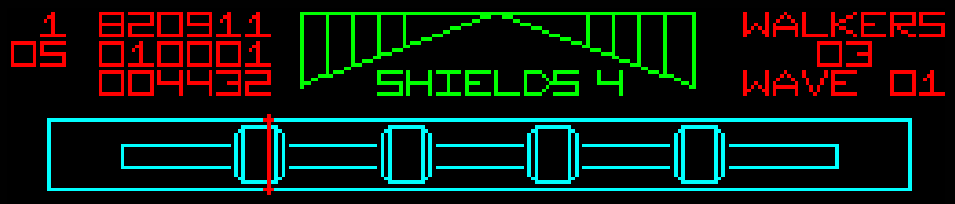
Reviewed by Mark Hall

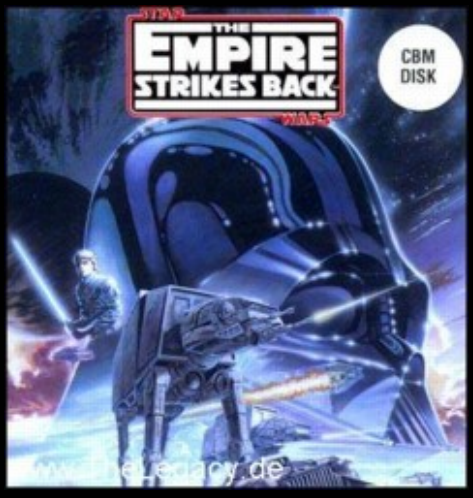
The Empire strikes back was one of the best movies ever and defied all who said sequels are never as good as the first film, or should I say the 4th. So how has the transformation from the big screen to the home computer faired.

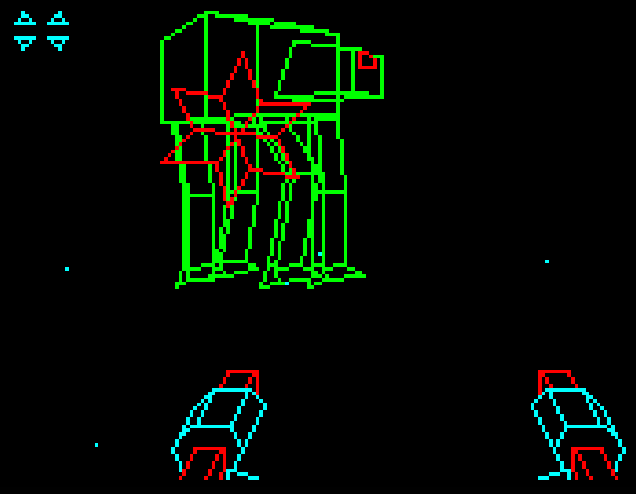
Empire as we all know and love follows on from where the death star left off, only this time we go up against some of the most awesome machinery the Empire has to offer.

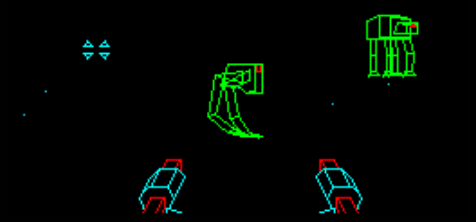
We now get the chance to pilot the famous snow speeder and go one on one with the huge Imperial Walkers and Pro-bots. And if that wasn't good enough, how about flying the Millennium Falcon in all it's glory through a Tie-Fighter infested shootout and the final stage where you have to avoid collisions in the asteroid storm.

The first level has you engaged in battle to try and keep the Rebel hideout a secret and stop the Pro-bots like in the movie from sending photo evidence of the Rebel structures and most of all their power generator back to the Dark Lord. To achieve this you need to shoot anything that moves and shoot the signals that are fired upwards into space.









The second level introduces the AT AT Walkers and the AT ST walkers.

The Rebel base has been discovered and the AT AT's are moving in to attack the Power generator. You can use a limited amount of Tow cables to disable the AT AT and laser fire to disable the smaller but faster units, or depending on how good your aim is, you could always shoot them between the eyes.

The 3rd level puts you in the cockpit of the Millennium Falcon where wave after wave of Tie fighters try to hunt you down until your bits and bobs are scattered all over the galaxy, unless you prevail of course.

The fourth and final level has you piloting the Millennium Falcon through an asteroid Storm in search for a place of safety from the Empire. And there is the problem, only four levels and even though these levels are good enough and entertain It took me longer to write this sentence than to pass the game.

This same problem existed with the Arcade version and was designed to pull punters in to have a quick go at a pound a shot. I just wish the creative juices at Domark could have come up

with a better idea and not just copy the arcade versions for us home users.

The graphics are faithful to the Arcade version and so is the music and sound that was sadly lacking in the first instalment. But even this can't save this three minute roller coaster from it fate of doom. Worth a go just to see the AT AT Walkers.

Other Formats

Commodore 64/128 (1988)

Amiga (1988)

CPCO RATING

Graphics	5
Sound	7
Gameplay	5
Lastability	3
Overall	6

STAR WARS
RETURN OF THE JEDI

Reviewed by Mark Hall

Domark must have some of the best marketing people in the business, not only do they grab the Star Wars licence for all three games but also sell shed loads to us the punter.

So how on earth did they get it so wrong with the Empire Strikes Back that added up to less than 30 second's of game play.

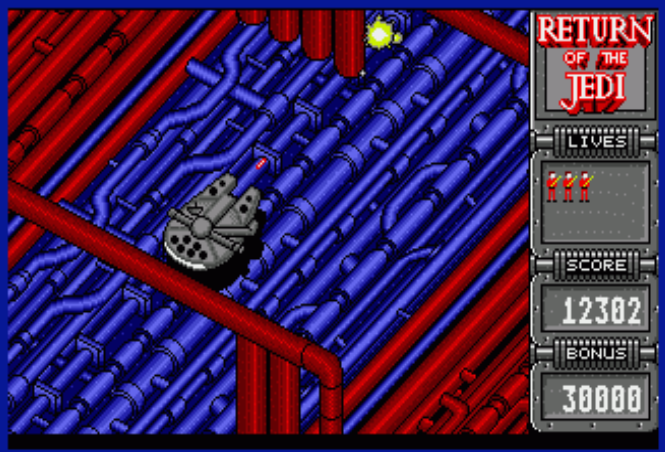
Domark Deliver another Arcade conversion this time in the guild of Return of the Jedi, with the potential to be one of the biggest crowd pullers the Amstrad's ever seen. But have they been greedy again like they were with Empire. And this time well and truly turned to the dark side of the force.

Out with the Vector graphics (Oh so sad) and in with the new super smooth scrolling 2D style full colour view. Well drawn backgrounds are now present and fantastic sprites that range from Speeder bikes, AT ST, and the Millennium Falcon alas are now used in the Star Wars universe to full effect. All this looks super and the later levels will have your jaw dropping in amazement.

The game is the first in the series to follow the film closely and features all the great vehicles that graced us on the big screen. In the first level you take on the role as Princess Leia on her Speeder bike and travel onwards through the forest of Endor in an attempt to reach the safety of the Ewok village. Storm troopers await you and hide behind trees waiting to pounce, not to mention traps you have to avoid that are set for the Empire.

Other Formats

Commodore C64/128 (1988)



Commodore Amiga (1988)

You then move onto another part of the forest to take control of Chewbacca's captured Scout Walker and just like the film Han Solo is waiting for you in an attempt to deactivate the shields that protect the Death Star High above the clouds. This is a fabulous level were you find yourself up against all sorts of Empire scum and must shoot or be killed as other Scout Walkers quickly approach. And this is only the first level.

On the second level you fly the Millennium Falcon as Lando Carlrissian and it is here where you get your first encounter with the famous Tie fighters and yes oh yes the Star Destroyers. Huge big ships that cover nearly all the screen in super Star Wars fashion. Attack waves of Tie fighters and blow up anything that moves.

The third level has you entering the Death Star and again you have to fight off an onslaught of T.I.E fighters on your way to the central reactor whilst also avoiding obstacles along the way. Once the reactor is pulverised you then enter a chase scene that sees a ball of fire hot on your heels and again you have to avoid obstacles to seek your escape. Once the Death Star is destroyed the celebrations and glory hunting begin.

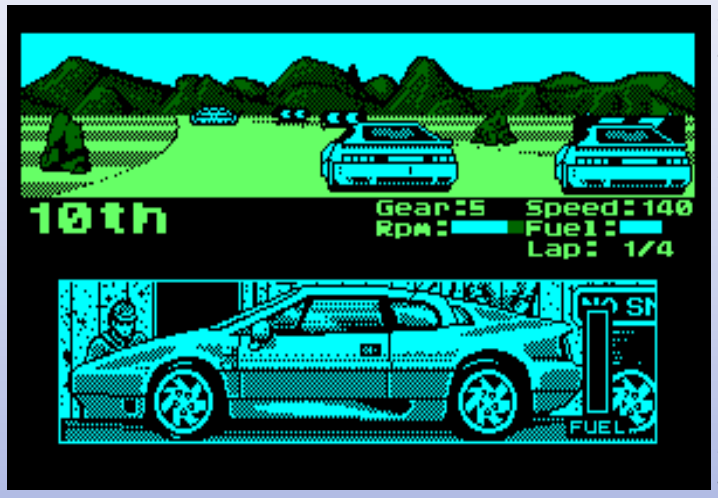
Probably the best Star Wars game ever made for the Amstrad range of computers. An adventure through the Star Wars universe that, although failing to offer anything new, is still one of the best of it's type and a hell of a lot of fun to play. May the force be with you young Skywalker.

CPCO RATING

Graphics	9
Sound	7
Gameplay	9
Lastability	8
Overall	8.5

LOTUS ESPRIT TURBO

Reviewed by Mark Hall



Get your motor running! There's a new set of wheels waiting.

Driving games on the Amstrad are available in droves, but none have really shone until Lotus came on the scene. Originally programmed for the Amiga and Atari ST, Gremlin decided to code it to the Amstrad CPC range of computers and boy what a fantastic job they've achieved.

The game as the title suggests is based on the Lotus turbo range of cars and pits you against some of the best drivers in the world in a bid to be crowned Lotus champion. Each stage has a set number of laps with weather conditions that vary along the way, strategical decisions also need to be made and you need to use these wisely in order to prevail through the ranks and onto later stages as the fuel consumption is a big issue in this game, sometimes finishing just short of the finish line due to lack of fuel.

There are three types of Championship up for grabs and are based upon difficulty settings these range from easy - 7 races, medium - 10 races and hard 15 races. The most unique implementation included in this game like its 16 - Bit counterparts is the inclusion of the two player option and split screen that has survived the conversion very well.

This enables you and a mate to put pedal to the metal all night long. It all works absolutely brilliant and once picked up is very difficult to put down. And probably the best two player racing game that's ever graced the CPC.

There is a downside though, oh yes. The game is a direct port from the speccy version and the quality of presentation and in game graphics shows this in abundance. But like I keep saying graphics don't make a game its what's underneath the bonnet that counts and believe me when I say this goes like a V6 rocket, one in a million.

CPCO RATING

Graphics	5
Sound	5
Gameplay	9
Lastability	9
Overall	8.5

Other Formats

Commodore C64/128 (1990)

Spectrum (1990)

Trail Blazer

Reviewed by Mark Hall

Imagine a race where the track stretches for hundreds of miles and a ball shaped futuristic race pod hurtling and bouncing around at speeds only man can imagine.

Obstacles placed carefully in your way and dark black holes positioned all throughout the track, nerves of steel and the hand and eye coordination of a god are the only thing that will help you stay alive. And if you should fall then forever be lost in the void of space. Imagine the Trail Blazer.

You know Trail Blazers something special the moment it first loads up as the attention to detail on the front end screen alone is out of this world. There is a scrolling text message at the beginning of the game that if you should attempt to read it claims to be the longest text message in the world.

The music is just something else and probably one of the catchiest tunes I've ever heard as you bomb along at the speed of light humming as you go. The sound effects are fantastic also and everything is put together just right.

The game features you travelling down some sort of cosmic race track against the clock. The track is forever changing and if you even blink to rest nine times out of ten you're a goner and find yourself travelling way out into space with a one way no return ticket courtesy of your faithful.

If the black holes don't catch you the different coloured squares along the way just might. That is they can work for or against you and all types of power ups are given depending on the colour squares you run over. For instance one colour will give you a speed boost or unbelievable jump power where as another may reverse your controls or slow you to a snails pace and cause a momentary lapse in concentration enough for you to crash and burn.

There are over twenty levels of play and each zone has to be completed before the sands of time run out. The game sounds overly difficult but given the right amount of practice and just a little patience you'll find its one of the best games ever made for the 8-bit computer.

The graphics in the game are fast moving and hold together really well, the sensation of speed is awesome and never lets up and just when you thought you couldn't possible go any faster along comes a power boost and away you go.

Simplistic idea that is well executed and provides one hell of a roller coaster ride.

CPCO RATING

Graphics	7
Sound	9
Gameplay	9
Lastability	9
Overall	9

Other Formats

Commodore C64/128

Spectrum (1986)

RENEGADE

Reviewed by Mark Hall

A knife edge world, in a city where kill or be killed is the norm. Police corrupt to the core and paid to turn a blind eye to the everyday goings on, nowhere to turn for help in a lawless society where the only way to live is go RENEGADE.

Your girlfriend calls and asks if you can meet her at her local bar, it's on the other side of town, a twenty minute journey on the tube. You accept to meet her unaware of the danger and pain that lies ahead.

You head off in the direction of the tube station, as you walk downwards into the underground subway you hear a loud scream followed by a gunshot! You rush to aid of whoever's been hurt but you quickly stop. All around you piercing eyes firmly stare, Kill him they shout. Look sharp think fast their coming for you.

Night swiftly falls and the chase is still on, Tired from exhaustion and unable to run any more you come to a dead end and can see no escape. Barbed wired fences stand 7 feet tall all around you and the only way out is to climb. You attempt to climb but it's no use the barbed wire is just cutting you to shreds. You decide to make a stance, no more running or hiding striving to stay alive, just fight!.

Renegade pits you against five levels of pumped to the max game play and is one of the best gore games ever released.

You start in a subway station where you have to beat the living crap out of everything that moves this includes skinheads and people who can't afford to get there hair cut. Punk styled gangs are a plenty and so are hells angels accompanied with motor bikes.

This level is perfect for perfecting all the moves Ocean and Imagine have kindly thrown in on the Amstrad version. Head butts, back kicks, flying kicks are all there but the one that reins supreme is the pummel. Jump on an opponent whilst his down and punch the living crap outta him, cool stuff eh.

Other Formats

Commodore C64/128 (1987)

Spectrum (1987)

Each level produces magical moments and on later levels you can pick up a wide range of weapons.

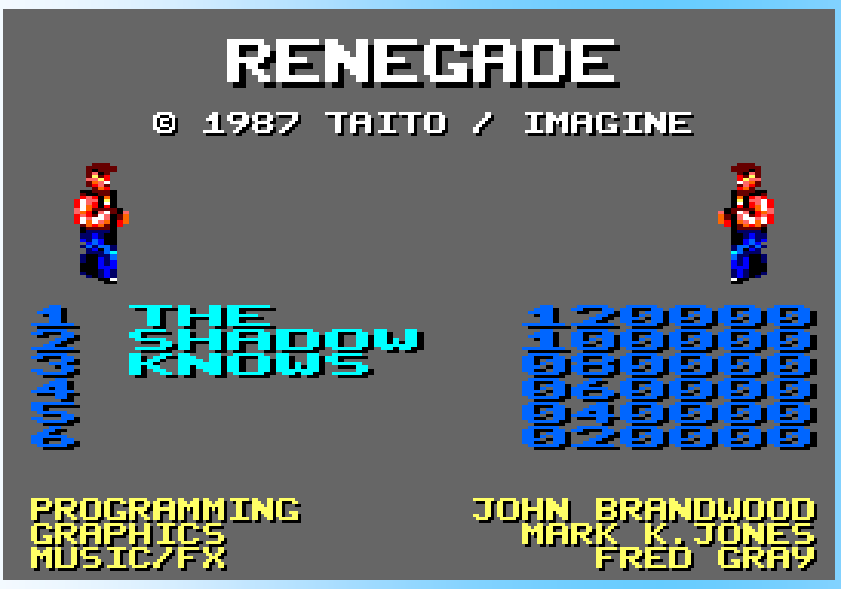
Weapons are especially needed on later levels when you go one on ten with big Bertha and her henchmen who love to take the odd pot shot whilst your slugging it out with old Bertha.

The action is none stop and for those of you that have never played the game before there's a special surprise ending awaiting you upon completion.

Graphics and sound are top notch and the backgrounds will amaze Amstrad lovers all over, the graphics look crisp and the whole game runs as smoothly as can be using nearly every colour the ye old but faithful CPC can muster. If there is to be a downside it's down to the way the screen scrolls, it doesn't basically it flicks.

One of the Amstrad all time greats that plays even better than the arcade version, I should know I've played them both to death.

All in all …….
SSSSSUUUUUUUPPPPP...
PEEEEEERRRRRRBBBBBBB…



CPCO RATING

Graphics	9
Sound	9
Gameplay	9
Lastability	7
Overall	9

switchblade

Reviewed by Mark Hall

To inflict death and misery is but a game to Havoc. Women, children he has no preference, some say they are easier prey. The strongest champions from all across the land have battled gallantly if only for a few minutes against the demon of the underworld. One by one have fallen until none were left.

Most Family's have been all but wiped out. Those were the lucky ones. Poverty and famine has reached out all across the land. And the ones that have survived thus far have done so out of revenge and anger.

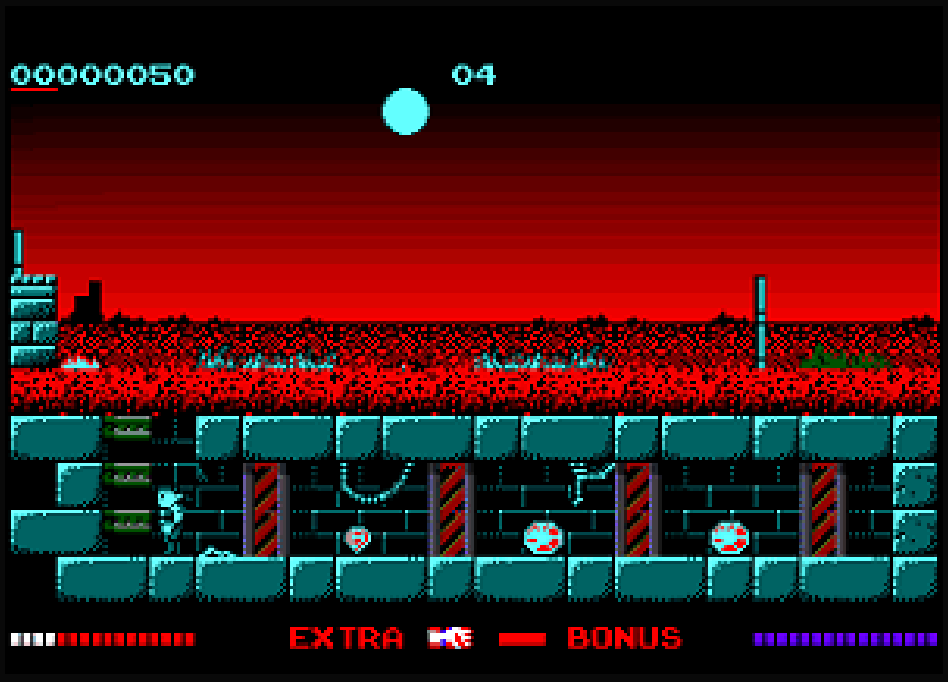
The once great protectors known as the Blade Nights are all but dead, killed by Havoc's own hands, and his army of un-dead soldiers.

The Blade-nights most powerful weapon named the Fire-blade was stolen by one of their own in an attempt to let Havoc wield its power, in exchange for his life. The blade-nights quickly acted and with all their power and might sent the Fire-blade scattering to all four corners of the globe.

Havoc angered by the news ordered an all out attack. The Blade-nights fought bravely, but without the fire-blade were no match for Havoc and his army's.

20 years on. Havoc still rules the land, a tyrant with no remorse. But wait a hero of the people emerges, hidden by the blade-nights and trained in all sorts of combat, his mission to recover the missing fragments of the Fire-blade.

The Fire-blade is secretly hidden across the globe, its whereabouts known only by Hiro the last of the Blade-nights. Once Hiro finds the Fire-blade only then will he stand a chance of dethroning Havoc. And avenge his people's death.

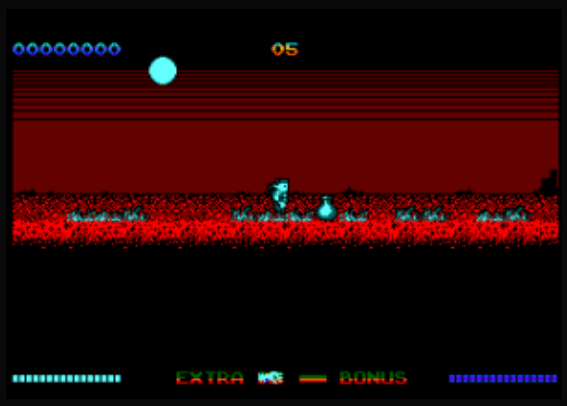


Other Formats

Commodore C64/128 (1989)

Commodore Amiga (1989)

GX4000 / PLUS VERSION - NOTICE THE EXTRA FEW COLOURS MINUS THE MULTICOLOUR CHARACTERS TRICK



ORIGINAL CPC VERSION - LESS COLOUR DETAIL BUT STILL JUST AS FANTASTIC TO PLAY

Our last chance is Hiro……

You start off above the surface, our hero standing proud and ready for all out war. The first thing you'll notice is the stupendous detail that has been put into the sky and surrounding graphics. The use of colour is used sparingly, until you look at the colour palette used just for the sky alone. After a quick scuffle on the surface and a few well placed round house kicks, its underground we venture. This is where everything really gets going and our Hiro finally gets to kick some ass.

The further you descent the more magical the game becomes. Rooms appear that need to be explored. Some are small and can be easily navigated, but the majority are huge. Each location contains traps and falling objects, not to mention a more powerful enemy each time.

Once the bad guy in each room has been defeated its time to collect the vast range of power-ups each room kindly supplies. Oh and the fragments of the Fire-blade itself.

The music plays throughout the game and whilst nothing special is durable, as are sound FX. The graphics are superb, although you get the feeling this game isn't even pushing the GX4000. It's identical to the disk based version but with a few more colours.

The game is huge and only the most skilful and patient will persevere. And finally hammer the last nail in Havoc's coffin. The game is littered with hidden areas and to get to them often involves kicking big holes through a wall, or jumping over or crawling under traps. All in all, probably one of the best action adventure games the Amstrad has ever seen. This game proves that Gremlin graphics really do live up to their name.

CPCO RATING

Graphics	8
Sound	7
Gameplay	9
Lastability	9
Overall	9

CONTINENTAL CIRCUS

Reviewed by Mark Hall

Ever fancied racing F1 cars at breakneck speeds all around the world, if so make this gem your next pit stop.

Fast cars, fast women, and miles of tarmac, Sounds like the perfect life I here you say. Well unfortunately I'm talking virtually here.

I remember first catching a glimpse of Continental Circus on an Amiga. I quickly fell in love with the game and spent the best part of a week constantly playing the thing round my mate's house. Until his mum kicked me out.

I heard that a release was imminent for the CPC, and asked myself if the Amstrad version could play and feel as good as its 16-bit counterpart. The answer is a resounding yes! Probably one of the best conversions the Amstrad had ever witnessed. The game gave a big shout out to all the other 8-bit owners and console freaks, that Amstrad was alive and kicking and a force to be reckoned with.

Continental Circus on the CPC left all other 8-bit conversions stalled on the line. The sheer speed everything moved at was nothing short of genius. Graphics were big and well drawn.

UP IN SMOKE!

Other Formats

Commodore C64/128 (1989)

Arcade (1987 US / 1989 World)

And nearly every colour the Amstrad had available was used to full effect. The road moved smoother than a baby's arse. And the control system felt just right.

The game starts you way at the back of the grid. And your job being an obvious one is to progress through the ranks in one of the toughest tournaments on earth. Don't let this put you off as there is a continue system and the difficulty level makes for a dam fine game with all the realism you could wish for. All this makes you think about speed needed when approaching a corner, overtaking and pit stops.

The sound in the game is top notch. The roar of the engine sounds authentic. And the swishing noise as you overtake another car is super. The music has also survived the transition from the Amiga and plays different tunes before the start of each race.

With a super smooth race track, graphics to die for, and a race car that goes like a bat out of hell. It all adds up to one of the best Amstrad racers of all time.

A QUICK PIT STOP TO PUT OUT THAT FIRE!

CPCO RATING

Graphics	8
Sound	7
Gameplay	9
Lastability	9
Overall	9.5

THE START OF THE AMERICAN GP

RAIN POURS FROM THE SKY!

RICK DANGEROUS

Reviewed by Mark Hall

If ever a game deserved the Indiana Jones licence stamped all over it, Rick dangerous would be the one. From the moment you load the thing up. And to the rare occasion you put the thing down. The game just looks and feels like an Indy outing.

It's always amazed me that an 8-bit version of the classic Raiders of the lost Ark never made it to the CPC. Good or bad it would have sold shed loads. On the plus side, if it had been released then this is how it should have been minus the Germans.
So dust down that lever jacket, grab your cowboy hat and whip. And lets join Rick on his new adventure on the CPC.

Our hero start his adventure over Amazon in search of an ancient lost tribe, called the Goolu. Without warning the plane nose dives and crash lands into the jungle. Rick stumbles around in an attempt to clear his head. All his colleges are dead and his equipment broken. No radio or food, just a pistol and some dynamite.

Rick gathers all he can hold and jumps from the plane he feels as though he's being watched by hundreds of piercing eyes and quickly wishes he'd stayed inside the aircraft. He makes a run for it and hundreds of wild Goolu's pursue him. Rick falls down the side of a mud hill for what seems like a lifetime and seems to have escaped. Just as Rick thinks it's safe a giant bolder comes hurtling towards him, oh boy…….

You start the game running from a giant bolder. You then try to escape by falling off the edges of steep rock faces. Once you successfully escape the bolder the action continues with you in some kind of underworld labyrinth. Go Indy.

trying to give up smoking. Bad idea I can tell you, 10,000 fags, and crisp packets later I completed the game and felt like s**t. You really have to use your noodle on later levels. Traps are cunningly layered throughout the Egyptian stage. Careful navigation is needed to stay alive and you have to be as cunning as the programmers were whilst putting the game together. This stage will take you many an hour to complete.

You must successfully navigate your way through the Amazon undergrowth. And either use your trusty stick, gun or dynamite to help you proceed to the next level. All weapons can be used to either open doors press buttons and generally cause all sorts of havoc.

The game is platform based, and features five stages. These are Amazon, Pyramids, Castle, missile base, and future. The whole thing runs smoothly and the graphics are top, probably as good in places as the Atari ST.

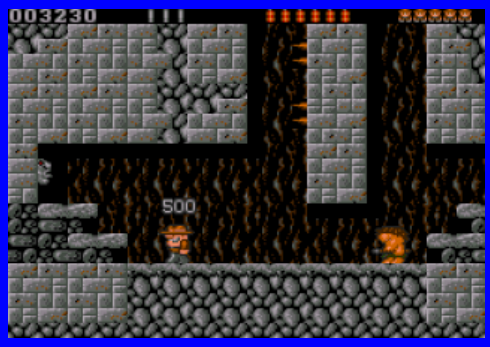
There is a minor problem though. The game is very difficult and can frustrate the not so patient amongst us. I played this game whilst I was

The Castle stage seems easy at first until you reach the third screen that is. Then you realise it's gonna be a long night. And hope that the power doesn't go like it did whilst I was on this level. I was nearly at the end as well.

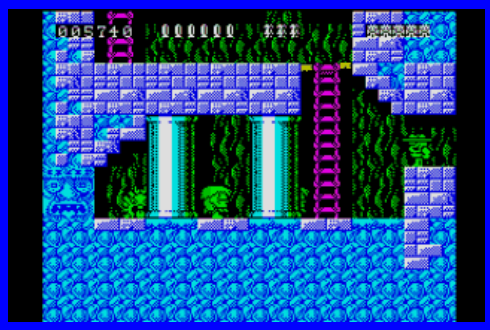
Despite the later, the game is a diamond and will keep you at it until you finally complete it. Probably one of the best platform adventure type games around for the Amstrad CPC. Don't play this game if you have a heart condition though. A game guaranteed to make you want to take up smoking.

Other Formats

Commodore 64 (1989)



Commodore Amiga (1989)

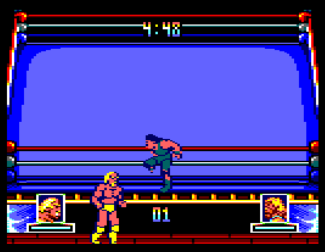


Sinclair Spectrum (1989)

CPCO RATING

Graphics	9
Sound	7
Gameplay	9
Lastability	9
Overall	9.5

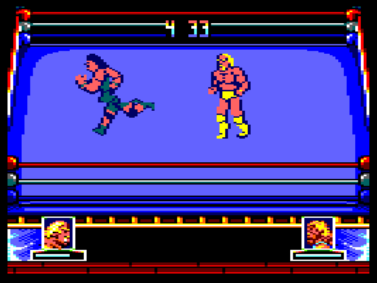
WWF WRESTLE



IN 1984 HULKAMANIA WAS BORN. NOW IT'S STRONGER THEN EVER AND I FEEL SORRY FOR YOU, NO PERFECT, WHEN THE HULKAMANIACS GET HOLD OF YOU DUDE!

Reviewed by Mark Hall

The World Wrestling federation is coming to town and Hulkamania is gonna run wild all over your CPC. The Amstrad finally gets a wrestling game featuring some of the toughest men on the planet with attitude an all. These guys are mean and put the 'F' in the word fear. I you smell what the Rock is cooking... 'Sorry this game is set in the 90's'



On starting Wrestle Mania up for the first time you will be pleased to see that all the flashy lights and music have been included along with a picture of the wrestling ring before each match. The WWF super stars even have a go at each other before the match starts, this sets the scene nicely as you prepare with joystick in hand to suplex the buggers. The presentation and in game graphics are very good, all the superstars look as good on an 8-bit as they do in real life if you know what I mean.

The game starts with an opening sequence and a brief history of each wrestler, this includes statistics and favourite quotes from the three superstars on offer. You then press the return button to acknowledge your choice of wrestler. As soon as you accept your wrestler the music of the WWF bellows out from your speakers, in the traditional sometimes over the top World Wrestling federation manner.

The controls for your wrestler are superb and easy to master, all the signature moves have been included and are easy to perform, once you get the hang of the joystick. If you are to win the bout then it's time to get mean and do the thing that wrestlers do best 'What do you mean talk' beating the living crap out of your fellow wrestler. And if your lucky you might get a 1,2,3 count.

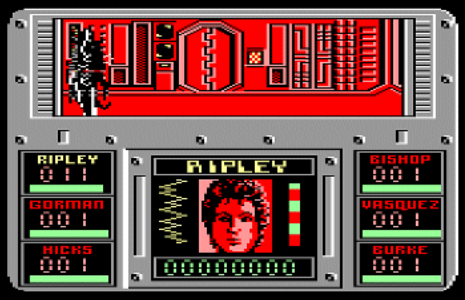
This is a top game of the WWF and oozes with quality, but there is a problem! the selection of wrestlers on show is limited to three. And the whole game can be completed in less than an hour, this disturbs me somewhat as the game cost £14.95 on disk, come on Ocean the disk is only half full. The thing that saves this game is the inclusion of a two player game, you and your mates can kick the crap out of each other long into the early hours of the morning.

The only WWF game to grace the disk drive of your CPC, a first outing Ocean can be proud of. lets hope the creative juices at ocean pour a little more into the next one with the profits they make from this.

CPCO RATING

Graphics	8
Sound	5
Gameplay	8
Lastability	6
Overall	7

ALIENS



Reviewed by Mark Hall

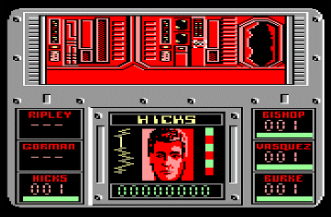
While everyone else as a kid was playing Rally racing games and Outrun, I was wondering through dark Alien infested corridors, a flame thrower in one hand and a half completed map in the other.

Oh and a packet of cheese and onion crisps at my side. I was locked in a world full of Aliens and spaceships 'nearly every sci-fi lovers dream' but instead I was scared shitless.

Aliens the computer game takes you into confrontation with the most ruthless creatures ever to have appeared on a cinema screen. And now I had them crawling inside my CPC. Picture the scene, speakers turned up loud and the lights dimmed, but not overly, as I needed to see the map. With eerie music, the sound effects of the Aliens getting ever closer.

This was one of those games where you really had to use your imagination and map building skills to the max. Imagine trying to play Half Life with a side scrolling engine whilst in a first person view. Times were hard for us gamers back then.

In the less graphical days of computer games as you all know and love, more emphasis had to be placed on the player's imagination, due to the graphical restrictions of that era. This is why I had to review this game. It's one of those games that run's wild with your imagination. Everyone I ever speak to about the old days and computer games nearly always mentions this game with fond memories.



The game starts similar to the film, darkness descends all around. Alien Warriors enter the base where you and your companions are waiting. Nervously flicking the safety catches off your Smart guns, you advance... The unforgiving physics and controls that usually turn more traditional gamers off?, as it turns out, is a resounding, "No."

The control system once mastered is a breath of fresh air, and the way you can send a crew member up to 9 rooms at once is an added bonus. The graphics throughout the game do there job as does the sound. The only gripe I have is with the flickering scrolling, but this does nothing to deter you from playing the game right through to the end.

The control of the smart gun and marine movement is fantastic for its time and allows you to move the cross air all over the screen whilst moving left or right, this allows you to run and shoot at the Aliens whilst they are pursuing you, great tactics. Commanding the crew members is relatively easy, a simple key press that represents their last name will see you giving all units orders. And enable strategic deployment of troops.
The game will have you on the edge of your seat throughout and will last far longer than the film did.

As you creep around corridors you will notice the Newt, the little girl that appeared in the film. And if good enough, a showdown in the Queens chamber with yours truly. All in all one of the best action/adventure/thriller Sci-fi games I have ever played.

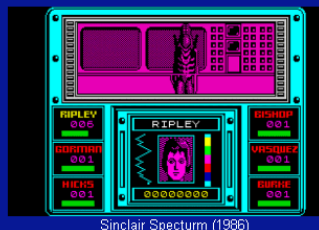
ALIENS
PRESENTED BY
Electric Dreams

CPCO RATING

Graphics	8
Sound	9
Gameplay	9
Lastability	8
Overall	8.5

Other Formats


Commodore C64/128 (1986)


Sinclair Specturm (1986)

5 GAMES THAT COULD WORK ON THE CPC

By Patrick Furlong

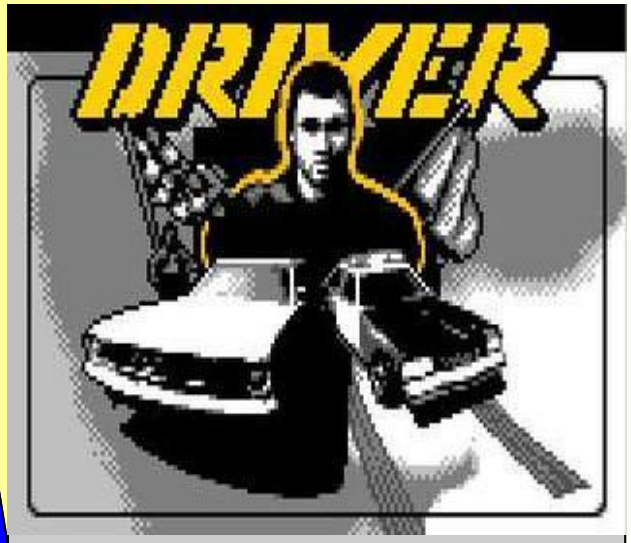
Throughout my 20 years of using computers, the CPC has remained my favourite of the home computers. While my days of using the real thing have long gone, I still use the emulators and when I see a particular game, I think "Someone should be able to make a CPC version!".

In no particular order, here are 5 games which could survive the transfer from new machine to old machine.

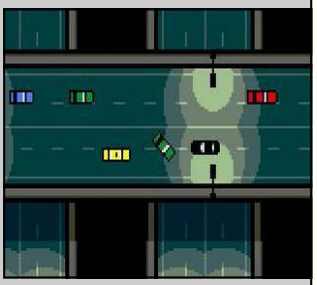
Harry Potter and the Philosopher's Stone / Harry Potter and the Chamber of Secrets

If this was done with MODE 0 for the graphics and MODE 1 or 2 for the text, a conversion from the Gameboy Colour version would look really good. A version of the Gameboy Advance version of Philosopher's Stone would really be pushing things for the 128k machines so a 256k expansion would be needed.

Nothing state of the art was required for the Gameboy Colour version so as I said before, a CPC version would look really good. It might need the use of memory bank switching and excessive multi-loads. It would be easy because of the overhead views.



Driver



Again, a conversion from the Gameboy Colour version would be a success. There have been lots of overhead view car games on the CPC in the past like Super Sprint and Grand Prix Simulator, so it would be simple for the CPC to handle this. Due to the size of the levels, it would be a 128k only game.

Synthised speech is a must for when the cops radio their back-up to say if the player has run a red light or hit another vehicle.

The Lord of the Rings

Someone could do a lot better than the bugged text adventures that were released in the mid 1980's. A 128k conversion of the Gameboy Advance games of The Fellowship of the Ring and The Two Towers could be very good!

Where Time Stood Still

One of my favourites on the Spectrum and Atari ST – oh, why was this NOT released on the CPC. The 6128 could have done a really good job on this game and there might have even been some colour!

I first read about this game in Sinclair User and thought it was really good. I managed to find an original copy, but problems with my Plus3 meant being able to load the game was rare until I got hold of an emulator (it's the game I play the most on my Speccy emulator). I looked for references to it in CPC magazines to see if one was released – I saw a reference to it in Amscene in AA26 or AA27 but when I saw adverts, it only mentioned Spectrum 128, Atari ST and IBM/Amstrad PC.

I would have killed to get a copy of this game for the CPC, but sadly, it was not to be. But this game is one I would LOVE to see on there one day.

Star Wars: Episode 2 – Attack of the Clones

Released only on the Gameboy Advance, this is a simple 2D platformer with the occasional 3D driving level – I admit I only played to level 4 before getting really bored and closing down the emulator.

Surely it wouldn't be hard for a 128k game to be made of this game, maybe even 64k if people wanted to push their luck a little.

END

5 GAMES THAT SHOULD HAVE BEEN SENT BACK TO THE DRAWING BOARD!

By Patrick Furlong

Throughout nearly 20 years of using a CPC, the most using a CPC464, then 3 years using a CPC6128 and the rest using emulators, I have seen some great games, some good games and some games so bad, I would have sent the designers back to the drawing board to start again! This article reveals five games I would have sent back to the drawing board.

The Star Wars Trilogy

I'm counting this as one game. When they first came out, I really liked them, but over the years I grew to dislike them, especially after playing better versions on the Nintendo consoles.

The Nintendo NES and Gameboy share the same Z80 processors as the CPC, so a CPC version of these games would be possible. Granted, Domark got the licence to make games based on the arcade machines, but I feel that if they got a different licence to make games not based on the arcade machines, they could have done well. They did very well with their adaptations of the James Bond films Licence to Kill and The Spy Who Loved Me.

Crystal Kingdom Dizzy

This is the only Dizzy game that was a MAJOR let-down. This was the last one I was able to play, borrowing it from a college friend in 1995 but when I loaded the game and played it, I felt very let down.

Why couldn't they have made it in the style of the previous Dizzy games. They could have kept the level system – it worked well for Wild West Seymour, but why did they have to use the 16 colour mode when 4 colours worked well for the previous games.

Doctor Doom's Revenge

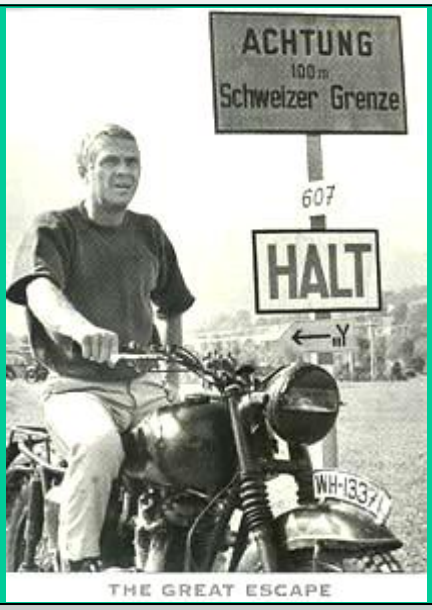
Being a Marvel fan, especially Spider-man, I looked forward to this game. While the plot and graphics were brilliant, the use of an excessive multi-load ruined the game, bearing in mind, I didn't have a disc drive or 128k until 1994.

I would have sent them back to the drawing board to make a better game which didn't need to use excessive multi-loads. This is what ruined some games for me.

The Great Escape

Next time a game company wants to use the name The Great Escape for a game, make sure it's more like the famous 1963 Steve McQueen film (or prefeablly, the 1952 Paul Brickhill book) and not based on your own storylines! (oh wait, SCI have finished making one like the film!)

The graphics engine and interaction were brilliant, later being used in the Spectrum/Atari ST only game Where Time Stood Still (see a different article for my moans about this!). Instead of setting it in a Colditz type environment, why not set it in open ground, like the real Stalag Luft 3?



THE GREAT ESCAPE

Keep the graphics, but change the interaction (more like Where Time Stood Still) and change the locations and add more problems and you will get a better game.


SPECTRUM SCREENSHOT

All the best Ocean GX4000 games

I never had a GX4000 or Plus machine so I couldn't play Ocean's top games like Robocop 2, Navy Seals and Plotting. On studying the games when playing them on emulators, they don't look like they are taking advantage of the new Plus features.

Empire and Gremlin both made Plus and normal CPC versions of Gazza 2 and Switchblade respectively with very little changes, so why couldn't Ocean have made standard tape/disc versions for the normal CPCs? This is the last game or set of games I would send back to the drawing board.

Famous Five

Five on a Treasure Island

Published by Enigma Variations in 1991 - Price £11.99 (tape), £14.99 (disc)

For over 60 years, people have read and enjoyed the Famous Five books written by Enid Blyton, including myself. in 1991, Enigma Variations brought out an adventure game based on the first book - Five on a Treasure Island.

This game is very faithful to the events in the book although the chronology is way off and the arrival to Kirrin is different to the book.

You can choose whatever character to be a anytime and you can also send a character to a room. Except for certain bits, all the other characters follow you.

For the disc version, it has very nice graphics which are in Mode 0. The game also makes use of RAMSAVE and RAMLOAD. You can talk to all the characters in the game although it is not recommended you try to talk to Uncle Quentin anymore than once because he can send you back home.

Until you make friends with George (I won't tell you how), she has a mind of her own and will not stay with the main group. Some characters are best suited to certain situations. Some parts of the game are easy to solve if you read the book and there are many ways to end the game, if you annoy Uncle Quentin enough times or if you catch a nasty cold to name two examples.

The parser is very good and you can get a complete list of verbs and nouns at any time should you ever need them. There is even a little in-joke but I'll leave that for you to find out.

I would recommend this game to anyone. However, unless you can put the game onto a real CPC machine, I would either download a Spectrum or Atari ST emulator and the respective versions of the game because due to a bug, it is impossible to complete the game using a disc image of the disc version of the game, although I'm told the disc image of the tape version works well but if you only play the tape version, you will miss out on the nice graphics.

The lack of graphics on the tape version concerned me - if the programmer had pushed things a little or even included a 128k feature, he might have been able to include graphics. The Sinclair Spectrum version, programmed by the same person featured graphics on the 128k tape version.

This is a very good game and is recommended to fans or even non-fans of the Famous Five - in fact, I got into the books after playing this game when Amstrad Action put it on their covertape in August 1992.

Overall Rating: 90%

SUBSUNK

Published by Firebird Software in 1986 Price: £1.99

Subsunk is a graphic adventure created using the Quill adventure creator. I first played this game in 1987, and myself and my family enjoyed playing it, but were unable to finish the game.

The graphics were created using text characters instead of using the Illustrator graphic creator for the Quill, but they work really well.

The plot is simple - you play Ed Lines, a reporter who is reporting on life onboard a British submarine when the craft is attacked by an enemy power. You hide as the enemy board and take the crew of the submarine prisoner. Then you find out that the submarine is on the bottom of the ocean and you have to find a way to get a subsunk message to either the nearest British ship or British Naval Command, but this is not easy - you start the game with a splitting headache and unable to concentrate on a few things. The first thing to do is to find a way to get a key and to make a medicine to cure your headache.

The parser in this game is very poor, with the game being very fussy about what you type to achieve some things. However, some objects will give you clues on how to achieve a few things necessary to complete the game. Another thing that lets the game down is that if you are wearing something, you cannot remove it, reducing the number of items you can carry.

You have to make things like glue, wear aftershave and a few other things. It pays to pay attention to stuff you read because they can give very subtle hints to solve certain problems.

Unlike many games, you get a very good end of game message which leads into the sequel - Seabase Delta.

This is a good adventure, which is let down by the very poor parser. Find this game for your emulator and play it.

Overall: 76%

THE COMPUTER

From 1986, the Eagle comic, home to many popular comic strips like Dan Dare, Doomlord, The Thirteenth Floor and Charlie's War, started a strip entitled Computer Warrior.

This strip featured teenagers who discovered the secret code to access the real life sections of their computers go they could play the games they were using for real. This area was dominated by a being known as the Warlord and the area was under threat from the Dark Forces who threatened to attack the real world.

A large group were to play games for real and the people who lost were sent to the Nightmare Zone where they were subjected to eternal torment. In each game, losers were sent to the Nightmare Zone until there was only one player left – Bobby Patterson who then had to face the Dark Forces' champions in a series of five games to banish them, earn the title Ultimate Champion and free his friends from the Nightmare Zone.

He won all five tests, sometimes when he was just moments from losing. He thought he would be free of this domain forever, but he and his best friend were taken back into the computer for one last game. Three Dark Forces members had escaped into one of the game zones, and Bobby Patterson and his friend had to go into the game zone and capture all three. This mission was successful.

The third chapter of this story began when Eagle advertised a chance for readers to appear in the strips. It started when Bobby brought a game, liked it so much and decided to play it for real. However, instead of the game, he was taken to the Warlord's chamber where he was dying and the Dark Forces about to escape from their prison. Before Bobby could go home, the Warlord requested him to take his place and died. So unless Bobby wore the Warlord's robes and took over, he was stranded and the Dark Forces could escape. He made a decision and became the new Warlord. He crushed the Dark Forces' prison and managed to return home.

He decided that as the Warlord, he could let other people play the games and send them home afterwards, even if they lost. The first contestant won his challenge, but the second contender lost. Before Warlord Bobby could send him home, Dark Forces took the contender to the Nightmare Zone. Bobby therefore set a challenge – his champions (Eagle readers) against the Dark Forces' champions – first to five points would win. If the Eagle champions won, the contender would be released and if the Dark Forces won, Bobby would go to the Nightmare Zone.

The gauntlet was set. However, only two games were played and won by the Eagle champions before greater forces intervened.

Bobby was summoned to the Chamber of Warlords who told him that he was no longer a Warlord and the contender was free. Bobby himself thought he was free, but they had other plans. Warlord Baal, set him on the path to be a Computer Warrior again and sent him to play another game. Bobby deliberately lost a few lives before they showed him the punishment for failure – a fate worst then the Nightmare Zone – the Scream Machine! Bobby agreed to finish the game, but he lost. However, he was not sent to the Scream Machine. It was a test and he passed. The Warlord said that he would need to complete ten tests to be free of it forever and the Dark Forces would be defeated. He reluctantly agreed.

WARRIOR

Written by Patrick Furlong

The first tests were completed and won. A few times, he would be reminded of the penalty of failure and once tried to rebel by attempting to destroy the computer, but Warlord Baal sent him to the Scream Machine – he escaped the fate by promising to complete the challenges.

The tests were interrupted when servants of the Warlords came to Bobby and told him that Warlord Baal had been taken prisoner by the Dark Forces in the Nightmare Zone and he had to go and rescue him. This was a successful mission, only just so the tests continued.

The tests were never completed – Eagle comic soon ended either during or after one of the tests and we never found out what happened next.

With the exception of a handful of games, the games were based on those released by US Gold – apart from their original games, it also included licensed from Sega, Capcom and others. Bobby's games were played on the Commodore 64 machine and the scenarios, number of levels and game play were different.

The first tests began with a group of people, but towards the end of them, only two were left and the winner went on to face the Dark Forces champions to become the Ultimate Champion.

The first tests included: Zylon Attack, Ghostbusters, Express Raider, Metrocross, Ace of Aces, Dream Warrior, Side Arms, Bride of Dracula (need confirmation on this) and Bionic Commando.

Ultimate Champion Tests

Road Blasters

Eagle had already featured a comic strip of Road Blasters, based on the Mattel toys which showed two kids racing in two cars belonging to the Turbo Forces and Motor Lords which were controlled by a satellite in orbit around the planet.

Bobby was sent to this satellite and had to race the Dark Forces' Champion – Skullgrin from the satellite. However, when Skullgrin got into a rage, he smashed the controls so they had to go to the planet to race. It looked for a time that Bobby would lose this game, when his car was badly damaged. But he was able to use another car belonging to his side to win.

Thunder Blade

This is a firm example of changes. In the game, there are four levels, but only three on here. In the C64 version, you only get three lives – in this game, Bobby got five lives. The end of level guardian for level two was changed from the desert tank to the fortress from level four.

The Dark Forces' champion was General Gor, who Bobby had to recapture

later on in a different test. This game was so difficult, Bobby lost four lives before the end.

The test ended when Bobby fired a missile at the airship at point blank range and though Thunder Blade had been blown up, but the airship was destroyed instead.

The Deep

This isn't a big name game, but was chosen to be the third test. This story was much like the game, the only differences were that Bobby only had one life and there was only one stage.

Bobby found himself Captain of a destroyer against the Dark Forces' Champion – Admiral Two-Face. One face would be friendly and full of charm while the other was full of evil.

Bobby nearly lost when the ship was torpedoed but the damage was minor. In level three, when he had to rescue the hostages, they were his fellow Computer Warriors who he was playing for.

When it was revealed that they had been told that if they were sunk, they would die for real, Bobby summoned the Warlord and they had a confrontation. Bobby had been told that if he won his friends would be released.

He gave the Warlord an ultimatum – free one of his friends to test his word or Bobby would quit.

The Warlord released one of his friends, well actually, it was his enemy Gummer, who cheated his way to be in the last two.

Zak McKracken and the Alien Mind-benders

This game was only released on the Commodore 64, Atari ST, Amiga and PC so is a little known game to readers of this magazine. In the game, aliens were using the phone system to turn Earth's population into morons.

I can't remember who the Dark Force's Champion was in this game, but he was in control of the phone network which was turning people into morons.

The game didn't start with the Warlord taking Bobby into the computer, but started after a meeting in his bedroom.

He realised this when he found his parents acting like children.

Using a fish tank and socks, he was able to transport himself to various locations – a toy store, Stonehenge and a busy city street where the Phone Exchange was.

Forgotten Worlds

The final test and it came at a bad time. Bobby had entered a competition to win a holiday to Los Angeles and all he had to do was win at a game called Chiller

Diller – a fictional game created by the comic strip writers. Before he could enter the semi-finals, the Warlord took him to play the final test.

The Dark Force's Champion was an evil mirror image of Bobby himself who went out of his way to cheat to try to beat Bobby, but he ended up getting blasted.

The game was one huge level with no end of level monsters. Bobby won the game and freed his friends, but their memories had been wiped.

Thunderblade – The Hunt for the Generals

That is my title, by the way. Bobby had changed so much since winning and freeing his friends. His best friend was having a sleepover at his house when he and Bobby were taken into the computer. General Gor (from the Thunderblade test) had escaped and with his two brothers, were hiding in the Thunderblade game zone. Bobby and his friend Martin had to capture them. This time, they were only given one life.

Only two of the levels were shown in this game. One of the Generals was in the gunship which ends level one, I can't remember where the second was hiding – maybe he was in a helicopter scene somewhere in Level 1, while General Gor was hiding in the fortress which ends level 4 in the actual game, but level 2 in the comic strip. This mission was completed successfully.

Indiana Jones and the Last Crusade

Bobby had brought this game and entered the computer's real life zone to play it. But he was brought to the Warlord's keep by the dying Warlord. Needless to say, Bobby took over the Warlord's job and was able to let Eagle readers play the game.

Eagle reader Andrew Porter was chosen to play this game. It was generally the same as the home comp-

-uter game and level four even started with a simple Sean Connery look-a-like!

Porter won the game and his prize was a copy of the game and the chance to call himself Computer Warrior.

Sleigh Ride

This was on of Eagle's fictional games. Bobby had brought the game as a present for someone and decided it would make an ideal prize for a lucky Eagle reader.

Reader Dave Dawson was chosen to play the game. He was given three lives and he had to deliver presents. However, there were goblins to stop him.

He had a bad start, losing two lives on the first few houses, but he improved, however, a goblin fired rockets at him and made him crash, losing his last life. He was taken back to the Warlord's keep where Bobby was about to send him home but the Dark Forces came and took Dawson to the Nightmare Zone. A challenge was issued to secure Dawson's release.

Turbo Outrun

I can't remember the name of the Eagle reader chosen to play this game, but there were four stages and it was like the game. After each stage, the reader could choose an extra add-on.

The Dark Forces' Champion – as usual, also went out of his way to cheat. In stage 2, the reader chose High-Power engine, stage 3, which was set in Texas, he chose Special Turbo, with which he torched a police car and for the final level, he chose Hi-Grip tyres.

He was able to win the game with plenty of time.

Ghouls N' Ghosts

Again, I can't remember the name of the person who played this game, but it was one huge level – level one.

Like Dave Dawson before him, he had a bad start but managed to make his way to the end of the game and was able to beat the monster before he lost his last life.

Black Tiger

Bobby was brought before the Chamber of Warlords to be told he was no longer a Warlord and Dave Dawson has been freed. However, before Bobby could celebrate, Warlord Baal told him he was to be a Computer Warrior again and set him on the Black Tiger game.

Because Bobby didn't want to play the game, he deliberately lost three lives before he was sent to the Scream Machine, a machine of torture which would be used on him if he lost. He promised to finish the game so he got the end of the game, losing a life on the way. He had one life left when he went against the Samurai Dragon. He unfortunately lost. Baal told him it was a test and he had passed, so he wouldn't go to the Scream Machine this time, but if he failed again, he would.

Crack Down

Warlord Baal summoned Bobby as he was about to face his worst nightmare – a visit to his aunt. Bobby had three lives and three levels to complete. The first level is in a compound, but he lost his first life after being ambushed. In the second level, he encountered enemies which were bullet proof, but was able to counter them. He managed to plant his explosive charge but an eruption before he was able to escape meant he lost his second life.

Bobby managed to escape to level 3, the main base of the enemy. He was able to fight his way through to the main villain (Dr X, I think), but was pinned down by a robot. He had to use an explosive charge to get rid of X and win the test.

Dynasty Wars

This test started in an unusual way. Bobby was in bed, when he was taken to the game area where he was ambushed by Chinese warriors before being taken to Baal's keep where he was told the game plot and given the usual warning about the Scream Machine.

I don't remember anything else about this game, so cannot tell you any more information.

UN Squadron

Based on the Capcom arcade game, Bobby plays Greg Gates, who flies the A10 Thunderbolt. The levels are faithful to the game and he flies with a wingman, who is killed halfway through the game.

He is joined by another wingman who completed one of the levels. With this wingman, there are rivalries. The new wingman even shoots Bobby down in one level, but he gets his revenge later on.

Like in the game, there are sections where the pilots are briefed and new weapons are shown by the clumsy professor (who drops a missile, which is only a demonstration dummy).

As usual, Bobby wins the game.

Time Warrior

This is one of Eagle's fictional games. Bobby has to complete four (or five) time zones. The first time zone is set far into the future and to win this, he has to defeat a big blob.

The next level is set in 1920s Chicago where Bobby has to capture a Mob boss. This and the next level were the best in this game. The next level sees him as a coach driver in the Wild West. The aim of this level was to get the coach and it's passengers to the nearest city, while under attack by outlaws. Bobby not only gets to the town, he also takes the chief outlaw prisoner.

The final level is set in the future. A power mad dictator plans to fire missiles from the sea to destroy major locations. Bobby wins by sending his submarine into the missile base, ejecting, and risking losing his last life, but is saved by the Warlord.

ESWAT

I don't remember much about this game, but it had four levels and made use of the power armour which was available to the player after completing three levels, Bobby got this after completing level 1.

Mercs

I missed all but one of the Eagles this came in and I can't remember anything about it.

Nightmare Zone

This wasn't a test of sorts. Two servants of the Warlords came to Bobby one night and told him that Warlord Baal had been taken into the Nightmare Zone and he had to rescue him.

I also missed most Eagles with this one, but I remember a few things about it.

Bobby was captured by the Dark Forces and taken into a big castle where Baal was being held prisoner.

After a fight, he kills who he thinks is the big boss man, but is revealed to be Baal in a mask.

After more fights, Baal saves Bobby. It is revealed that he used his powers to switch identities with a servant of the Dark Forces, who was killed by Bobby instead.

Street Fighter 2

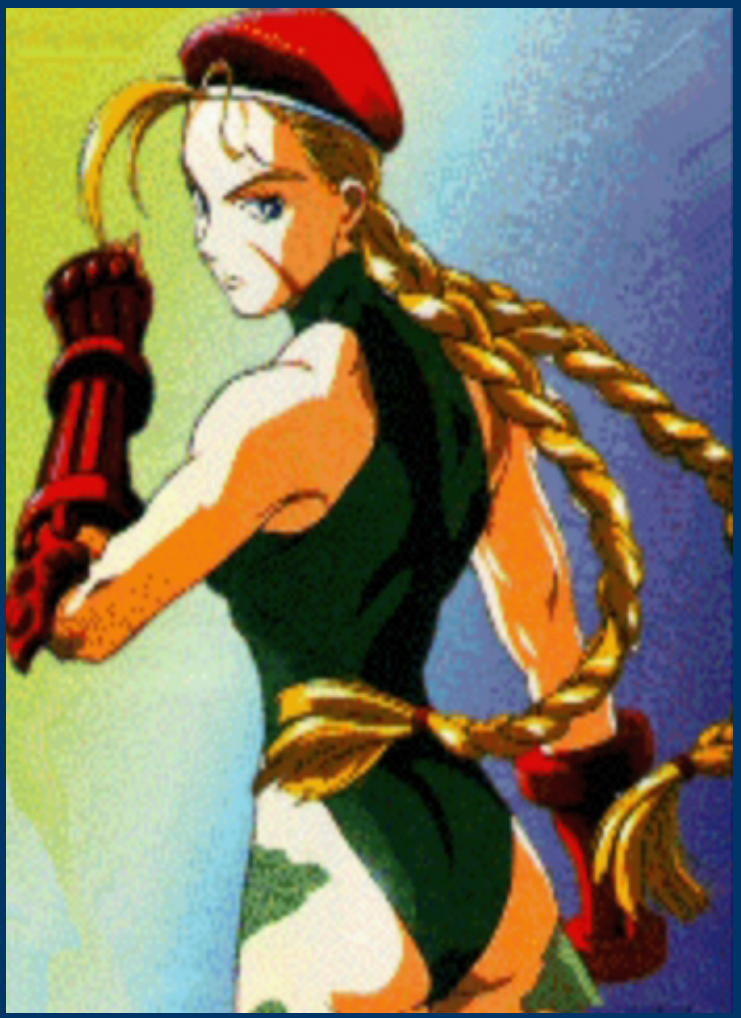
I only saw one episode of this strip. I don't know if Eagle finished during this story, or if it finished after it finished, but I will never know.

Bobby played Ryu and he had to go around the world defeating champions. The episode I read saw him face the American Guile, but got into trouble with a sonic boom.

Over the years, I really enjoyed reading the Computer Warrior strip. I have since lost my Eagle comics, but would really like to read the strips again, not only Computer Warrior, but Charlie's War and The Thirteenth Floor (I missed what happened to Max the computer after one of his victims reported to the police that he saw DI Ingram – Max's arch enemy trapped on the Thirteenth Floor).

If anyone can send me scans on CD-ROM, I would be grateful. Or, if anyone can help plug the gaps or make corrections to this information, they can write to me at:

patrick@wych01.freeserve.co.uk



LOOKING <<< BACK

Written by Patrick Furlong

Reflections on 19 years as a CPC user– Patrick Furlong

Although the Amstrad CPC464 came out in the spring of 1984, I didn't get my one until Christmas of that year. My father brought it for me from Currys in Gosport.

The first ever game I played on it was Harrier Attack and I had a lot on fun on that game as well as Oh Mummy and the two Roland games which came with the 12 pack which came with the machine.

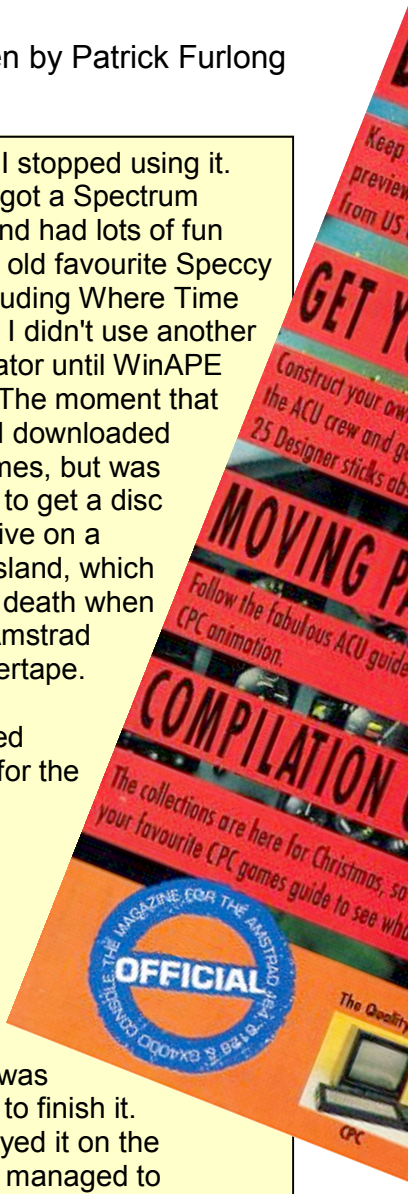
It was a while until I got hold of further games for my computer. They included a pirate version of Caves of Doom and proper copies of Professional Snooker Simulator and some game whose title I cannot remember which never loaded so it had to go back.

Over the next 6 years I built up my collection of tape games which included stuff like the Star Wars trilogy, the Dizzy games, Dr Dooms Revenge (which was one of the most temperamental games in term of loading).

I started collecting Amstrad Computer User and Amstrad Action from 1989 and really enjoyed the covertape games. It took me years to be able to play the Addams Family demo, but my favourite demos were Lemmings and Total Recall. Although I never got a copy of the full game of Total Recall, I still played the demo. The same applied to the Lemmings demo, although I got the game on disc when I got my CPC6128 in 1994.
In 1996, I got my first PC and promptly sold my CPC collection to help raise some cash to finance stuff for the PC, but I kept my magazines and I still have some of them to this day.

The moment I got access to the Internet at college,
I downloaded CPCEMU and a collection of games, but it was so fast, so I stopped using it. Later on, I got a Spectrum emulator and had lots of fun playing my old favourite Speccy games including Where Time Stood Still, I didn't use another CPC emulator until WinAPE came out. The moment that came out, I downloaded tons of games, but was never able to get a disc image of Five on a Treasure Island, which I played to death when I had the Amstrad Action covertape.

When I used emulators for the first time.
I played Subsunk, which my family and I had played for over 19 years and was never able to finish it. When I played it on the emulator, I managed to complete the whole game along with a few other games.



I have many Archive CD-ROMs in my collection and most of them have a collection of my favourite CPC games, which to tell the readers the truth, I don't play much nowadays. I mainly use my Gameboy Colour, Psion Series 5 (which I am typing this article on now) and Nintendo Gamecube for most of my games, but I do still play text adventures from companies like Level 9 and Magnetic Scrolls.

In the 12 years I owned a CPC, I have owned or still own many good machines. In no particular order, they are: Sinclair Spectrum, Commodore 64, Atari ST, Commodore Amiga, Nintendo NES, Nintendo Gameboy, SNES, Nintendo 64, Atari XE, Sharp MX400, Sega Megadrive, Sega Master System, Sega Gamegear and Sony Playstation.

For years, the Amstrad CPC was my favourite machine and ever since I got onto the internet, I have been involved in many Internet discussions, sharing my knowledge in the CPC range to the world and many fans.

I have a five year old daughter and she plays the occasional game of Super Mario Sunshine on the Nintendo Gamecube, but she loves it when she watches me play Dizzy games, particularly Treasure Island Dizzy - when she sees that on the screen, she just sits still and watches. When she decides she wants to play Treasure Island Dizzy, I'll stick it on for her. Codemasters would be onto a winner if they made Dizzy games for the modern game consoles but would be committing career suicide if they only made games for the PC and X-Box (boo! hiss!).

In conclusion, the Amstrad CPC introduced me to the world of computers well and the CPC's I owned served me well to the best of it's abilities. My young cousin (he's 10) takes the 3D graphics on the Playstation and X-Box for granted and he still can't believe that when I was his age, 3D games were state of the art for the time and I enjoyed playing 2D games a lot - sometimes I enjoy playing these games more than the games I play on the Nintendo Gamecube today, including The Two Towers and The Legend of Zelda (which I think is the best game ever).

Technical Contents

106 Knowledge Base Series

The Knowledge Base series covers a number of topics from turning on and using a CPC to more advanced stuff such as transferring CDT files to tape. There's also information on modified CPCs and a little more besides.

110 Z-80 Programming Series

The Z80 Programming Series is a number of articles written by Xavier Glattard in many of the past issues of CPC Oxygen. It covers Z80 programming for all Z80 based computers such as the CPC, Spectrum etc.

144 The Zilog Z80

John Kavanagh casts an eye back to a time when men were men and when processor clock speeds could be counted on one hand (in megaheartz!), so the opening few lines says. Read about the Zilog family of processors and learn your eZ80s from your Z80s, from your Z-80000!

124 Copychr$ Series

Here you can read every Copychr$ article ever published on the CPC Oxygen site. Written by Frederik Leighton over four entertaining and easy to follow parts covering the Copychr$ programs.

Parallel Connections

Written by John Kavanagh

WARNING:

CPC Oxygen and it writers takes no responsibility for any damage to hardware or person/s caused from following this project. We also take no responsibility for any errors and omissions.

After a long vacation from techie articles, John Kavanagh is back and this time he has overdid it by getting two computers to interface just like the animals on the Discovery channel! Somebody call the police!

Over the years emulators have become quite popular for retro computer users as it is often easier to click a mouse a few times then to dust the cobwebs from your old classic computer, searching for cables etc. Then there are the thousands of disc images on the net which are a lot easier to access then searching through a load of 3 inch discs.

However, using emulators can be soulless at the best of times, a bit like the PC itself. Also, it often feels weird using protext on a CPC emulator when you got Office XP a click away, totally pointless in ways. Then the games themselves, playing a CPC classic on a 19 inch monitor with a PC joystick does it no justice at all. For true classic gaming experience you'll need to play them on a CPC. But how do you get them there when your original discs are long lost or worse, corrupted, plus the fact that a 3 inch disk don't fit in a 3.5 inch drive (well it does, but it don't work and probably would ruin your drive) Well this is where this article comes in, we'll show you how to transfer .dsk images from the PC to the CPC using a parallel interface and a few others things along with that. The article is divided into three parts which are as follows:

Part 1: Making the Parallel Cable
Part 2: Transferring simple ASCII files
Part 3: Transferring .dsk images

Don't worry, making the cable is quite simple and cheap enough, although you do need to be a little handy with a soldering iron (or in my case a soldering gun!).

PART 1: Making the Parallel Cable

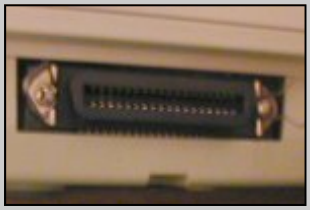
The design of the cable comes from the documentation of CPCEMU, a wonderful DOS based emulator by Marco Vieth. The programs type-ins from part 2 of this article are also from the CPCEMU documentation.

There are three types of parallel interfaces depending on where and when you got your CPC. Centronics is the most common connector within mainland Europe, while a simple edge connector was used in Britain and Ireland. The CPC Plus used a 25 pin D connector similar to that of a PC. For this article we're be making a cable for the CPC Plus, however there not much different with the other connectors, so you should not run into any major problems.

The box out 'CPC Interfaces' shows you how to identify your computer's interface:

CPC Interfaces

EDGE CONNECTOR, WAS GOING TO USE A U2 JOKE HERE BUT COULDN'T THINK OF ANY!



CENTRONICS, COULD NOT GET A PHOTO OF A BACK OF A CPC BUT THIS PHOTO TAKEN FROM A STAR LC-100 PRINTER SHOULD DO.

STANDARD PARALLEL CONECTOR AS USED IN THE PC.

Now to the CPC end, kinda the same as the PC end except now you got to use the colour of wire you applied to pin 2 to pin 13 on the CPC end and so on. Look at the drawing above for the rest of the pins, or if you prefer a table, you can find that on another part of this

page. If your using edge connectors or centronics, find the pin numbers and do exactly the same as above. Usually pin numbers are written very small next to the pins.

Once both ends are done, check the connections. Now apply your screw-lock covers if you wish, but remember they can be a nightmare to remove once they are on.

Part 2: Transferring ASCII files

Now that you got your cable made, its time to test it by transferring some files, but first you need to have the correct software both on your PC and CPC. Luckily everything you need is supplied with CPCEMU. Download it directly from the CPC Oxygen website at www.cpcoxygen.net

Catch 22!

The software for the PC is located in the "Utility" directory of the CPCEMU emulator, problem is, so is the CPC software! So to get the software on your CPC you going to need to use the parallel cable but to use the parallel cable you'll need to have the software on a 3 inch disc in your cpc, hence... Catch 22!

The solution (apart from attaching a 3.5 inch drive to your CPC) is to type in a smaller program on your CPC and work from that to transfer the main program.

The small program you need to type into your computer is named **CPCREC.BAS** and the main CPC program that is going to be transferred by cable is CPCPARA.BAS. Type the program into your CPC exactly as showing on the next page, only pressing RETURN before each line number. Save it as CPCREC.BAS. Reboot the computer and run the program, you should see the message "CPCREC v1.0 -

Shopping List

CPC using a Centronics interface

1 x 25 pin male D Connector
1x Screw Lock D type covers
1 x Cable with at least 7 wires
1 x Centronics Connector

CPC using Edge Connectors

1 x 25 pin male D Connector
1 x Screw Lock D type covers
1 x Cable with at least 7 wires
1 x 34 way Edge Connector

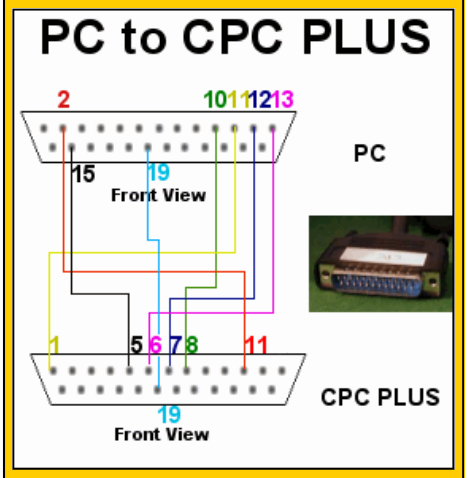
CPC Plus

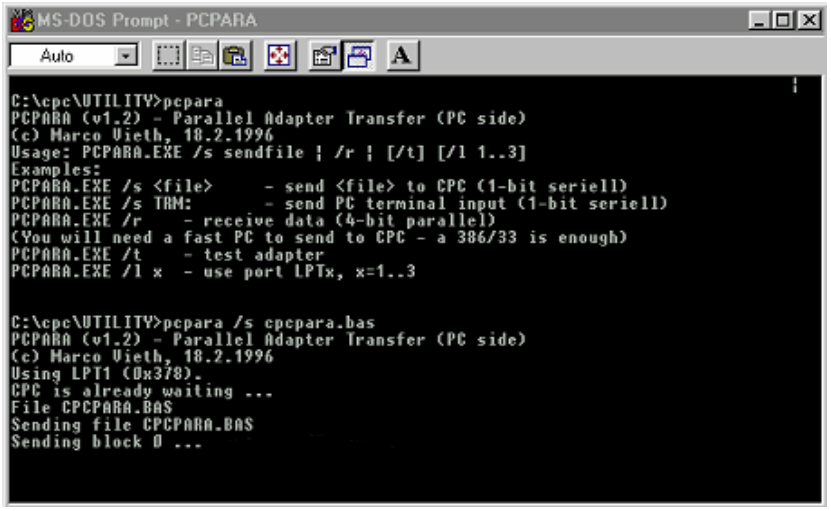
2 x 25 pin male D Connectors
2 x Screw Lock D type covers
1 x Cable with at least 7 wires

PC to CPC Connections

The following is the pin layout from the PC to the CPC.

Pin 02 to Pin 11
Pin 10 to Pin 08
Pin 11 to Pin 01
Pin 12 to Pin 07
Pin 13 to Pin 06
Pin 15 to Pin 05
Pin 19 to Pin 19





AN EXAMPLE OF WHAT TO EXPECT DURING THE TRANSFER.
THE PROCESS, WORKS BEST IN DOS.

Make sure there's enough room on your disc to receive the program, 10K is more then enough.

If you are in Windows on the PC, reboot to DOS as Windows can cause problems at the best of times. Change to the directory where CPCEMU is and then into the "Utility" directory, type the following and press return:

PCPARA /S CPCPARA.BAS

The process seems surprisingly speedy and should only take a few seconds as the block numbers rise. If you experiencing errors, check the connection and make sure you're not in Windows.

If the transfer is successful, you should have a program called CPCPARA.BAS in your CPC's disc, you can check this by typing |dir. Congratulations, you just transferred your file to the CPC!

The above when typed into a PC will send the file, filename.bas to the CPC if CPCPARA is running on it and is set to receive data. Well to send files from the CPC to the PC, just swap the S for a R and leave out the filename, so it would read:

PCPARA /R

Make sure CPCPARA on the CPC is set to send the file, then select either Fast or Slow, type the filename and then after a few moments and it'll be transferred to the current directory of the PC.

Finding Turbo Transfer

Turbo Transfer is programmed by Tino Longueira. More information on the great piece of programming can be found at ***http://skyscraper.fortunecity.com/rsi/53/***

CPCREC.BAS

```
100 REM CPCREC.BAS - program to receive a file from the PC
110 REM Marco Vieth, 26.7.1994
120 REM
130 CLEAR:DEFSTR a:DEFINT b-z
140 ladr=&A000:IF PEEK(ladr)=&FE THEN 180
150 PRINT"Please wait...":MEMORY ladr-1
160 sum=0:FOR i=&A000 TO &A0A6:READ t$:
POKE i,VAL("&"+t$):sum=UNT(sum+PEEK(i))
170 NEXT:READ t$:IF VAL("&"+t$)<>sum THEN
PRINT"Checksum ERROR !":STOP
180 CLOSEIN:CLOSEOUT
190 OUT &EF00,&FF:'inactive
200 MODE 2
210 PRINT"CPCREC v1.0 - program to receive files"
220 PRINT"from a PC using the parallel interface"
230 PRINT
240 a=SPACE$(255)
250 CALL ladr,@a:IF LEN(a)=0 THEN 250
260 IF a="TRM:" THEN PRINT"Terminal not supported.":STOP
270 PRINT"Receiving file ";a
280 OPENOUT "!"+a
290 CALL ladr,@a:IF LEN(a)>0 THEN PRINT#9,a;:GOTO 290
300 CLOSEOUT:PRINT"Ok.":PRINT:GOTO 210
310 DATA FE,01,C0,DD,6E,00,DD,66,01,E5,CD,7E,A0,E1,D0,36
320 DATA 00,C9,C5,D5,E5,11,20,4E,06,F5,ED,78,E6,40,6F,3E
330 DATA BF,06,EF,ED,79,06,F5,ED,78,4F,E6,40,AD,20,06,1B
340 DATA 7A,B3,20,F1,37,F5,3E,FF,06,EF,ED,79,F1,38,3A,1E
350 DATA 00,16,08,F3,21,10,27,06,F5,79,E6,40,4F,ED,78,E6
360 DATA 40,A9,20,08,2B,7C,B5,20,F4,37,18,1D,3E,DF,06,EF
370 DATA ED,79,ED,79,06,F5,ED,78,FB,4F,17,17,CB,1B,06,EF
380 DATA 3E,FF,ED,79,15,20,CC,A7,7B,E1,D1,C1,FB,C9,CD,12
390 DATA A0,38,23,77,B7,37,28,1E,47,23,7E,23,66,6F,04,18
400 DATA 12,11,0A,00,CD,12,A0,30,08,1B,7A,B3,20,F6,37,18
410 DATA 05,77,23,10,EC,A7,C9
420 DATA 4C94
430 END
```

THE ACTUAL FILE TRANSFER PROCESS

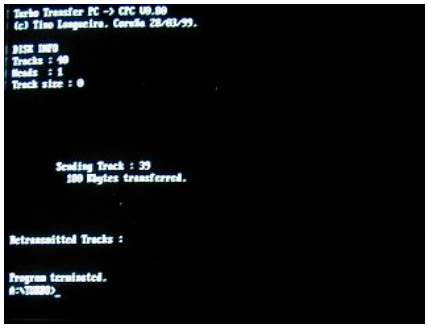
THE FUN STUFF!

It's all very good sending files back and forth between two very different computers, but what about all those thousands of .DSK images full of games and other great software. Well, they can also be transferred to the CPC in an understandable format leading the way for access of every CPC game you ever wanted (almost).

Ok, for those who don't know, a .DSK image is a file containing an image of all the files on a CPC disc, for example - a game such as Elite, word processor program or whatever. It the standard format for CPC emulators. Since most of today's downloads are in .DSK format it can be very desirable to be able to transfer those .DSK files back on to 3 inch discs for use on the Amstrad.

GETTING DOWN TO IT!

To start, download Turbo Transfer V0.80 from the bottom of the page. Don't download version V0.92 yet as it don't have the CPC software in a format you can currently use i.e. it's in .DSK format and since you don't have 'Turbo Transfer' on the CPC yet, it won't know what to do with it.
Once downloaded, unzip it the the



TURBO TRANSFER SCREEN ON THE PC

Utility directory within CPCEMU so you don't have to worry about correct paths when entering commands. While you at it, drop a few .DSK images in there as well, you'll be using them soon.
Switch off both computers, I'll assume you have your CPC on almost all of

TURBO TRANSFER IN FULL SWING ON THE CPC

the time :-) and connect the cable you made in issue 7. Reboot the PC into DOS (if you using XP I'll have a solution for next issue) and enter that Utility directory within the CPCEMU directory. Now, switch on the CPC and load CPCPARA.BAS (hope you still have it) and press R to receive.

On the PC, keeping in the same directory type:

PCPARA /S Turbotr.bas

and when that's done, type:

PCPARA /S Turbotr.bin

Now that Turbo Transfer is on both your PC and CPC we can have some fun (ok, I know I promised that earlier), reboot your CPC and load Turbo Transfer by typing, wait for it!

Run"Turbotr.bas"

WARNING: *Take out your 3 inch DISC and inset it with a blank one as all information will be lost. Even if the .DSK image is only a small file, all the data on disc will be erased, forever! You have been warned.*

It loads and asks you to press any key, but don't yet. First you got to send the .DSK image from the PC first. You can do this by typing:

Turbotr filename.dsk

Since I was transferring the game Elite, the filename was elitee.dsk.

Now press any key on the CPC within a split seconds the .DSK image begins to transfer to the black disc in your CPC (it is blank, right!?).

Both computers display information while the process is happening, it should only take a about a minute. Once it done, reboot your CPC, and load the disc as normal.
Congratulations you've just transferred your first .DSK file to your CPC.

UPGRADING TO .92

It's recommended that you upgrade to the latest version of Turbo Transfer which is at time of writing is v0.92. The CPC files are in a .DSK image so that why you had to use version 0.80 first.

When you have the latest version downloaded, extract TURBOTR.DSK to the utility directory of CPCEMU and transfer it with (remember to have Turbotr running on the CPC also):

Turbotr turbotr.dsk

Make sure you're using the usual version 0.80 on the PC as it not good to mix versions. Once transferred, extract the latest PC version of Turbotr.exe to the utility directory thus erasing version 0.80 and having the superior version 0.92 on both the CPC and PC. That's it, check future issues of CPCO for possible follow-up articles. Have fun!

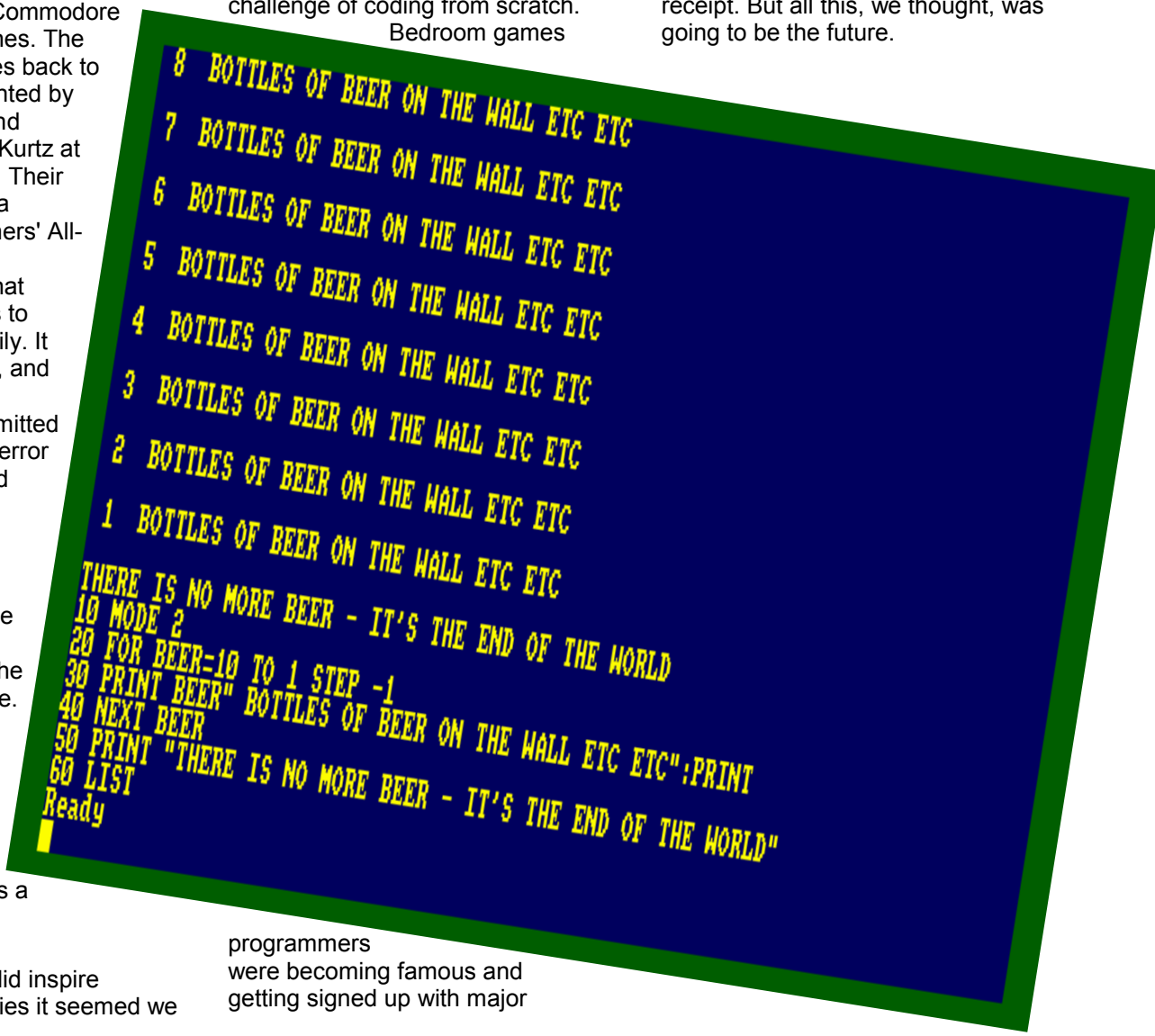
ELITE - TRANSFERRED FROM .DSK IMAGE ON THE PC TO 3 INCH DISC SUCCESSFULLY EXCEPT FOR THE FACT THAT THE ELITE DSK IMAGE DON'T WORK CORRECTLY ON THE AMSTRAD OR ANY OF THE EMULATORS I TRIED.

B is for Basic

Written by Sean McManus

This article appeared in Wacci as part of its 'A-Z' of the Amstrad, and introduces Amstrad Basic.

By the time the Amstrad came along, Basic was the default programming language on most home computers including the competing Acorn, Commodore and Sinclair machines. The language itself dates back to 1964 and was invented by John G. Kemeny and Professor Thomas Kurtz at Dartmouth College. Their goal was to create a language (a Beginners' All-Purpose Symbolic Instruction Code) that would enable users to write programs easily. It should be forgiving, and allow for obvious information to be omitted and provide useful error messages. It should shield users from needing to know anything about the hardware, and it should be extensible so more advanced users could make the most of the machine. Compared to machine code (which locks up the machine if you get it wrong, without any error messages), Basic is a doddle.



And the language did inspire people. In the eighties it seemed we would conquer the world by learning to program our home computers. A public domain scene was built around enthusiasts writing programs to meet their needs, either because there wasn't anything specific enough on the market or because they relished the challenge of coding from scratch. Bedroom games programmers were becoming famous and getting signed up with major software houses, and Basic was the first step to stardom. I knew someone in a small business who taught himself to program his Spectrum so that he could run his stock management on it. Data was loaded from tape and the print-outs had all the archival qualities of a till receipt. But all this, we thought, was going to be the future.

Programming was well-supported by magazines, with Amstrad Action carrying listings and coding tips until its closure. In the early days, Amstrad Computer User published listings that ran not only over pages and pages of an issue, but also serialised programs between issues. It seems incredible now that people typed these in, but many of them found that playing with these programs gave them the first insight into how they might write their own. Nor were these programs sub-standard. Some of the authors had clearly been inspired by Amsoft's games to create something better and cheaper for computer users, and given the standard on the market at the time, it was relatively easy to impress.

Over time, published listings got shorter and ACU started a regular 10 liners section. The brief was to create something impressive, useful or interesting in ten lines or fewer. The quality varied a lot and the end result was usually of more interest to other programmers than end users. While people could type in the multipage games listings in the early days and enjoy playing them, the 10 line games were too simple to have lasting appeal. To appreciate any beauty in them, you had to like looking at how they were written. Some useful utilities appeared on these listings pages, but there were many trivial demonstrations that catered much more for the author's urge to show off than for any unsatisfied programming need among readers.

One reason that the CPC could sustain a community of programmers for so long after the other 8 bit machines had been found out for being toys is that the version of Basic that came with the machine was so good. It was written by Locomotive Software, a team who worked for Acorn before hand. It allowed programmers to do nearly everything that the machine's own built-in programs could do, although machine code programmers would later teach the machine new tricks. The demonstration tape which ran on machines in shops, which was designed to amaze and delight its audience with various light shows and software demos, was itself largely written in Basic and some of the early games on the machine were also coded in Basic.

The few omissions in what Basic could do were mostly included in Basic 1.1, an updated language with a few new commands which featured on later computers. These new commands included a fill routine and a command to find out what text character was on-screen at a particular point. Had these appeared earlier, they might have been more useful, but most programmers were aiming for cross-compatibility between the early machines and the later ones with the programs they wrote so the new commands were largely ignored. Magazines and software houses insisted on compatibility, and even PD libraries preferred programs that worked across most machines unless there was a good reason why they shouldn't. Memory management remained clumsy, a missed opportunity to make it easier to exploit the potential of the 128K machines.

But Basic was slow. It's an interpreted language, which means that each time a command is run, the computer has to translate it into machine code before it can do anything with it. This translation meant that Basic would never be enough for actions that need speed like graphics handling, but luckily Amstrad Basic could be combined with machine code by using RSXs. These were commands preceded by a bar that could be added to Basic by running a machine code program. When the command was entered, the machine code was run.

Programming Basic enabled many Amstrad users to invent their own worlds on their home computers. It was an intellectual sport, where there was pride in pushing the machine and making it do something new. And for many people, programming the machine was more satisfying than playing the games that were available for it.

As it turned out, learning Basic wasn't the way to conquer the world. Computers became too complex for that and applications like spreadsheets and database programs became powerful enough to meet even the most esoteric requirements. The point and click interface means that most home computers don't even have a programming language installed now for people to use if they want to begin coding. Nor are computers standardised enough for Basic to enable users to leverage all the hardware's features.

But those Amstrad users who learned to program will never fear the lack of software for the machine. If ever they need it to do something new, they'll flop open the manual, play their favourite album and let the good times roll.

Sean McManus website can be found at: http://www.sean.co.uk/books/amstrad
It contains the BASIC programming tutorial "The Basic Idea" which was commercially published in 1994.

This article appeared in Wacci as part of its 'A-Z' of the Amstrad, and introduces Amstrad Basic.

Super 464

Written by John Kavanagh

On a cold winter's evening while browsing the net as people do I came across a web page which in an instant left me totally gob smacked, what looked like a greatly modified 464.

I promptly emailed a certain Ian Dowse asking for permission to include part of his project for inclusion within CPCO which to my surprise was equally prompt in replying. I knew this was the answer to the first main feature for CPCO and what a great one it is.

RESET SWITCH - USED FOR RESETING THE COMPUTER'S NI-CD BATTERY BACKED-UP RAM.

Floppy Heaven

Looking at the photos the first noticeable modification is the inbuilt 3.5 inch floppy drive along with a two button switch. At first sight this seems impossible as it's buried deep in where the CPC electronics are, but all becomes clear when it was explained that in later versions of the CPC 464 the circuit board only covered roughly half the length of the computer. Of course any floppy drive is completely useless without a disk controller.

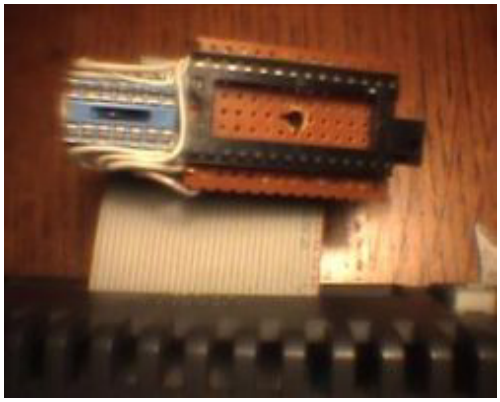
FRONT OF CPC SHOWING INSTALLED DISK DRIVE ALONG WITH A DIGITAL SIDE SWITCH.

A uA765 controller was used for this along with a CPC 6128 style edge connector at the rear for connecting an external drive. That is where the two button digital switch comes in, pressing 1 to access the internal drive and pressing 2 for accessing any 3, 3.5 or 5.25 inch drive you may have. Of course this can still be achieved the old fashioned way with the SIDE command. Basically there's enough kit there to suit most CPC users, and of course with a proper 3.5 inch floppy drive, access should be quite fast as well.

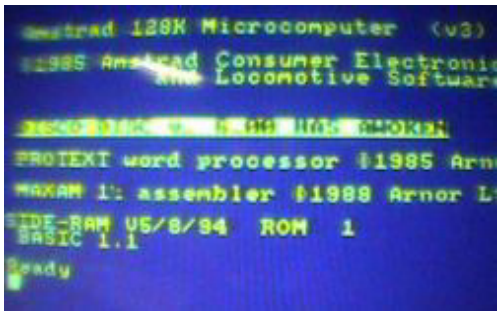
Ram and more Ram

Perhaps more interesting is the on-board 512K of RAM, based on two 256K SIMM modules which means that there is about 8 times more physical memory then the original 464 and 4 times more memory then the CPC6128. As with all Z80 processor based computers only 64k is available at any one time, the rest of the memory is accessible through bank switching.

Amazingly there is ever more RAM in the form of 128K of internal battery packed static memory. This is used to emulate expansion ROMs such as Maxam and Protext and the upgraded version of Amstrad Basic 1.1. Being not a real ROM, memory is lost during switch off which means there's the painful task of loading the disk Rom image from tape followed by loading other ROMS as needed from disk into 128 RAM. To help fix this problem there is an build-in Ni Cd battery to keep memory alive in RAM even when the CPC is switched off.



EPROM PROGRAMMER SOCKET.



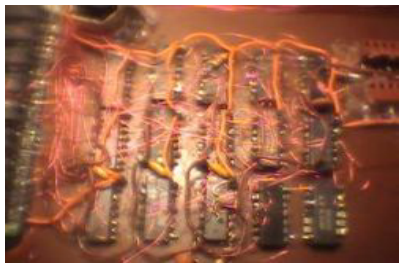
START UP SCREEN SHOWING BASIC 1.1 AND A FEW ROMS LIKE PROTEXT AND MAXAM ASSEMBLER.

Even More Goodies

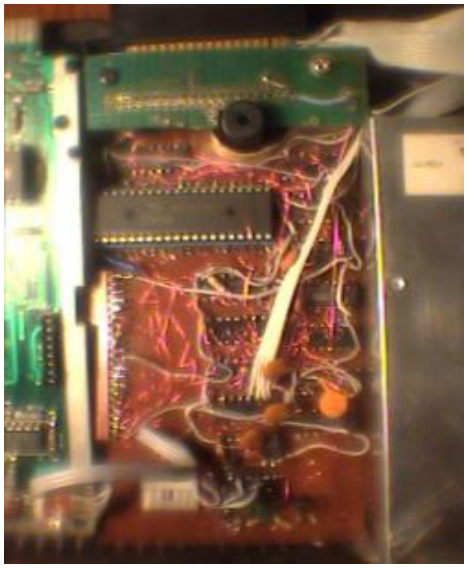
You want more? OK, how about an 8bit printer port instead of the 7 bit standard. Almost forgot to mention that there's an internal EPROM programmer. I used the word internal very loosely since the actual EPROM socket where the EPROM is inserted is actually hanging from the back of the computer. It also requires an external power supply.

Verdict

This is truly one of the most impressive upgrades I ever seen on a CPC. There really is enough power to get tasks completed quickly, for example can you name another modern personal computer that can have a word processor loaded in 5 seconds flat from power on ? The memory, all 640K of it is more then enough for any CPC user, but then again that what Bill Gates said about the PC! The only two things I can knock about the upgrade is the way a lot of the components may look a little messy when soldiered onto the motherboard. It's only a small niggle. More seriously is the noticeable lack of cooling, surely with so much more electronics pack into the CPC case can only cause more heat. As a lot of CPC owners already know (especially in hotter countries) this could cause problems down the line. Having said that, overall I think it a fine upgrade which leaves the 464 a computer that's too good to resist messing about with.



LARGE 512 KILOBYTES OF RAM WITH CONTROL LOGIC.



THE FLOPPY DISK CONTROLLER BOARD. IT SITS BETWEEN THE MAIN CPC BOARD AND THE 3.5 INCH DISK DRIVE.

Specs for 464 and modified 464

ITEM	CPC 464	Modified 464	
CPU	Z80 4MHz	Z80 4MHz	
RAM	64 KB	512 KB PLUS 128 KB FOR ROMS	
DRIVES	TAPE	TAPE, 3.5 DRIVE	
OS	BASIC 1.0	BASIC 1.1	
PRINT PT	7 BIT	8 BIT	
EPROM	NONE	YES	
EXTRA	NONE	DISK CONTROLLER, 6128 STYLE EDGE	CONNECTOR, DIGITAL SIDE SELECT SWITCH, NI-CD BATTERY FOR 128KB RAM. MAXIM AND PROTEXT ON ROM.

THE FUTURE – BOX OUT

If reading this article has wet your appetite for modifying your CPC, then I got great news for you. It is planned to over the coming months to include a project for upgrading your CPC. Due to a lot of our readers being worried about opening up there loved and cherished CPC's and soldiering bits onto it you be glad to know that in the upcoming project there will be no opening up of your CPC at all!

It is planned to produce a system case that sits pretty under your monitor and will be connected to the CPC through the expansion edge connector. It going to have build in disk drives, loads of memory, ROMS, EPROM programmers, the lot basically. All of this which smoothly lead us on to a little problem.... I can't built it! You will! Well at least some technical readers out there who have a deep love for CPC. CPC Oxygen will just be the HQ where all the ideas for the project are going to get put together. All technical information for the project will be included in each issue as it builds up, until there's enough information to actually built the project.

I look forward to this project and hopes it be a success, it could just be the project to push the CPC into the 21 Century.

KNOWLEDGE—BASE 1
Starting a CPC

Written by John Kavanagh

Right from the simplistic beginning to the complex end, Knowledge base is the key to your answers.

On first looks of this article, many CPC users will believe that this article is too simple to be of any use... and they may be right. But still, there are alot of people who have never seen a CPC before and this is what part 1 of knowledge base is about. From there we will be moving on and on, getting slightly more complex each time, building up a databank of information. Now that I explained myself, lets start:

After reading on how to switch a CPC on, you be presented with a screen with a few lines of text. What no operating system? Well emm that is the operating system. The CPC was made in 1984, so there's no graphical user interface, mouse controls as standard etc.

Not surprisingly you may want to make the computer "do something" right now. Ok, type the following:

```
10 out &bc00,1
20 for x=1 to 40
30 for t=1 to 50
40 next t
50 out &bd00,x
60 next x
```

Remember to press ENTER after every line. Now type: RUN and press ENTER.

Pretty, ya? well you just wrote your first BASIC program, BASIC (Beginners All-purpose Symbolic Instruction Code) is a simple programming language which is flexible enough (if a bit slow) for most projects. Over the coming months, we will show you how to program properly and actually understand what you're typing. We may also start a few simple programming projects along the way.

Now you may actually want to load one of those funny looking cassette games you may have lying around.

Firstly, to reset your computer, hold down CTRL + SHIFT + ESC. It a good idea to do this when loading stuff so that's there's no conflict between the program on tape and what's in memory. If the usual ctrl shift and esc don't work, just flick the on switch off and then back on again.

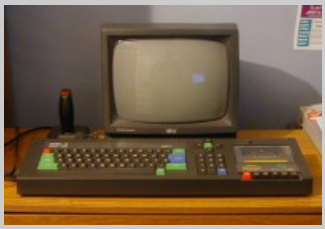
Insert your cassette, I refuse to explain how to put a cassette into a datacorder, so you got to work that bit out yourself. Make sure you rewind the tape to the beginning of side A. Now do the following:

Hold down, CTRL and while holding that down, press the small ENTER key. A message should pop up saying.

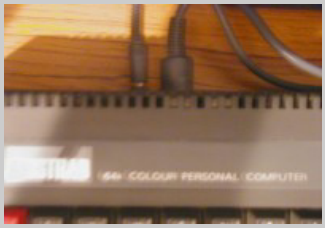
Press PLAY and then any key.

Press PLAY on your datacorder and then press any key, well any key except ESC and SHIFT. Wait a few minutes and then you're game will be loaded. Have fun.... Until next time.

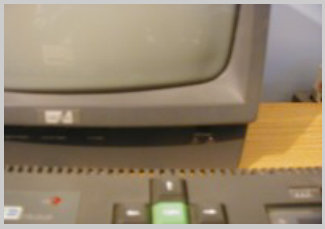
Starting a CPC



This example explains how to switch on a CPC 464, but its similar in all CPC models.



When you have the monitor and computer positioned, plus both the monitor cable and power cable from the monitor to the CPC. Remember, if it won't fit, it's the wrong socket. Idiot prove really.



Ok, now the fun part - switching it on. Just press the ON button under the monitor.

Now press the ON switch on the side of the keyboard. Note the volume control, it may come in handy if you not hearing any sound from your CPC. That's it! the operating systems pops up instantly, no waiting around like modern PC's.

You may want to plug in a joystick/joypad in here. Most digital controllers from game consoles like the Sega Master System should work. Failing that, you should be able to get an compatible controller from a small ads paper.

KNOWLEDGE—BASE 2

Written by John Kavanagh

This month we get seriously involved in BASIC programming and we'll also talk a little about PLAYTZX, a utility to get PC CDT files onto a real CPC.

BASIC stands for Beginners' All-purpose Symbolic Instruction Code. its basically (no pun intended) a interpretive programming language which is easy to learn as it's code is almost like human instructions. To see what we mean, have a look at the following piece of code:

```
10 Print "Hi There!";
20 goto 10
RUN
```

Type this program into your CPC, not forgetting to Press ENTER after each line. If type correctly you will see a series of words words run across the screen and scroll down the screen after each line. Lets see how this program works:

```
10 Print "Hi There!";
```

The first part "10" tell the computer to store the following commands within memory, otherwise the command would have started before you could type the rest of it. It could be any number between 1 and 65535 but 10 is a logical start.

"Print" means there going to be an output of some sort, in this case the text in the brackets " " will be displayed on screen. So for example you could have had "Guten Tag :-)" in the brackets instead of "Hi There!", in fact oyu could have any text you want. The ";" bit tell the computer to display the next next character immediatly following the previous one. A "," character at the end would jump 13 characters before displaying the next print command. If there's nothing at the end, then then computer just jumps to the next line.

```
20 goto 10
```

Again, the "20" at the beginning means that the command will be stored in memory. Goto 10 means excatly that, to go back to line 10, which was the print command.

```
Run
```

This just Runs the program stored in memory. Simple? Oh ya, almost forgot, to end the program press the ESC key twice. To delete a line, for example line 20, use the following:

```
Delete 20
```

Notice how there's no numbers before the command, that because you want the program to run the moment you press ENTER instead of just having it stored in memory.

If you had a large program, be it a database or a space invaders clone and you need to delete alot of lines fast, for example all the code between line 50 and 700 then use the Delete command like this:

```
Delete 50-700
```

Also, it is possible to type the line number and press enter to delete that line. For example, type:

```
10
```

This will delete line 10. This is excatly not by the book but it's nice to know. I'll leave you all with a little program to try out:

```
10 Mode 2
20 Input "What is your name";a$
20 Input"What is your age";age
30 Print "tell me, "a$" how does it feel being "age". Oh ok, never mind"
```

I'll explain how this works in Issue 4, we may also get involved in some graphics programming.

PLAYTZX

Playtzx is a utility to output the now popular .cdt files to a cassette recorder so you can play all those old classic computer games on your CPC just like the way life meant it to be!

Playtzx (programmed by Tomaz Kac) was originally designed to play the Spectrums .tzx format but does a great job with the CPC as well as many other formats including the SAM Coupe and Jupiter ACE.

Bascially it works by playing, in the CPC case, cdt files as heard by the CPC's datacorder. The sound is then outputted with a simple self made cable to an ordinary hi-fi system or a cassete recorder that has a input connector. Even a simple cassette recorder with a microphone input will do. To use PlayTZX you'll need to open a MS-DOS screen if using Windows. (we'll talk about other formats in next month's issue). It best to have your cdt files and PlayTZX program all in the same directory. To transfer the cdt file to tape do the following:

Plug a cable from your PC's speakers output to your Hi-Fi sound input. Bootup the computer (may take a while since it's a PC)

It best to test the input levels by playing some music/sound on your PC, perhaps a wav file, and record that from your Hi-Fi. Play back the cassette and listen to the sound quality, too low and you need to turn up the volume control within windows, if distorted then you'll need to lower the input volume.

When you got that right, it time for the real thing. Make sure you got a blank cassette in your Hi-Fi, slightly forward it a bit to make sure to started recording at the right place (a quick second on the forward button should do).

In the MS-DOS window, goto the directory that holds your PlayTZX program and cdt file. To start the transfer type the following:

```
PLAYTZX zzzzzzzz.cdt
```

Zzzzzzzzz is the name of the cdt file. Press RECORD on your Hi-fi and Return (Enter) on your PC and soon you'll hear the glorious sound that's hauntingly familiar playing into your Hi-fi.

Now you wait, wait and wait some more as you start to remember how annoying it was to load programs by cassette. Once finished you can stop the cassette and close thw MS-DOS window. Rewind too ,after all you don't want to wear down your datacorder in your CPC, now do you :-)

Loading the tape in your CPC be an experience. I actually haven't done it yet due to not having a sound input in my hi-fi, not the one next ot my CPC anyways. But that all be sorted by next issue. Next month we'll show you how to make the cable and to find other ways of getting your cdt files onto tape. We'll also show you how to avoid problems. Until next time. bye.

KNOWLEDGE—BASE 3

Written by John Kavanagh and Julian Cassin

This month Julian Cassin talks to us about his modified 664, while John Kavanagh get tied up in cable in an effort of saving a few Euro.

Un-Jiggle my 664!
Firstly, let me introduce myself. I am Julian Cassin, an active CPCer from Down Under. After seeing the feature on the modded 464 in issue one of CPC Oxygen I thought it might be worthwhile mentioning what I had done to my 664 (hopefully this would start a trend in getting other modded CPC owners to show off theirs).

It started when my hardware expansions at the back of the 664 all daisychained started to become annoyingly unreliable causing the odd crash when bumping the CPC, or causing it not to turn on unless I jiggled them. After a little thought I figured I'd experiment to see what could fit
inside.

I had an Amstrad RS232C serial interface (which has a gender-bender for the expansion port with it). This gender-bender fitted nicely inside the strange bent expansion plug of the ROMBO ROMbox and allowed me to put the ROMBox internally into the CPC while still keeping the expansion port usable. I had to file away about 0.5mm at the top of the expansion port plastic where the ribbon cable goes into the CPC to allow a nice firm fit.

Pulling apart my Hackit revealed it was nothing more than a normal CPC ROM allocated to ROM ID0. As the ROMBO has a Socket with ID0, which isn't really useful for much else, I stuck the Hackit into that, but it needed a switch as I would permanently be booting into Hackit. So, after a bit of tracing the ROMboard tracks, I found a suitable track to cut and put a switch onto. I had to experiment also in finding a ROMboard disable switch, but in the end I found a suitable track to cut.

The Multiface also was in a funny box, and because I had the ROMBO internal (which offered now an internal expansion port), I plugged the Multiface into that. Sadly, I haven't got around to putting in a Multiface disable switch
yet (so it can still be detected by software).

The 3.5" disc drive has the exact same dimensions as the top of the inbuilt 3" drive - so it is just taped to the top with it's power extended from inside the CPC and a short ribbon running from the drive B port.

As for the other switches, ABBA, selectable Disc ROMs & OS ROMs, I had a friend install the ROM switches about 10 years ago. The ABBA switch I think was published in AA.

For the future, I have considered putting the RS232C inside also (there is just enough room) but I haven't got around to it yet. Also internal stereo amplified speakers would be cool. Possibly with an inbuilt D/A converter (digiblaster?) and of course a disable Multiface switch.

The modifications though are not as radical as the ones in the Super464 (and boy, would I love some of the mods in that) but at least I now have a very reliable expanded 664 which no longer fails to turn on or crashes when I bump it.

Check the following link for more imformation:

members.optushome.com.au/zhulien/

Audio Connections

Last issue I wrote about getting .cdt files from a PC onto a cassette for use in a real CPC. I also said that an audio cable from the PC's output socket to the mic input of your hi-fi was needed. So this month we talk about making this simple cable.

NOTICE THE LEFT SWITCHES ON THE SIDE OF THE COMPUTER? KOOL OR WHAT?

A READY MADE 3.5 mm JACK AUDIO CABLE

For those of you who prefer to purchases such cable. It is known as a 'stereo 3.5 jack to 3.5 jack audio cable'. You can also get a mono version for about the same money but the stereo version seems better value if you planning on using the cable for other stuff that require stereo. It should not set you back more then five or six Euro.

For those who still not quite sure what a '3.5 mm Jack' is, have a look at the headphones of your Walkman. Yes, that's a 3.5 mm jack. Most people have a coupe of those cheap headphones lying about, I know I have. The process is simple:

1. Get old of two pair of busted headphones or anything else that has a 3.5 mm audio jack.

2. Examine both headphones and choose the one with the longest cable (as the CPC always seems to be miles way from any hi-fi).

3. Simply cut the cable, pull the two wires apart slightly and strip a centimetre or thereabouts off each wire.

4. Do the same for the other headphone, you could leave the wire for that on too if you wish for an extra long cable, if not then remember not to cut it too short and make life difficult for you. Always leave about four or five cm.

5. When you have both ends stripped, simply join them together and twist (not too tightly) remembering that it a good idea to join the same colours together.

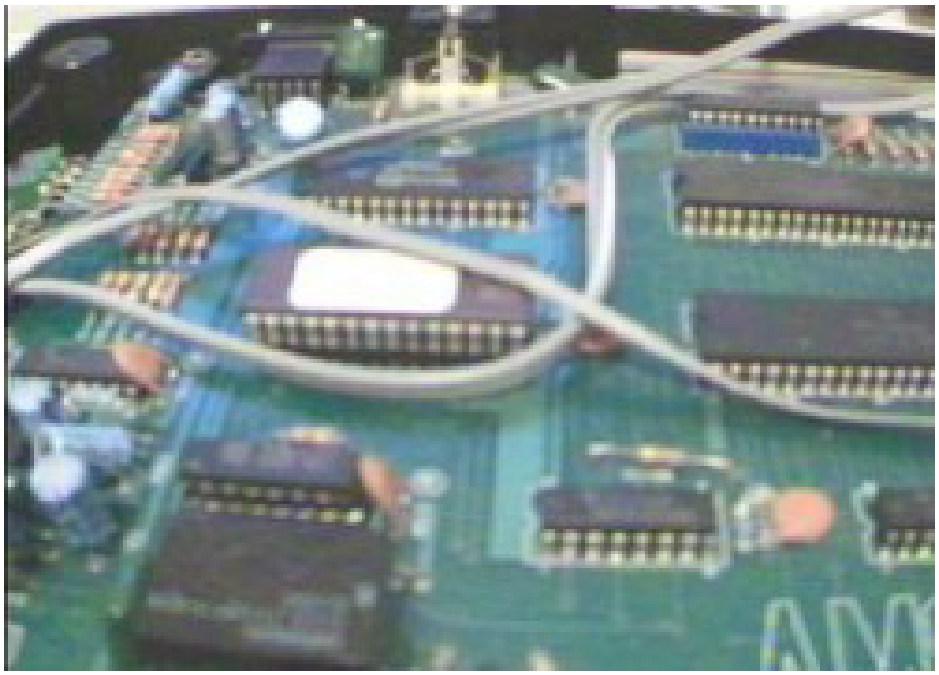
6. Put coloured insulated tape around each wire, making sure no bare wire is showing, and then around both wires making a secure connection.

Thats it, simple or what. Refer back to Issue 3 for more information on connecting a Hifi to your computer. We might do a main feature article on connecting the CPC to the world and more. Let us know what you think at the usual address.

Almost forgot, the ongoing BASIC tutorial will return next month. Until then, have fun playing with your datacorders!

664 Specifications

- Z80A CPU @ 4Mhz
- 576Kb RAM
- Inbulit 40/80 track ROMs
- Inbuilt 464/6128 ROMs
- Inbuilt Protext, Maxam, Disco6, Super2 & Utopia ROMs
- Inbuilt 40 track 3" FDD
- Inbuilt (sortof) 80/40 track 3.5" FDD
- Internal Multiface 2
- Internal 6 socket ROMboard
- Internal Hackit! cartridge
- ABBA switch
- ROMboard OFF switch
- Hackit! OFF switch
- 40/80 track switch
- 464/6128 OS switch



ONE OF THE MORE UNUSUAL MODS, DUEL 464 / 6128 ROMS. WITH THIS YOU COULD CHANGE FROM A 464 ROM TO A 6128's WITH A FLICK OF A SWITCH.

Z80 Programming

Written by Xavier Glattard

We begin in this fourth issue of CPC-Oxygen a series of papers about the Z80 programming with assembly language.

These lessons will be about low level programming of the Z80 microprocessor. The problems we will tackle will not be CPC specific. The code will run on any Z80 based system with only minor modification : Amstrad CPC, PCW, Sinclair ZX 80/81, Spectrum, etc. We assume that you know the basic of the Z80 microprocessor : we will not explain what is a register, nor what are the Z80 registers. We will not explain binary arithmetic, nor hexadecimal numbers. We also assume you have at your disposal the software tools to use the pieces of assembly code found in theses papers.

The exact content of these papers is not well defined. We will start with simple mathematical functions. At the end (if any) we write a full C++ compiler with full library (hem... within the next twenty years, ok ? ;o). The next parts will depend on your reactions. Give us your opinion, send us your question, suggest us some problems...

johnkavanagh@cpcoxygen.net

Lesson 1 - Multiplying

The Z80 is a very old processor. Compared even to the Intel 8088 (the ancestor of the Pentium), She is slow and not very powerful. Oh ! Understand me : I love this chip ! I like Her like I like to recall my first kiss : not the best one, but She changed my life.

Z80 is simple and efficient. She can add two numbers. She can substract a number to an other. But She can't multiply, nor divide. Then we have to help Her.

Here is the simplest way to multiply two numbers: ***(see code 1.1)***

It is very simple. Over simple. And very inefficient.

Try to remember the way you proceed to multiply two numbers. With pen and paper, I mean. I know, it's so far away, you were so young, computer are so useful, and so on. But try any way. I help you with an example:

```
  123
x 45
------
 615
 492-
 ------
= 5535
```

Do you remember know ? Right! You got it! First you multiply the first number (123) by 5
(the less significant figure of 45) : you get 615. Next you multiply the first number by 4 (the next, and last, figure of 45), then by 10 : you get 4920. Add it to the previous result :

615+4920 = 5535.

Now, an other example, simpler:

```
   0110
x  1101
   ---------
   0110 (a)
   0000- (b)
   0110-- (c)
   0110--- (d)
   ---------
= 1001110
```

Simple, isnt'it ?

Oops ! 0+1+1=...0 ? 0+1=...0 ? No, of course. I forgot to tell you : this operation use binary numbers :
1+1 = 10

There is something really cool with binary figures : they are very easy to multiply, and the Z80 do it really fast!

0 x 1 = 0
1 x 1 = 1
anything x 0 = 0
something x 1 = thesamething

No carry. Never.

Have a look to our example:

(a) multiply the 1st number (0110) by the first figure (1) (from right to left) of the second number (1101) : you get the 1st number itself, and keep it.

(b) multiply the 1st number by the 2nd figure (0) of the 2nd number : you get zero, and do nothing.

(c) multiply the 1st number by the 3rd figure (1) of the 2nd number : you get the 1st number itself, shift it twice to the left (multiply by two) and sum it with the previous result.

(d) multiply the 1st number by the 4th figure (1) of the 2nd number : you get the 1st number itself, shift it three times to the left and sum it with the previous result.

Code 1.1:

```
; The 1st number is in a
; The 2nd number is in l
; The result will be in a
; Both numbers have to be between 0 and 15 (4 bits long)

ld b,l ; init loop index
ld h,a ; store the 2nd operand in h
xor a ; do a = 0 ; same as ld a,0 but shorter
Begin:
add a,h ; a = a + h
djnz Begin ; b = b - 1
; if b {CARSPECIAUX 185 \f "Symbol"} 0 then goto Begin
End:
; The result is in a, between 0 and 255 (8 bits long)
```

Code 1.2:

```
; The 1st number is in a
; The 2nd number is in l
; The result will be in a
; Both numbers have to be between 0 and 15 (4 bits long)
ld h,a ; h will use for intermediate result
ld b,4 ; init loop index : 4 bits
xor a ; do a = 0 ; same as ld a,0 but shorter
Begin:
sra l ; shift l to the right and receive the falling bit in the carry
jr nc,NextBit ; if the falling bit is 0, do nothing
add a,h ; else a = a + h
NextBit:
sla h ; shift h to the left : do h = h * 2
djnz Begin ; b = b - 1
End:
```

There is a slightly different way to do this operation:

(a) multiply the 1st number (0110) by the first figure (1) (from right to left) of the second number (1101): you get the 1st number itself, and keep it. Multiply the 1st number by 2 and keep it.

(b) multiply the 1st number (01100) by the 2nd figure (0) of the 2nd number : you get zero, and do nothing. Multiply the 1st number by 2 and keep it.

(c) multiply the 1st number (011000) by the 3rd figure (1) of the 2nd number : you get the 1st number itself and sum it with the previous result. Multiply the 1st number by 2 and keep it.

(d) multiply the 1st number (0110000) by the 4th figure (1) of the 2nd number : you get the 1st number itself and sum it with the previous result.

Not very clear ? Well. The assembly code may help you. ***(see Code 1.2)***

There are many ways to do this. This one use four registers (a, b, h and l) and is 13 bytes long.

Hey ! The first piece of code is better ! It use only three registers (a, b and l) and is 4 bytes long.

Your right, the first example is shorter. But the loop is running l times on average, i.e. 8 times. In the second case, the loop is only 4 times: the values of the operand do not matter.

In fact, you are really right: the first example is better anyway. The reason is that we have chosen to work with only four bits for simplicity. When we multiply two 8 bits numbers, the first method loops 128 times on average (256/2). The second method loops only 8 times (8 bits). So the second one is better when the operands are greater than a break-even value. Look to the exercises below, and send us your solution: better ones will be published here.
Have fun.

Xavier Glattard
xavier.glattard@netcourrier.com

Exercises:

1 - Multiply by Zero.
The example 1.1 do not return Zero when the second number is Zero (in c). Fix it.

2 - 8 bits operands (part 1)
The example 1.1 works with operands bigger than 15 (more than 4 bits). But an overflow occurs if the result does not fit in a byte. Fix it and receive the result in hl (16 bits register).

3 - 8 bits operands (part 2)
Fix the example 1.2 and receive the result in hl.

4 - 8 bits operands (conclusion)
Compare precisely the two methods in term of machine cycle. You may use an opcode table : you would find some on the web.

Links

DEVSEEK Programming Languages Assembly z80
http://www.devseek.com/Programming/Languages/Assembly/z80/

Thomas Scherrer Z80-Family Official Support Page
http://www.geocities.com/SiliconValley/Peaks/3938/z80_home.htm

Z80 Documentation - ticalc.org
http://www.ticalc.org/pub/text/z80/

Zophar's Domain Z80 Technical Documents
http://www.zophar.net/tech/z80.html

ETC - ZiLOG Z80
http://www.mitton.fsnet.co.uk/archive/z80/

Information on the Z80 processor
http://www.classicgaming.com/epr/z80.htm

Z80 Programming

Written by Xavier Glattard

Here you again!? Not yet discouraged? That's good!

Lesson 2 - Dividing

I hope you like mathematics, because this month we will continue with numeric calculation. In the last issue we have studied Multiplication. Now, we are going to learn how to divide one number by an other. This is a few more difficult, so be quiet and read on carefully.

NB: We will solve the exercises about multiplying next month. Send me your work!

We begin this lesson same way we began the previous one. Dividing with Z80 may be quite simple: **(CODE 2.1)**

I guess this can be optimized. But I prefer to keep it simple.

The loop runs between zero and 255 times. Really inefficient, so we have to look for a better way to do that. Again the answer will come from the past. Try a example:

```
 1513 | 14
-14   |----
---   | 108
  113 |
 -112 |
 ---- |
    1 |
      |
```

Ok? Well, we'd better split the problem into small steps:

(a) We have to consider as many figures of the dividend (from left to right) as needed to be greater than the divisor. So we take tow figures : 15 is greater than 14.

(b) We divide 15 by 14: we find 1, that gives the first figure of the quotient (from left to right).

(c) We multiply 14 by 1, then substract the result from 15 : we find 1, the first remainder, that we use in place of the 15. The new dividend is 113.

(d) Return to step (a) while the dividend is greater than the divisor.

The quotient is three figures long, so we need three stages :

```
 1513 | 14        1513 | 14        1513 | 14
-14   |----      -14   |----      -14   |----
---   | 1        ---   | 10       ---   | 108
  1   |            11  |            113 |
      |                |           -112 |
      |                |           ---- |
      |                |              1 |
      |                |                |
```

The second step seems to be different, but it is not. In fact, the very first step is different. Consider this :

(a) Take the 1st figure of the dividend. If it is lesser than the divisor, add a zero at the right of the quotient and re-do step (a) with another figure of the dividend.

(b) The selected figures become a partial dividend. Divide it by the divisor : it gives one figure that we add at the right of the quotient.

(c) The remainder becomes the new partial dividend. Return to step (a) with an other figure of the real dividend

So in fact, the first stage has to be split in to parts :

```
 1513 | 14        1513 | 14
      |----      -14   |----
      | 0        ---   | 01
      |            1   |
      |                |
```

Let's try a example with binary figures :

```
 11011 | 101
-101   |----
----   | 00101
   111 |
  -101 |
  ---- |
    10 |
       |
```

Step (b) is simpler : dividing the 'partial' divisor gives always 1, and subtracting the divisor from the partial

Code 2.1:

```
        ; The dividend is in a
        ; The divisor is in l
        ; The remainder will be in a
        ; The quotient will be in l
        ld c,0 ; Init. Quotient

Begin:
        cp l
        jr c,End ; If l>a then there is nothing more to do
        sub l ; One time : remainder in a
        inc c ; Increment quotient
        jr Begin

End:
        ld l,c ; Put the quotient in l
```

dividender give the 'partial' remainder. Let's go on with the code : **(See CODE 2.2)**

What do you think of that ? The only hard bit is the cpl/rlc opcodes :

- if l (divisor) is greater than a (partial dividend) then we have to add 0 at the right of the quotient ; cp l set the carry, then we just have to complement it and put it at the back of c (the quotient).

- if l is less than a, then we have to add 1 at the right of the quotient ; cp l reset the carry, then we complement it and put it at the back of c.

The loop runs 8 times (values do not matter) and we use 5 registers. The loop is 12 bytes long (one in the conditional part) : 8 x (11+ (1/2)) + 6 = 98 bytes run. That's not too bad. Let's optimize the code : look at h and c. First the line ld c,a is useless : it resets c before use, but c is never read and we put each eight bits one by one. Then let's analyse the values of h and c:

```
b=8 : h=XXXXXXXX, c=........
b=7 : h=XXXXXXX., c=.......y
b=6 : h=XXXXXX.., c=......yy
 ...
b=1 : h=X......., c=.yyyyyyy
b=0 : h=........, c=yyyyyyyy
```

(X,y represent important values, . is a useless bit)

Moreover, we only read from h and write to c. So we could use only one register to store both values:

```
b=8 : c=XXXXXXXX
b=7 : c=XXXXXXXy
b=6 : c=XXXXXXyy
 ...
b=1 : c=Xyyyyyyy
b=0 : c=yyyyyyyy
```

The optimized code: **(See CODE 2.3 next page)**

h is not used any more, and rl h/rl c became rl c (but we must add another one before the loop) : 8 x (10+1/2) + 7 = 91 bytes run. That's better. Here is an other way to do the job: **(see Code 2.4 next page)**

Slightly better: same number of opcodes in the loop (two bytes less in the whole code), but one of them moves to the conditional part, so it is not always run : 8 x (8+2/2) + 5 = 77 bytes run. Yep!

Next month we will work with 16 bits values: adding, substracting, multiplying and dividing. May be some of you already know solutions? Send them to me! Better ones will be published.

xavier.glattard@netcourrier.com

Code 2.2:

```
        ; The dividend is in a
        ; The divisor is in l
        ; The remainder will be in a
        ; The quotient will be in l
        ld h,a ; Dividend in h
        xor a ; Init. partial dividend to zero
        ld c,a ; Init. quotient to zero
        ld b,8 ; 8 binary figures

Begin:
        rl h ; Put a bit of the dividend in the carry
        rla ; ..then add it at the right of the partial dividend
        cp l
        jr c,NextBit ; If the divisor is still greater, we need an other bit
        sub l ; Remainder in a. Carry is still zero

NextBit:
        cpl ; Complements carry
        rl c ; Put the figure in the carry at the right of the quotient
        djnz Begin

End:
        ld l,c ; Put quotient in l
        ; The remainder is in a
```

Code 2.3:

```
        ld c,a ; Dividend in c
        xor a ; Init partial dividend to zero
        ld b,8 ; 8 binary figures
        rl c ; Put a bit of the dividend in the Carry (once only)

Begin:

        rla ; Put Carry at the right of the partial dividend
        cp l
        jr c,NextBit ; If the divisor is still greater, we need an other bit
        sub l ; Remainder in a. Carry is still zero

NextBit:

        cpl
        rl c ; Put carry in bit 0 and put bit 7 in carry
        djnz Begin

End:

        ld l,c ; Put quotient in l
```

Code 2.4:

```
        ld c,a ; Dividend in c
        xor a ; Init partial dividend to zero
        ld b,8 ; 8 binary figures

Begin:
        sla c ; Put zero in bit 0 and put bit 7 in Carry
        rla ; Put Carry at the right of the partial dividend
        cp l
        jr c,NextBit ; If the divisor is still greater, we need an other bit
        sub l ; Remainder in a
        inc c ; Set bit 0

NextBit:

        djnz Begin

End:

        ld l,c ; Put quotient in l
```

Z80 Programming

Written by Xavier Glattard

Yep ! A reader send me a letter ! John, you're not my only reader !

His first name is Sébastien, he is french (I guess) and he argues about the way I evaluate the code for optimization. Sébastien says that the length in bytes of an instruction and its running time are often very different : inc h is 1 byte long and uses 4 cycles , inc hl is also 1 byte long but uses 8 cycles. Sébastien is right : my method is not very accurate. But it is simple, and gives quite good results when comparing similar codes. In this month lesson, we will use both methods.

Like Sébastien you are invited to send me any comments about these lessons and the CPC and the weather and so on : xavier.glattard@netcourrier.com

ERRATUM : In lesson 2, I wrote the mnemonic cpl instead of ccf. The first one complements the accumulator. The second (and good) one complements the carry.

Lesson 3 - Compute with words

In this month lesson is still about basic mathemacital operations : we will learn how to compute with 16-bit words on a 8-bit microprocessor.

There are two 16-bit registers for computing : hl is the more powerfull and may be compared to a 16-bit accumulator, and de is the usual second operand. The register bc is a counter. The other one are mainly used for specific or indexed memory access (remember : the Z80 address bus is 16-bit wide). So we will operate on hl and de.

Adding two words is quite simple : Z80 can do that (add and adc mnemonics). She also can substract two words, but be careful! : She always substract with the carry (sbc memonic) and we need to clear it before use. These instructions operate on hl (or ix/iy) and de (or bc), and the result is store in hl (or ix/iy). The carry is set if needed. Quite simple.

For multiplying we need some more code. We will re-use the code from lesson 1 : we have learn how to multiply two quartets (4-bit words), resulting in a byte (8 bits). We may extend the procedures to larger words. The code is self explained. **(See CODE 3.1)**

This way of doing the work is quite efficient. An other one to shift de to the left is to add de to itself (multiply by 2). But all 16-bit instructions use hl. Anyway, let's try this. **(See Code 3.2)**

The first version is four byte long and use 8+8 machine cycles. the second one is only three byte long but use 4+11+4 cycles. Make your choice.

Again division is more complex. The 8-bit procedure may be re-used, but it needs a lot of adaptations. Not only the result is twice larger (word instead of byte), but the operands are also twice larger. Z80 can't manipulate so many 16-bit values at the same time. We might try to use ix or iy registers : we would have to load it with the value

Code 3.1:

```
        ; The 1st number is in a
        ; The 2nd number is in l
        ; The result will be in hl
        ld d,0
        ld e,l ; l expands to de
        ld h,d
        ld l,d ; 0 to hl
        ld b,8 ; 8 bits

Begin:
        rra ; Rotate and put bit 7 of a in carry
        jr nc,NextBit
        add hl,de

NextBit:
        sla e ; Shift left : bit 7 falls in carry
        rl d ; Rotate left through carry
        djnz Begin

End:
```

Code 3.2:

```
(...)
NextBit:
        ex hl,de ; Swap hl and de
        add hl,hl ; Add hl to itself
        ex hl,de ; Swap back hl and de
        djnz Begin

End:
```

Code 3.3:

```
        ; The dividend is in hl
        ; The divisor is in de
        ; The quotient will be in de
        ; The remainder will be in hl
        ld hl,0
        ld a,h ; a = high byte of dividend
        ld c,l ; c = low byte of dividend
        ld b,8 ; 8 bits
        rla ; Left rotate of a, bit 7 falls into carry

Begin:

        adc hl,hl ; Right shift of hl, carry is pushed into bit 0
        sub hl,de ; Substract de from hl
        jr nc,NextBit ; If hl was greater than divisor then carry is not set
        adc hl,de ; Restore previous value of hl

NextBit:

        rla ; Put carry into bit 0, bit 7 falls into carry
        djnz Begin
        ld b,a ; Use b for swapping a and c
        ld a,c ; a = low byte of divisor
        ld c,b ; c = high byte of quotient
        ld b,8 ; 8 bits
        rla ; Left rotate of a, bit 7 falls into carry

Begin2:

        adc hl,hl ; Right shift of hl, carry is pushed into bit 0
        sub hl,de ; Substract de from hl
        jr nc,NextBit2 ; If hl was greater than divisor then carry is not set
        adc hl,de ; Restore previous value of hl

NextBit2:

        rla ; Put carry into bit 0, bit 7 falls into carry
        djnz Begin2
        ld d,c
        ld e,a ; de = quotient

End:
```

Code 3.4:

```
                ; The dividend is in hl
                ; The divisor is in de
                ; The quotient will be in de
                ; The remainder will be in hl
                ; Negate de
                xor a ; a = 0
                sub e ; a = 0-e
                ld e,a
                sbc a,a ; a = (a-a)-carry = -carry
                sub d ; a = -d-carry
                ld d,a
                ld a,h ; a = high byte of dividend
                ld c,l ; c = low byte of dividend
                ld hl,0
                ld b,8 ; 8 bits
                rla ; Left rotate of a, bit 7 falls into carry

Begin:

                adc hl,hl ; Right shift of hl, carry is pushed into bit 0
                add hl,de ; Substract divisor from hl
                jr c,NextBit ; If hl was greater than divisor then carry is set
                sbc hl,de ; Restore previous value of hl

NextBit:

                rla ; Put carry into bit 0, bit 7 falls into carry
                djnz Begin
                ld b,a ; Use b for swapping a and c
                ld a,c ; a = low byte of divisor
                ld c,b ; c = high byte of quotient
                ld b,8 ; 8 bits
                rla ; Left rotate of a, bit 7 falls into carry

Begin2:

                adc hl,hl ; Right shift of hl, carry is pushed into bit 0
                add hl,de ; Substract divisor from hl
                jr c,NextBit2 ; If hl was greater than divisor then carry is set
                sbc hl,de ; Restore previous value of hl

NextBit2:

                rla ; Put carry into bit 0, bit 7 falls into carry
                djnz Begin2
                ld d,c
                ld e,a ; de = quotient

End:
```

of hl, and use adc ix,ix in the loop, but... this mnemonic does not exist ! Z80 can not add to ix/iy with carry. Other 16-bit registers can not receive the result of an operation, so we would have to use hl, but it is already in use.

We may split the code in two parts and work with the accumulator : each part works with one byte of the dividend. (see CODE 3.3)

Ok ? No.

The previous code uses the mnemonic sub hl,de that does not exist ! We may use the mnemonic sbc hl,de, but it is 2 byte long and uses 15 cycles. And we must clear the carry with or a. Let's think for another way to do the job.

The more efficient 16-bit instruction is add hl,de which is one byte long and use 11 cycles. But we need a substraction... Hey ! A substraction IS an addition ! Let's negate de before the substraction. Here is one of the fastest ways to do this. (see Code 3.4)

Yes, it works, believe me. You may try an example. Note that the jr nc,NextBit becomes jr c,NextBit.

The code of the 16-bit division is quite long. We said that Z80 can not manage so many 16-bit value at the same time. We was wrong : a 16-bit register is made of two 8-bit registers, so we may create a new 16-bit register. The a and c registers are available : let's use them as ac 16-bit register. (see CODE 3.5)

The code is shorter, but what about speed ? We added the rl c mnemonic (8 cycles) in the loop (68 cycles) : +12%. Again, you have to choose.

Next month we will begin to learn high level programming in assembly language : procedure and procedure call, function, and even (later) class and virtual method call. Your Z80 will warm up.

Code 3.5:

```
                ; The dividend is in hl
                ; The divisor is in de
                ; The quotient will be in de
                ; The remainder will be in hl
                ; Negate de

                xor a ; a = 0
                sub e ; a = 0-e
                ld e,a
                sbc a,a ; a = (a-a)-carry = -carry
                sub d ; a = -d-carry
                ld d,a

                ld a,h
                ld c,l ; ac = dividend

                ld hl,0
                ld b,16 ; 16 bits
                rl c
                rla ; Left rotate of ac, bit 15 falls into carry

Begin:

                adc hl,hl ; Right shift of hl, carry is pushed into bit 0
                add hl,de ; Substract divisor from hl
                jr c,NextBit ; If hl was greater than divisor then carry is set
                sbc hl,de ; Restore previous value of hl

NextBit:

                rl c
                rla ; Left rotate of ac, bit 15 falls into carry
                djnz Begin
                ld d,a
                ld e,c ; de = quotient

End:
```

THE

DiGITAL alternative

Coming Soon...

Z80 Programming

Written by Xavier Glattard

We begins this month to learn GOOD programming with assembly language. Yes ! It is possible to write good program in assembly language ! Easy to write and easy to understand. We only need some tools, some tips and a methodology. Let's start with the art of procedure call.

Lesson 4 - Procedure call (part 1)

First we need to answer a question : what is a procedure ? A procedure is a piece of code that may be run at any time you want in your program. For example the code that multiply two bytes should be in a procedure.

When you have to call a procedure you often need to transmit some values : name of the file to be opened, size of the box to be drawn, etc. These values are called parameters or arguments. The arguments may be transmitted in many different ways : in global variables (that may be used anywhere in the program), in registers (which are fast global variables), in dynamically allocated variables.

In Locomotive Basic (and old languages like Fortran) we only have global variables. The example below uses a sub program (a 'basic' procedure) to draw a box. The procedure needs to know the position and the size of the box. These values are stored in (global) variables by the main program. **(See CODE 4.1)**

Code 4.1:

```
100   BOXX=50:BOXY=20:BOXW=110:BOXH=45
110   GOSUB 1000:REM Draw a box
120   END
1000  REM Draw a box
1010  MOVE BOXX,BOXY
1020  DRAWR BOXW,0
1030  DRAWR 0,BOXH
1040  DRAWR -BOXW,0
1050  DRAWR 0,-BOXH
1060  RETURN
```

We can do the same with assembly language (you may write it down as an exercise). But we should better use registers that are faster. **(See CODE 4.2)**

We have learned how to negate a 16-bit value in **lesson 3 (CODE 3.4)**. We have to use the ix register and then the stack to load the value in an other register.
C language (and Pascal) uses dynamically allocated variables. **(See CODE 4.3)**

Which is the better way ? That's hard to say : each language is good in its way : dynamic allocation is quite difficult with Basic (not impossible). Let's try a more complicated example. We now have to draw something like a checkboard : some boxes will be drawn as rows and columns, with space between them. First, the Basic program. **(See CODE 4.4)**

Don't even think of coding this program in assembly language using only registers : there just are not enough registers ! We could use global variables, as we do in Basic, but we would not learn anything. Moreover global variables have a lot of disadvantages.

First, we need to find specific names for each variable so all the names will be different. This cause long names, hard to type, that may cause some mistakes. All these names have to be managed by your assembler program and slow it down.

Global variables use a lot of memory : they are statically allocated (at program loading). The memory they occupied can not be freed, even if it is used only once.

Procedures can not be called recursively when you use global variables : this is the great difference between a Basic sub program and a C procedure. You can live without recursively, but you need to know this limit.

Most of all, global variables are quite slow : the ld hl,(nn) instruction uses 5 M cycles and 16 T ; ld de,(nn) uses 6 M cycles and 20 T states ! You often code with assembly language for its speed, so you rather have to use registers.

Code 4.2:

```
Begin:

        ld  hl,50       ; X position
        ld  de,20       ; Y position
        ld  bc,110      ; width
        ld  ix,45       ; height
        call DrawBox

End:

        ...
        DrawBox:
        call Move    ; Move to (hl,de)

        ld  h,b             ; ld hl,bc
        ld  l,c
        ld  de,0

        call Drawr   ; Drawr to (hl,de)

        push ix             ; ld de,ix

        pop de

        ld  hl,0
        call Drawr   ; Drawr to (hl,de)


        ld  h,b             ; ld hl,bc
        ld  l,c
        call NegateHL
        ld  de,0
        call Drawr   ; Drawr to (hl,de)

        push ix             ; ld hl,ix

        pop hl
        call NegateHL
        ex  hl,de
        ld  hl,0
        call Drawr   ; Drawr to (hl,de)

        ret
```

How do we allocate memory dynamically, with no use of ld rr,(nn) - like instructions ? When we allocate 'dynamically' we do not know at coding time exactly where the value will be stored. The address of the allocated memory will be computed at running time and stored in a 16-bit register. The simplest and best place to allocate is the stack ; the associated register is sp.

There are two way of building a 'stack frame' where procedure arguments will be stored : the 'C' way and the 'Pascal' way. In the Pascal way we push arguments in the logical order we choose and the stack frame is freed before the procedure returns ; in the C way, arguments are store in the reverse order and the stack frame is freed after the procedure returns. For example, with the draw_box procedure, the arguments are : (x,y,w,h); in the C stack frame we push the height first ; in the Pascal stack frame we begin with the x position. In fact, since the stack grows bottomward, arguments are in the logical order in the C stack frame, and in the reverse order in the Pascal stack frame. Let us use the C way (we will see a smart property of the C stack frame in a further lesson). **(See CODE 4.5)**

This is over simple, not really simple and quite hard to read. Is it fast ? Each argument read needs at least two ld and two inc/dec : 2 x (2M + 7T) + 2 x (1M + 6T) = 6M + 26M. Let us try an other way. **(See CODE 4.6)**

That's far more easy to understand ! If you use EQU variables your code is easier to maintain ; you can even switch to Pascal stack frame with only minor changes. And source code is shorter (42 lines against 55). Each variable read is only two ld : 2 x (5M+19T) = 10M+38T. That seems slower.
In the first case we use hl as a 'stack frame pointer'. In the 2nd case we use ix, so hl may be use for any other computing : in the first code we have to swap hl and de before we negate the value ; it takes 4 x (1M+4T) = 4M+16T, so the sum is 10M+42T !! What do you think about that ? In some case the difference may be even greater. Indexed registers are not so bad.

In the next issue we will see more complex uses of stack frames : large objects, structured objects, pointers, pointers to structured objects and local variable allocation. Nothing really difficult : only real world examples.

Don't forget your sun lotion.

Xavier Glattard - xavier.glattard@netcourrier.com

Code 4.3:

```
void draw_box( int x, int y, int w, int h )
{
        move(x,y);
        drawr(w,0);
        drawr(0,h);
        drawr(-w,0);

drawr(0,-h);
}



int main() {
draw_box(50,20,110,45);

return 0;
}
```

Code 4.4:

```
100 BOARDX=40:BOARDY=30:REM Bottom left corner of the board
110 BOARDHSPC=5:BOARDVSPC=5:REM Vertical and horizontal
spaces between boxes

120 BOARDROWS=4:BOARDCOLS=5:REM Numbers of rows and
columns

130 BOARDBOXW=20:BOARDBOXH=20:REM Size of the boxes140
GOSUB 2000:REM Draw board150 END1000 REM Draw a box(...)

2000 REM Draw a board

2010 FOR BOARDROW=1 TO BOARDROWS
2020 FOR BOARDCOL=1 TO BOARDCOLS
2030 BOXX=BOARDX+(BOARDBOXW+BOARDVSPC)*(BOARDCOL-1)
2040 BOXY=BOARDY+(BOARDBOXH+BOARDHSPC)*(BOARDROW-1)
2050 BOXW=BOARDBOXW
2060 BOXH=BOARDBOXH
2070 GOSUB 1000:REM Draw a box
2080 NEXT BOARDCOL
2090 NEXT BOARDROW
2100 RETURN
```

Code 4.6:

```
DrawBox: ; int x, int y, int w, int h
BoxX EQU 0
BoxY EQU 2
BoxW        EQU 4
BoxH EQU 6

        push ix
        ld  ix,2+2          ; two bytes for the return address,
                        and two more for previous value of ix

        add ix,sp           ; ix = address of the first argument

        ld l,(ix+BoxY)     ; hl = Y position
        ld h,(ix+BoxY+1)
        push hl
        ld l,(ix+BoxX)     ; hl = X position
        ld h,(ix+BoxX+1)
        push hl
        call Move
        ld hl,0
        push hl
        ld l,(ix+BoxW)     ; hl = width
        ld h,(ix+BoxW+1)
        push hl
        call Drawr
        ld l,(ix+BoxH)     ; hl = height
        ld h,(ix+BoxH+1)
        push hl
        ld hl,0
        push hl
        call Drawr
        ld hl,0
        push hl
        ld l,(ix+BoxW)     ; hl = width
        ld h,(ix+BoxW+1)
        call NegateHL
        push hl
        call Drawr
        ld l,(ix+BoxH)     ; hl = height
        ld h,(ix+BowH+1)
        call NegateHL
        push hl
        ld hl,0
        push hl
        call Drawr
        pop ix
        ret
```

Code 4.5:

```
Begin:
        ld  hl,45       ; height
        push hl
        ld  hl,110      ; width
        push hl
        ld  hl,20       ; Y position
        push hl
        ld  hl,50       ; X position
        push hl
        call DrawBox

        ld hl,4*2   ; 4 arg.
        or a        ; carry = 0
        sbc hl,sp
        ld sp,hl    ; free the stack frame
End:
        ...
        DrawBox:        ; int x, int y, int w, int h
        push hl
        ld hl,2+2   ; two bytes for the return address, and two more
for previous value of hl

        add hl,sp   ; hl = address of the first argument (X pos)

        ld c,(hl)   ; bc = X position
        inc hl
        ld b,(hl)
        inc hl      ; hl = address of the 2nd arg. (Y pos)
        ld e,(hl)   ; de = Y position
        inc hl
        ld d,(hl)
        push de
        push bc
        call Move

        ld de,0
        push de
         inc hl     ; hl = address of the 3rd arg. (width)
        ld e,(hl)   ; de = width
        inc hl
        ld d,(hl)
        push de
        call Drawr
        inc hl      ; hl = address of the 4th arg. (height)
        ld e,(hl)   ; de = height
        inc hl
        ld d,(hl)
        push de
        ld de,0
        push de
        call Drawr

        ld de,0
        push de
        dec hl
        dec hl
        ld d,(hl)
        dec hl      ; hl = address of the 3rd arg. (width)
        ld e,(hl)   ; de = width
        ex de,hl
        call NegateHL
        ex de,hl
        push de
        call Drawr
        inc hl
        inc hl      ; hl = address of the 4th arg. (height)
        ld e,(hl)   ; de = height
        inc hl
        ld d,(hl)
        ex de,hl
        call NegateHL
        ex de,hl
        push de
        ld de,0
        push de
        call Drawr
        pop ix
        ret
```

BONZ — The Game!

Check out Issue 10 of CPC Oxygen Online (cpcoxygen.digi-alt.net) for another Z80 programming article from Xavier Glattard. This time it's Bonz, a complete game programmed in just 2 kilobytes of the finest Z80 code! Full source code is included in the article as well as the tutorial itself which explains how it all work together.

Copychr$ to the Fore

from issue 5 of CPCO Online

Written by Frederik Leighton

The Basic program ZIP.BAS, 2K only, is pretty handy to CAT through your disks using the arrow keys. Better still, you don't really need any of the following explanations to use it. Anyway, it consists of two parts: a DEMO and a MODULE.

DEMO in lines 10-1000.

As is, allows you to chain CATs while switching disks until you Run or Load a "filename=C$" from disk or Quit. There is space for your code here. Pick your colours in Line 90 or REM them out.

The DEMO calls module **zipcat.cpc** in line 200. After the GOSUB 64000, the validated file name is returned. The program now waits on your choice; hit R, L, M, C or Q to get better than a BEEP.

Lines 310 and 320: testing the string C$ before RUN or LOAD. An empty string would cause the program to exit. Better to start all over again.

ZIPCAT.CPC 1K, a Basic only module in lines 64000-.

It returns a filename you may select on screen without typing. You will exit back to the calling program with the goods in string C$.

WHAT IT DOES NOT DO:

The module runs on CPC 6128 only and has limitations, as it is written using the minimum of Basic code to get the job done. To run on CPC 464 it will need some additions (see boxout, 'What about the 464!'). Maximum number of files that can be shown is 64, the 3" disk limit. It will only function in MODE 2 and it doesn't care what it returns. Behaviour in emulation is unknown.. feedback welcome.

WHAT IT DOES DO:

Apart from the above it does everything… it was made to do. The module supports GOSUB, MERGE and RENUM.

It lets you cycle through the file names any which way, using the "Arrow" keys to move a Reverse Video Cursor around. The "Return" key is the only way back out; you may use it any time. Hitting key "C", checked in the INKEY loop, will repeat CATs on & on. Don't fall asleep now with your finger on the thing, or you will have to send off for a new drive belt.

WHAT YOU DO:

At this point you know enough to just type in the program and use it. It's fairly nifty all by itself and the rest is just the boring bits.

Careful, save before all testing as every Run C$ or Load C$ etc. will erase your new code from RAM. Type SAVE"ZIP.BAS to write it to disk, RUN"ZIP to use it, LOAD"ZIP to modify it.

If you have a hard time getting things to work check the punctuation again. The placing of the semicolons ";" is very important, as that exactly fixes the location of the next PRINT commands.

No joy messing with the module but do try new stuff in the calling program. Experiments involve files, so use a disk with bogus data. RENAME is possible, same for "Erase all *.BAK files in one go" and ERASE itself. But whisking files away by just hitting "E" is a bit hairy, so put in a solid

confirmation and back out routine.

WHAT IT TAKES:

Your Mum used to say; "Now you just eat all your variables, dear", and she was right all along. Now it's your turn to care for a little program… It's still too small for those reals!, remember. Lots of integers% and some string$ will do.

C$ - holds the current selected filename.

c%, z% - simple placeholders or loop counters.

x%, y% - current screen co-ordinates, as in LOCATE x%,y%.

x2%, y2% - right and bottom limits of the cursor box, values will vary.

xb%, yb% - hold the prior co-ordinates of the cursor.

bk$ - holds the prior, original filename.

fb% - flag bak ; fb%=1 forces a screen cleanup - the prior filename in string bk$ is printed back at prior location xb%,yb%.

HOW IT SEEMS TO BE DONE:

COPYCHR$(#n), a wallflower which is really Cinderella. Waltzing her around the screen will give you a look at one character after another.

Successively reading & assembling the CHR$(x) present at each location will COPY your screen's character content. All without having to look for the weird bits inside the machine to get at it.

As used here, the set-up treats an estimated 4 filenames per second or 48 chars/s. When given a straight run reading whole chunks of the screen everything is much faster. Gimme the PEEKs, hold the POKEs.

Typical use: #0 being the normal screen

```
10 LOCATE some%,where%:char$=COPYCHR$(#0)
20 LOCATE here%,there%:PRINT char$
```

COPYCHR$ is called the first time in lines 64020 and 64030 to map the screen layout. The filename spacing in CATs and ¦DIRs can be counted out. If a dot "." is present so is a file.

This sets the variables x2%, right cursor limit and y2%, bottom cursor limit. In this module the left cursor limit=1 and the top cursor limit=7, are invariable. These boundaries are to be tested further on, to keep the cursor from zooming off the screen with a file name or making a nuisance of itself somewhere else. That is done in lines 64200 and 64210, looping back when a limit is exceeded.

COPYCHR$ is used again in line 64300. The characters are read and fed in one by one to spell out a file name in string C$. That's called concatenation.

Filename - total 12 characters, 8 for name, 1 dot, 3 for extension. Using the on/off switch CHR$(24), string C$ will overwrite the onscreen filename image with Reverse Video in line 64410. Don't worry, before leaving here, setting fb%, xb%, yb% and string bk$ will allow us to put things back as they were, so nobody can tell we had it all messed up.

RELOOKING: Don't do it.

Line 64220 is there so you don't tear your hair out when there is not more than a single line of filenames. REM it out and try; where have all the files gone? Copychr$ can't read Reverse Video in MODE 2 and returns with the null value CHR$(0). The result is garbage on screen.

A MONEY$ SAVING TIP:

One "UP arrow" & one "LEFT arrow" will take the cursor diagonally from top-left to bottom-right. This avoids a gazillion operations in a large CAT screen and so will save you lots of TIME=money$.

NEXT:

Having come this far you may not have wasted your time after all.

Next up: CAT.BAS,.3K. Same but extra bits.
CATs & ¦DIRs (*.*) + the MODE with the pretty colours & joystick too.

```
10 REM ** ZIP.BAS **
20 'Lines 10-1000 are a DEMO calling ZIPCAT.CPC in lines 64000-
30 '
90 INK 0,15:INK 1,0:BORDER 15
100 GOSUB 200:CLS:END
200 GOSUB 64000 'Zipcat.cpc will return filename C$
210 LOCATE 5,2:PRINT"Run Load More Cat Quit"
300 tex$="RLMCQ":GOSUB 900:ON k GOTO 310,320,330,340,350
310 IF C$=SPACE$(12) THEN 340 ELSE CLS:RUN C$
320 IF C$=SPACE$(12) THEN 340 ELSE LOAD C$
330 LOCATE 5,2:PRINT CHR$(18);"Your Code Here..":FOR zip=1 TO 2000:NEXT
340 GOTO 200
350 RETURN
360 '
900 REM * Merci Monsieur M. Archambault
910 r$="":WHILE r$="":r$=UPPER$(INKEY$):WEND
920 k=INSTR(tex$,r$):IF k=0 THEN PRINT CHR$(7):GOTO 910
930 RETURN
1000 '
64000 REM ** module ZIPCAT.CPC V 1.0 2002 (c) by F. Leighton **
64010 MODE 2:fb%=0:x%=1:y%=7:x2%=61:y2%=25:LOCATE 1,4:CAT
64020 FOR z%=8 TO 25:LOCATE 9,z%:IF COPYCHR$(#0)="." THEN NEXT ELSE y2%=z%-1
64030 FOR c%=29 TO 69 STEP 20:LOCATE c%,y%:IF COPYCHR$(#0)="." THEN NEXT ELSE x2%=c%-28
64040 LOCATE 30,y2%+2:PRINT CHR$(24);" * C=Cat * ";CHR$(24);
64100 GOSUB 64200
64110 IF INKEY(2)<>-1 THEN y%=y%+1:GOTO 64100
64120 IF INKEY(0)<>-1 THEN y%=y%-1:GOTO 64100
64130 IF INKEY(1)<>-1 THEN x%=x%+20:GOTO 64100
64140 IF INKEY(8)<>-1 THEN x%=x%-20:GOTO 64100
64150 IF INKEY(18)<>-1 THEN LOCATE 30,y2%+2:PRINT C$;:GOTO 64500
64160 IF INKEY(62)<>-1 THEN 64000 'C
64170 CALL &BB06:GOTO 64110
64200 IF y%<7 THEN y%=y2% ELSE IF y%>y2% THEN y%=7
64210 IF x%<1 THEN x%=x2% ELSE IF x%>x2% THEN x%=1
64220 IF y%=yb% THEN IF x%=xb% AND fb% THEN 64500
64300 C$=SPACE$(12):z%=0:FOR c%=x% TO x%+11:z%=z%+1:LOCATE c%,y%:MID$(C$,z%,1)=COPYCHR$(#0):NEXT
64400 IF fb% THEN LOCATE xb%,yb%:PRINT bk$;ELSE fb%=1
64410 LOCATE x%,y%:PRINT CHR$(24);C$;CHR$(24);:xb%=x%:yb%=y%:bk$=C$
64500 fr!=FRE(""):RETURN
```

What about the 464!

As you may know the 1984 version of the 464 only came with Basic 1.0 instead of 1.1 ad therefore unable to handle a few extra commands present in other CPC's. To fix this we found a routine on Sean McManus website at http://www.sean.co.uk/a/amstrad/index.shtm , thanks to Sean McManus for that.

'(C) Sean McManus'

```
10 DATA dd,7e,00,cd,b4,bb,f5,cd,60,bb,32,00,00,f1,c3,b4,bb
20 addr=40000:FOR g=0 to 16:READ a$:POKE addr+g,VAL("&"+a$):NEXT
```

The program can be placed anywhere in memory by changing the addr value in line 20. When you see a copychr$ command like this:

```
a=COPYCHR$(#0)
```

replace it with the following:

```
CALL 40000,0:a=PEEK(0)
```

The returned value will be store at address 0 so you can get that by pointing the peek command to line zero, simple. if someone does get it going on a 464 with a disk drive (I don't have one) then let us know.

Copychr$ over the top

from issue 6 of CPCO Online

Written by Frederik Leighton

HARKING BACK

In ZIP.BAS, CPCO issue #5, we took COPYCHR$ for a spin around the block. The interesting part there was the module ZIPCAT.CPC, reading and returning an onscreen file name for use in any main program. "First principles, Clarice, first principles..", as the digestively favoured Dr. Lecter puts it. Think.. What do we covet..? Why, a better interface of course!

Please refer back to CPCO issue #5 for a minimalist look at the set-up & its associated comments. The following takes things from there and might be hard to grasp otherwise. One immediate correction to ZIP.BAS in LINE 920. After … PRINT CHR$(7) place a semicolon, to read … PRINT CHR$(7);:GOTO 910. If not, repeat beeping will scroll things out of whack.

SURF'S UP

Here, in CAT.BAS, Listing #1, we're dressed to party and hot to trot. This boosted CAT is a bit of a workhorse, allowing you to generally scoot around your disks using "Arrow" keys or Joystick. Since all of its features are non-risk they can be activated with one key press to speed things along. It can be used as a standalone program or a "mergeable" module both, without any modifications. This means there is an overhead of about 15% in code size compared to single use versions. However, after cutting all corners everything comes in at 3K which still allows both dissemination throughout your disk library & fast chain merging.

Remember, for the program to be useful as a standalone no understanding of its workings are required. This as a thoughtful gesture to GAMERS everywhere ;-) CATDEMO.BAS, Listing #2, shows how to CHAIN MERGE using CAT.BAS to print handy CATs & DIRs and read some of those pesky ASCII files. In the DEMO you will also find some Fast Edit keys, which might seem out of context. However it is possible to obtain a very efficient Load, List, Edit, Save & Run cycle. If the program is used as a module then CATCALL.BAS, Listing #3, will help you to get a "real time" handle on the variables used to steer things.

STANDALONE USE WALKTHROUGH

Simply RUN"CAT. Then move between filenames and

select using the "Arrow" keys or your Joystick. The menu bar provides the following options:

X..SavEra..Cat..Quit..Ret*Fire=Run..Load. The dots stand for spaces. There should be two spaces between every option, which is hard to figure from the code. This way everything will shoehorn into a single line in MODE 1 (see pic) and look decent in MODE 2.

Colour Options: Not in menu bar. Use "O" & "P" keys (0-Paper) to set PAPER & BORDER to any colour from 0 to 26. Use "U" & "I" keys (Under-Ink) to write in any colour with INK 1. Current colours are printed on screen bottom left. The default colours can be edited in LINE 60040. p% for PAPER, k1% for INK 1. When used as a module the calling program's colours remain unchanged, only k3% for INK 3 is always forced to make sure file selection will show in MODE 1.

In fact, the first four lines are written to allow easy editing of the default settings:

Line 60010, set dir$ to your choice, like "*.BAK" to see only BAK files in DIR screens. Otherwise default dir$ will turn into "*.*". Any dir$ may later be modified on the fly and will stick as long as the program is in memory. This allows for "file name searching" over many disks. When editing new dir$ (Line 60630) the SHIFT+ COPY method can be used for easier modification. Line 60020, j% for job. j%=1 by default for CAT. j%=2 will force a DIR on start-up. Line 60030, m% for MODE. m%=2 by default for a MODE 2. m%=1 will force MODE 1 on start-up. If you change these defaults to suit a particular disk, be sure to save that version of CAT.BAS to that disk before going on.

MENU BAR OPTIONS

X - Hitting "X" key the first time will take you to a subroutine in Lines 60550-. Now, every time you hit "X" you will cycle through the same four choices: Cat mode 1, Cat mode 2, Dir mode 1 or Dir mode 2. Hitting "Return" accepts current choice. In case of a DIR you may keep the current dir$ by hitting "Return" again. To modify a dir$ press the first character of the new dir$ TWICE. The first press will force a CLS in Line 60630, the second gets you going. Yup, a cut corners critter there. Note: If a large number of files causes the screen to scroll in MODE 1 then this will be detected in Line 60200 and things will loop into MODE 2.

SavEra - Hitting "S" key will Save the CAT.BAS currently in memory to disk. Hitting "E" key will Erase CAT.BAS from current disk. This means you can quickly transfer it to any disk, load, run and modify programs using it and then remove it anytime. As long as a CAT.BAS is on the current disk the key combination CONTROL+COPY will run it (Line 60050). No need for any kind of hacking to switch between programs. "The medium is the menu." Let no more be said.

Cat - Hitting "C" key will repeat the current type of CAT or ¦DIR,dir$. You are free to swap disks anytime.

Quit - Hitting "Q" key simply ends the program.

Ret*Fire=Run - "Return" key or Joystick Fire button will RUN selected program. Here again GAMERS come into their own. Adapt the defaults to suit your game disks. Use a dir$ set to the extension of your game file(s), use MODE 1 with your very own pretty colours and letters big enough to actually read. Now without ever letting go of your joystick you can hop from one game to the other. If you ask your kid sister nicely she will set things up just so. I said NICELY!

Load - Hitting "L" key will LOAD selected program and also set the "f7" key to print LIST. Sequence: use CAT.BAS to select a program, hit "L", hit "f7", and hit "Return". You can be looking at a listing within seconds from putting a disk in the drive! Debug your first lines of code before your Wintel box has stopped looking for its missing start-up files. Every time.

Note: Case of Run or Load - If current filename contains only spaces we will set a flag empty e%=1 in Line 60500. Testing e% avoids running or loading an empty string, which would halt the program.

MODULE USE WALKTHROUGH

This is how we used ZIPCAT.CPC in issue #5, check back for the analogies. In fact if your main program just needs a filename and nothing else, use that module. Its size in lines 64000- is only 1K, not exactly bloatware.

CAT.BAS brings along more capabilities instantly available to your programs, but you trade for 3K size. Since CAT.BAS is supposed to be disseminated over your "active disks" anyway, it is well suited to CHAIN MERGE from smaller programs as needed. See CATDEMO.BAS, Listing #2 for this type of use.

The program supports RENUM. Can be joined to an existing program using MERGE or CHAIN MERGE. Then use GOSUB to call the module. You MUST pass the variable ret=1 to force module behaviour.

In the Menu Bar the option SavEra is replaced with the option SELECT!

Select - Hitting "S" key will RETURN to calling program with nul=0 and selection in string C$.

Quit - Hitting "Q" key will RETURN to calling program with nul=1 & C$="".

Coding sequence in general:
After MERGE or CHAIN MERGE use following algorithm:

' Sub
ret=1:GOSUB module
IF nul THEN nothing, exit routine, GOTO return 'Quit
ELSE do your stuff with filename C$ here 'Select RETURN

You may pass m%, j% and dir$ with your own values to override the defaults in CAT.BAS. For example j% =2:dir$="*.ABC" would give a DIR showing only the files created by application ABC etc.

If you find yourself this far involved in things it might help to type-in CATCALL.BAS, Listing #3 and experiment some with your own values in Line 60. After using Select or Quit option in CAT.BAS, a screen with current settings will be shown. Otherwise just reading that Listing may be instructive, as it was written to provide a snapshot of the value of all useful variables. After mucking about with the colours using the "UIOP" keys you get to see a line with the new colours as well.

MORE ON COLORS

Another cut corner here, by the way. When in use as a module, hitting one of the colour keys for the first time, will actually "offset" you by one from the CAT.BAS default. Just hit opposite key to get back if needed (after hitting P use O etc.)
Waffle, waffle, which colour is the prettiest of all? You decide after fiddling to your hearts content. The default colours have been picked to be flicker-free and readable when writing code in MODE 2, as you will be stuck with them after a Load-List. Not necessarily nice, what? By the way, I love the MODE 1 screen with the CPC default colours as in CATCALL.BAS, yellow on blue with red file selection. Whoopee.. as it were. However, the result in MODE 2 is not acceptable and fiddle it is.

MORE VARIABLES

Compared to ZIPCAT.CPC the following variables have also been added:

y1% - top limit of the cursor block.

xx% - the distance between the beginning of filenames on the same line.

v% - value 0 or 24. This allows for activating Reverse Video in MODE 2 only.

t% - contains the count of files VISIBLE on screen (Lines 60240-260).

Depending on the use of MODE 1 or 2 and on the use of CAT or DIR we will have to pick appropriate values from DATA lines for certain variables.

MORE RIGMAROLE

In Lines 60500-520 we turn any empty C$ containing 12 spaces into a "square" printable in MODE 1 or into one space for MODE 2. Also picking PEN 3 here gives us a distinctive colour to signify selection in MODE 1 without interfering with anything in MODE 2 (PEN 1 = PEN 3).

NEXT

Might as well let the CAT : out of the bag; this is all leading up to a working, BASIC only, programming environment where we'll be using COPYCHR$ to read both filenames AND menu commands from screen while providing more useful fast edit keys.

Look for DOS.BAS to streamline code writing & file handling. Run"DOS and (almost) never type another command line! Honest.

AUTHORS NOTE

Current CAT.BAS is a full rewrite from 15-year-old code and has had a few extras squeezed in. If you are still an active CPC user then I think it is truly a big time saver and smoother-out of things. Besides, it has been known to reduce bedwetting and nail biting in GAMERS, as well as make life easier for their kid sisters! This stuff was made up for lack of anything comparable among the type-ins at the time and I wonder if anybody has come across similar code before. Your comments and suggestions are more than welcome. Especially since it is John who will be fielding them.

COPYCHR$

```
60000 REM ** CAT.BAS (c)2002 by F.Leighton
60010 IF ret=0 THEN dir$=""
60020 IF j%<1 OR j%>2 THEN j%=1
60030 IF m%<1 OR m%>2 THEN m%=2
60040 p%=1:k1%=16:k3%=6:INK 3,k3%:IF ret=0 THEN INK 0,p%:INK 1,k1%:BORDER p%
60050 KEY 158,"RUN"+CHR$(34)+"CAT"+CHR$(13):KEY DEF 9,1,224,224,158 '^+Copy
60060 dir$=UPPER$(dir$):IF dir$=""THEN dir$="*.*"
60070 ON j% GOTO 60080, 60120
60080 DATA 21,5,20,0
60090 DATA 61,6,20,24
60100 IF m%=1 THEN RESTORE 60080 ELSE RESTORE 60090
60110 GOTO 60150
60120 DATA 16,5,15,0
60130 DATA 61,6,15,24
60140 IF m%=1 THEN RESTORE 60120 ELSE RESTORE 60130
60150 READ x2%,y1%,xx%,v%
60160 PAPER 0:PEN 1:MODE m%:LOCATE (m%-1)*17+1,m%:v$=CHR$(v%)
60170 PRINT v$;SPC(m%-1)"X  ";:IF ret THEN PRINT"Select"; ELSE PRINT"SavEra";
60180 PRINT"  Cat  Quit  Ret*Fire=Run  Load"SPC(m%-1)v$
60190 nul=0:fb%=0:x%=1:y%=y1%:LOCATE 1,y1%-3:IF j%=1 THEN CAT ELSE |DIR,dir$
60200 LOCATE 9,y1%-1:IF COPYCHR$(#0)<>CHR$(32)THEN m%=2:GOTO 60070
60210 FOR z%=y1%+1 TO 25:LOCATE 9,z%:IF COPYCHR$(#0)="."THEN NEXT ELSE y2%=z%-1
60220 FOR c%=xx%+1 TO x2% STEP xx%:LOCATE c%+8,y1%
60230 IF COPYCHR$(#0)="."THEN NEXT ELSE x2%=c%-xx%
60240 t%=0:FOR z%=y1% TO y2%:FOR c%=9 TO x2%+8 STEP xx%
60250 LOCATE c%,z%:IF COPYCHR$(#0)="."THEN t%=t%+1
60260 NEXT:NEXT:LOCATE 21,y1%-2:PRINT"Files:";t%
60270 IF j%=2 THEN LOCATE 16,y2%+2:PRINT"Dir: ";dir$
60280 GOSUB 60450
60290 IF INKEY(2)<>-1 OR INKEY(73)<>-1 THEN y%=y%+1:GOTO 60280
60300 IF INKEY(0)<>-1 OR INKEY(72)<>-1 THEN y%=y%-1:GOTO 60280
60310 IF INKEY(1)<>-1 OR INKEY(75)<>-1 THEN x%=x%+xx%:GOTO 60280
60320 IF INKEY(8)<>-1 OR INKEY(74)<>-1 THEN x%=x%-xx%:GOTO 60280
60330 IF (INKEY(18)<>-1 OR INKEY(76)<>-1)AND e%=0 THEN CLS:PRINT C$:RUN C$ 'Ret
60340 IF INKEY(63)<>-1 THEN GOSUB 60550:GOTO 60060 'X
60350 IF INKEY(60)<>-1 THEN IF ret=0 THEN CLS:PRINT"Save":SAVE"cat":GOTO 60070 E
LSE IF e%=0 THEN 60540 'S
60360 IF INKEY(58)<>-1 AND ret=0 THEN CLS:PRINT"Era":|ERA,"cat.bas":GOTO 60070'E
60370 IF INKEY(62)<>-1 THEN 60070 'C
60380 IF INKEY(67)<>-1 THEN CLS:IF ret THEN nul=1:C$=CHR$(0):GOTO 60540 ELSE END
60390 IF INKEY(36)<>-1 AND e%=0 THEN CLS:KEY 7,"LIST":PRINT"Load: ";C$:LOAD C$'L
60400 IF INKEY(42)<>-1 AND k1%>0 THEN k1%=k1%-1:INK 1,k1% 'U
60410 IF INKEY(35)<>-1 AND k1%<26 THEN k1%=k1%+1:INK 1,k1% 'I
60420 IF INKEY(34)<>-1 AND p%>0 THEN p%=p%-1:INK 0,p%:BORDER p% 'O
60430 IF INKEY(27)<>-1 AND p%<26 THEN p%=p%+1:INK 0,p%:BORDER p% 'P
60440 LOCATE 1,y2%+3:PRINT k1%;p%;:CALL &BB06:GOTO 60290
60450 IF y%<y1% THEN y%=y2% ELSE IF y%>y2% THEN y%=y1%
60460 IF x%<1 THEN x%=x2% ELSE IF x%>x2% THEN x%=1
60470 IF y%=yb% THEN IF x%=xb% AND fb% THEN 60540
60480 C$=SPACE$(12):z%=0
60490 FOR c%=x% TO x%+11:z%=z%+1:LOCATE c%,y%:MID$(C$,z%,1)=COPYCHR$(#0):NEXT
60500 e%=0:IF INSTR(SPACE$(12),C$)THEN e%=1:IF m%=1 THEN C$=CHR$(143)ELSE C$=" "
60510 IF fb% THEN LOCATE xb%,yb%:PRINT bk$;ELSE fb%=1
60520 PEN 3:LOCATE x%,y%:PRINT v$;C$;v$;:IF C$=CHR$(143)THEN C$=" "
60530 xb%=x%:yb%=y%:bk$=C$
60540 PEN 1:fr!=FRE(""):RETURN
60550 IF NOT INKEY(63)<>-1 THEN 60610 ELSE CLS:z%=z%+1:IF z%>4 THEN z%=1 'X
60560 IF z%=1 THEN j%=1:m%=1:PRINT"C a t__1"
60570 IF z%=2 THEN j%=1:m%=2:PRINT"C a t__2"
60580 IF z%=3 THEN j%=2:m%=1:PRINT"D i r__1"
60590 IF z%=4 THEN j%=2:m%=2:PRINT"D i r__2"
60600 PRINT:PRINT"Return to select"
60610 CALL &BB06:IF NOT  INKEY(18)<>-1 THEN 60550 'Ret
60620 IF j%=2 THEN PRINT:PRINT"Return to keep DIR ";dir$:CALL &BB06 ELSE 60640
60630 IF NOT INKEY(18)<>-1 THEN CLS:PRINT SPC(11)dir$:INPUT"New Dir? : ",dir$
60640 RETURN
```

```
1 ' Listing #2
10 REM ** CATDEMO.BAS using CAT.BAS as a module.
20 '
30 ret=1:CHAIN MERGE"cat.bas",60
40 '
50 ' * Menu
60 CLS:LOCATE 1,2:PRINT"Cat  Printcat  Openasc  Fastkeys  Quit"
70 tex$="CPOFQ":r$="":WHILE r$="":r$=UPPER$(INKEY$):WEND
80 k=INSTR(tex$,r$):IF k=0 THEN PRINT CHR$(7);:GOTO 70
90 ON k GOSUB 200, 300, 500, 700, 900:GOTO 60
100 '
200 ' * Cat
210 GOSUB 60000:RETURN
220 '
300 ' * Print CATs & DIRs
310 GOSUB 60000:IF nul THEN 400
320 WIDTH m%*40:LOCATE 1,y1%:PRINT C$:LOCATE 1,m%:PRINT CHR$(18);
330 PRINT TAB(10)"Printer Online ?";CHR$(7):CALL &BB06
340 PRINT#8,STRING$(m%*40,"-")
350 PRINT#8,"Name:":PRINT#8:PRINT#8,"Side:"SPC(4)"Date:":PRINT#8
360 FOR z%=y1%-2 TO y2%+2:FOR c%=1 TO m%*40
370 LOCATE c%,z%:PRINT#8,COPYCHR$(#0);
380 NEXT:NEXT
390 PRINT#8:PRINT#8,STRING$(m%*40,"-")
400 RETURN
410 '
500 ' * Open ASC files
510 GOSUB 60000:MODE 2:IF nul THEN 610
520 PRINT"OPENIN --> ";C$:PRINT
530 PRINT"QUIT = Return or Fire.  SCROLL = Any Key.":PRINT:CALL &BB06
540 OPENIN C$
550 z%=0:WHILE NOT EOF:z%=z%+1
560 LINE INPUT#9,buf$:PRINT z%;SPC(1)buf$:fr!=FRE("")
570 IF INKEY(18)<>-1 OR INKEY(76)<>-1 THEN 610 ELSE CALL &BB06
580 WEND:CLOSEIN
590 PRINT:PRINT TAB(25)STRING$(8,143);" <<  E O F  >> ";STRING$(8,143)
600 PRINT CHR$(7):GOTO 520
610 CLEAR INPUT:RETURN
620 '
700 ' * a few Fast Edit Keys
710 KEY 143,"RUN"+CHR$(13): KEY DEF 50,1,114,82,143 '^R
720 KEY 145,"SAVE"+CHR$(34)+CHR$(13): KEY DEF 60,1,115,83,145 '^S
730 KEY 7,"LIST " 'f7
740 CLS:PRINT"Type the filename to be saved.":PRINT
750 PRINT"Hold down CONTROL key."
760 PRINT"Hit LEFT arrow key, then S key to SAVE.":PRINT
770 PRINT"Use CTRL+R to RUN a loaded program or f7 to LIST it."
780 PRINT"Use CTRL+Copy to run CAT.BAS"
790 CALL &BB06
800 RETURN
810 '
900 ' * Quit
910 CLS:DELETE 60000-
```

```
1 ' Listing #3
10 REM ** CATCALL.BAS, a CHECKLIST of CAT.BAS Variables
20 '
30 CLS:INK 0,1:INK 1,24:INK 2,20:INK 3,6:BORDER 0
40 '
50 ret=1 'ret=1 is a MUST when using GOSUB after a MERGE.
60 m%=1:j%=2:dir$="*.bas" 'Use these variables to OVERRIDE defaults.
70 '
80 'Setting a flag chain merge, fcm=1 helps to avoid repeat merging.
90 IF fcm=0 THEN fcm=1:PRINT"fcm=";fcm:CHAIN MERGE"cat.bas",100
100 GOSUB 60000
110 '
120 MODE 2:PRINT:PRINT SPC(2)"Current CAT.BAS Variables:":PRINT:PRINT
130 PRINT SPC(2)"Selected File C$: ";C$;" _Nothing only if you used QUIT."
140 PRINT:PRINT SPC(2)"ret=";ret;"_MUST be passed as ret= 1 to force MODULE beha
vior."
150 PRINT:PRINT SPC(2)"nul=";nul;"_Returned as nul= 1 for QUIT, nul= 0 for SELEC
T."
160 PRINT:PRINT SPC(2)"Mode m%=";m%;"_Passed or Returned, should be 1 or 2"
170 PRINT:PRINT SPC(2)"Job j%=";j%;"_Passed or Returned, should be 1 for CAT, 2
for DIR."
180 PRINT:PRINT SPC(2)"dir$:  ";dir$;"  _Passed or Returned, Dir$ are STICKY. De
fault is *.*"
190 PRINT:PRINT SPC(2)"Module Colors Returned:"
200 PRINT:PRINT SPC(2)"INK 0,";p%;":INK 1,";k1%;":INK 3,";k3%;":BORDER";p%
210 PRINT:PRINT SPC(2)"Use O & P keys to change Paper color, U & I keys to chang
e Ink 1."
220 PRINT:PRINT:PRINT SPC(2)"Press Return to Exit, Any Key to Loop."
230 CALL &BB06:IF INKEY(18)<>-1 THEN CLS:DELETE 60000- ELSE 10
```

Copychr$ carry the day

from issue 7 of CPCO Online

Written by Frederik Leighton

DOS.BAS - CODE BOSS:

7K, all BASIC including limitations is meant to create an INTERFACE for CPC users and combines a full-fledged MENU & CAT screen. Most everything is controlled with the Arrow & Return keys or JOYSTICK & Fire button. Presentation is pretty modular, so you can slip in your own adaptations fairly easily.
Combining keyboard assignations and commands from the Menu speeds up code writing and editing. Disk management is also helped along by some KEY DEF hocus-pocus involving Return key and Fire button, definitely not in the textbooks.

The downside is that you will totally forget how to use the command line and become completely dependent on this CPCdos. Free copy, get your free copy here now... while it lasts ;-)

A good way to use the joystick is clamped or stuck to the desktop (the four legged variety to be sure). One hand moving the cursor & the other handling the keyboard makes for better speed than using a mouse.

Sorry, only tested on CPC 6128 with one standard 3" drive. If interest we can try to adapt this Copychr$ business to rest of the range, see also site www.sean.co.uk on Downgrading.

LINE-UP:

CPCO #5, ZIPCAT.CPC 1K module returning a filename, Lite.

CPCO #6, CAT.BAS 3K module and standalone, useful placed on all active disks. Good to Nosy Parker or just switch between programs, Fuzzy.

CPCO #7, DOS.BAS 7K standalone, recommended on all "programming" disks but bosses any disk you swap in, Harry Potter.

These programs are really tied together by a Copychr$ engine which is very much the same from one version to the other, even when used on such disparate elements as Menu and Cat. To get more particulars and a list of variables, please check back.

Following is a walkthrough of every menu option and we'll also see where magic and sleight of hand part their ways.

A LONG HOT SUMMER:

Perfect to draw the blinds and square off with yet another Listing. By the way, let us know if there are problems reading or printing any Listings. Quality takes a hit in favour of downloading pages but maybe they can be archived or put on a CD.

LAYOUT:

Everything up to Line -2000 concerns the Menu, the Cat and the Cursor movements necessary to pick a file name and a command from the menu.

Between 2000 - 4000, a subroutine for each menu option, 16 in all.

After 4000- several household routines and KEY DEF set-ups.

7K is quite a byte, even with the phone off the hook. So you might like to do Lines 10-2040 and Lines 4000-5240 first. Then you can slot in the options code as you go along. The execution order in Line 2030 is all mixed up because the options have been moved around in the menu. You can do the same, just double check the line numbers and save as you edit.

PLEASE REMEMBER:

Take all precautions handling your files and always make backups of anything important, which includes work in progress. This program has just been tinkered with till it somehow works and might still FLOP at your end. Also, all "2 drives" code is untested and may be flawed, so double-check please. While typing in DOS.BAS better keep saving as any LOAD, etc. will wipe your code from RAM.

WALKTHROUGH:

After the program has run once, pressing "CONTROL+COPY" together will give you a RUN"DOS. This allows you to cycle between programs and versions as long as DOS.BAS is present on the current disk. Remember to use the Fast Edit keys to save any work in progress BEFORE running DOS.BAS
Sixteen options in all. Three are available with one keystroke (recommended use); "C" key will always give you a new CAT, "R" and "L" keys will RUN or LOAD your selection only while the cursor is still in the CAT screen.

MENU OPTIONS:

These options appear in the same sequence as in the Listing. They are followed by their LINE number and their MENU order number as in line 2030. In that Line the value of variable k, as determined in Line 530, will pick the corresponding option.

Load - 2100 -12
Faster to use "L" key while in CAT screen.
You will get the full set of Edit keys.

Cat - 2200 - 11
Use "C" key ANYTIME to loop into a new CAT.

List #8 - 2300 - 9
Allows for printing only your choice of Line numbers. Of course it's print as print can for everybody these days. I have found it possible to print to an inkjet and a laser printer. The present Listing is obtained using the original printer cable and the command WIDTH 80 as a must.

Run - 2400 - 13
Faster to use "R" key while in CAT screen.
All your keys are set to default (CALL &BB00), except "Control+Copy" combination set to RUN"DOS.

Xdrive - 2500 - 7
If you have two drives please set flag drive fdr=2 in Line30. (If you have more than two drives get a CPC NG :-)

This untested option should shuttle current drive between A and B at every use. Drop a line if we can do better here.

OpenAsc - 2600 - 14
Hope this will work for all ASCII files out there.

New - 2700 - 4
Ready to start typing fresh code. Will load the full set of Edit keys. (Replacing NEW with END is useful while editing DOS.BAS itself).

Erase ! - 2800 - 8
A bit out of the way, on purpose. Remove BEEP (variable b$ in Line 2840) if aggravation. Use just one "LEFT Arrow" key press to get from option "List" to option "Erase !". After validating the Erase option use the "DOWN Arrow" or Joystick DOWN movement = CHR$(10), to confirm the command. In case of mistake use UnErase option RIGHT AWAY. It is possible and fastest to do all the Erase work with just the Joystick. Put your feet up & have a go, just don't bust a gut leaning over to swap disks.

*.BAK - 2900 - 15
Uses Erase option's code.

Rename - 3000 - 3
A hand to Michel Archambault for the filename checking code in Lines 4600. Better type in a valid file name or you'll still be there come morning.

Cat #8 - 3100 - 10
Handy CAT printouts. You can get 2 full 64 file count CATs on one page.

List - 3200 - 1
Good for a Screen listing. You may choose the Line numbers to be shown using normal conventions. "Return" will show all from start. Actually, to just get a listing it's smoother to LOAD with the "L" key and then hit "f7" followed by "Return".

EditKey - 3300 - 6
Will load the full set of Edit keys and give you a screen listing of current key settings. Whenever in doubt use this option as a reminder of what goes where. Some care has been given to the layout but this is a personal matter. The code is in Lines 5000-. If you are already using some of these features then put your own in here. If not, give these a good try & then adapt to suit your preferences. Keys "f2" and "f3" can be given an assignment on the fly. After the Ready prompt hit "f2", type Allan Sugar King & hit "Return". Now hit "f2". Hit it again & again & again. It's really spooky when the knocking on the door starts. After all, you had the blinds closed and the phone off the hook, so how did they find you out?

SaveDos - 3400 - 5
Saves a copy of DOS.BAS to current disk. Allows for optimum use of program. Use Erase option when redundant.

CopyBas - 3500 - 2
If the normal LOAD and SAVE commands work on the selected file then so should this routine. Allows for Renaming and drive switching. Better test 2 drives code. If you don't change disks, you can still copy under a new name. It's important to WAIT for the Ready prompt before removing the current disk. After all, the program you want to copy takes some time to load and ripping out your machine's insides is bad form any day.

UnErase - 3700 - 16
Use this menu option twice and then Rename for a full UnErase. First time should take you to USER 229, Keeper of the Lost Files. Select your file and use UnErase AGAIN. Now you return as USER 0 with a file named #UNERASE.JOY under the cursor. Be sure to Rename as wanted.

All this is only effective if used BEFORE erased file has been overwritten. Better do some practice runs.

COLOURS:

The "UIOP" keys allow you to pick any colours you like. Current PAPER and INK 1 are shown on screen. Use these numbers in line 50 to set your own defaults.

CONTROL:

"Control" shows as an UP Arrow on screen but prints as "^" in Listing. Please press "Control" and indicated key together for execution.
^Z - CLS
^C - normal CAT
^R - will RUN the program currently loaded. Use +++
^COPY - will RUN"DOS.BAS if present on current disk. Use +++
To use the next two : First type a filename, then hold down "Control" key, hit "LEFT Arrow" key and then indicated key.
^S - SAVE"EDITFILE. Use +++
^L - LOAD"DISKFILE. Never used it yet.

SPINDIZZY:

Caution! This "Load, List, Edit, Save, Print & Run" cycle has no brakes.
Start CPC, type run"dos. Arrow to a program, hit "L", hit ""f7", hit "Return".
Edit using key assignments.
Type a short temp filename & a number (R2 D2). While holding down "Control" key, hit "LEFT Arrow" and then "S" key.
Press "Control" + "R" key together. Test your code. Keep editing, saving and testing.
Or press "Control+Copy" keys together to get back to DOS.BAS for a printout or to switch versions etc.

THE HARRY POTTER FACTOR:

Without the List, List #8 and Copy options in the menu the programs utility would take quite a hit. But these "hands off" functions are not normally found in our BASIC environment. They require a program to somehow issue specific commands to be executed AFTER some other program has been loaded. Not done. This mixing of programs. Nothing good will come of it. CPC can't take it, too old, too expensive. Out of memory. Replicant code and what not. Cloning next. Mad as a Red Hatter.

Tosh, a flick of the wand & KAZOOM... 8bit rules. Actually, a bit of a hack is even more fun (see Lines 4000-). First the commands are strung together in the variable com$, Lines 3590 & 4030.
For a LIST: z$ holds the Line numbers and chan$ switches between #0 or #8. A complete com$ might look like this: List 10-1000 ,#8

```
10 REM *** DOS.BAS ENGLISH Version (c)2002 by F. Leighton
20 '
30 DEFINT a-z:v$=CHR$(24):b$=CHR$(7):DIM m$(20):fdr=1 'fdr=2 for 2 drives
40 MODE 2:KEY DEF 76,0,13
50 p=1:k1=16:INK 0,p:INK 1,k1:BORDER p:PAPER 0:PEN 1
60 '
100 REM *** Menu
110 DATA 2,72,1,3,10,2,9
120 DATA 16,List,CopyBas,Rename,New,SaveDos,EditKey,Xdrive,Erase !
130 DATA List #8,Cat #8,Cat,Load,Run,OpenAsc,*.BAK,UnErase
140 RESTORE 100:GOSUB 200:RESTORE 100:GOSUB 500:GOSUB 2000:GOTO 140
150 '
200 REM ** Show menu
210 GOSUB 600:CLS:fb=0
220 READ m:FOR k=1 TO m:READ m$(k):m$(k)=" "+m$(k)+" ":NEXT
230 k=0:FOR z=y1 TO y2 STEP yy:FOR c=x1 TO x2 STEP xx
240 k=k+1:IF k<=m THEN LOCATE c,z:PRINT m$(k);
250 NEXT:NEXT:LOCATE 1,y2+1:PRINT STRING$(80,154)
260 '
300 REM *** Cat
310 DATA 1,61,7,25,20,1,12
320 RESTORE 310:GOSUB 600:fc=1:LOCATE 1,y1-3:CAT
330 FOR z=y1+1 TO y2:LOCATE x1+8,z:IF COPYCHR$(#0)="."THEN NEXT ELSE y2=z-1
340 FOR c=x1+xx TO x2 STEP xx:LOCATE c+8,y1:IF COPYCHR$(#0)="."THEN NEXT ELSE x2
=c-xx
350 t=0:FOR z=y1 TO y2:FOR c=9 TO x2+8 STEP xx
360 LOCATE c,z:IF COPYCHR$(#0)="."THEN t=t+1
370 NEXT:NEXT:LOCATE 21,5:PRINT"Files:";t
380 GOSUB 1000:fc=0:caty1=y1:caty2=y2:x$=C$:IF fe THEN nul=1 ELSE nul=0
390 LOCATE 30,caty2+3:PRINT v$;" *** ";x$;" *** ";v$
400 RETURN
410 '
500 REM ** Choose menu option
510 GOSUB 600
520 GOSUB 1000:IF fe THEN 520
530 FOR k=1 TO m:IF m$(k)<>C$ THEN NEXT
540 RETURN
550 '
600 REM ** Read cursor data
610 READ x1,x2,y1,y2,xx,yy,sp:x=x1:y=y1:RETURN
620 '
1000 REM *** Joycursor ***
1010 GOSUB 1200
1020 IF INKEY(2)<>-1 OR INKEY(73)<>-1 THEN y=y+yy:GOTO 1010
1030 IF INKEY(0)<>-1 OR INKEY(72)<>-1 THEN y=y-yy:GOTO 1010
1040 IF INKEY(1)<>-1 OR INKEY(75)<>-1 THEN x=x+xx:GOTO 1010
1050 IF INKEY(8)<>-1 OR INKEY(74)<>-1 THEN x=x-xx:GOTO 1010
1060 IF INKEY(18)<>-1 OR INKEY(76)<>-1 THEN 1330
1070 IF INKEY(50)<>-1 AND fe=0 AND fc=1 THEN x$=C$:GOTO 2400 'R un
1080 IF INKEY(36)<>-1 AND fe=0 AND fc=1 THEN x$=C$:GOTO 2100 'L oad
1090 IF INKEY(62)<>-1 THEN 140 'C at
1100 IF INKEY(42)<>-1 AND k1>0 THEN k1=k1-1:INK 1,k1 'U
1110 IF INKEY(35)<>-1 AND k1<26 THEN k1=k1+1:INK 1,k1 'I
1120 IF INKEY(34)<>-1 AND p>0 THEN p=p-1:INK 0,p:BORDER p 'O
1130 IF INKEY(27)<>-1 AND p<26 THEN p=p+1:INK 0,p:BORDER p 'P
1140 LOCATE 41,5:PRINT "Paper:";p:LOCATE 61,5:PRINT "Ink:";k1;
1150 CALL &BB06:GOTO 1020
1160 '
1170 REM ** Verification
1200 IF y<y1 THEN y=y2 ELSE IF y>y2 THEN y=y1
1210 IF x<x1 THEN x=x2 ELSE IF x>x2 THEN x=x1
1220 IF y=yb THEN IF x=xb AND fb THEN 1330
1230 '
1240 REM ** Concatenation
1250 C$=SPACE$(sp):z=0:FOR c=x TO x+sp-1:z=z+1
```

```
1260 LOCATE c,y:MID$(C$,z,1)=COPYCHR$(#0):NEXT
1270 fe=0:IF INSTR(SPACE$(sp),C$)THEN fe=1:C$=" ":GOTO 1310
1280 z=sp:WHILE z>1 AND MID$(C$,z,1)=" ":z=z-1:WEND:C$=LEFT$(C$,z+1)
1290 '
1300 REM ** Display
1310 IF fb THEN LOCATE xb,yb:PRINT bk$; ELSE fb=1
1320 LOCATE x,y:PRINT CHR$(24);C$;CHR$(24);:xb=x:yb=y:bk$=C$
1330 fr!=FRE(""):RETURN
1340 '
2000 REM *** Execution DOS ***
2010 KEY DEF 76,0,13:CLEAR INPUT:bak=0:ref=0
2020 f$=" Fire or Return ":ref$="Quit ="+f$:imp$=" Printer Online ? "
2030 ON k GOSUB 3200,3500,3000,2700,3400,3300,2500,2800,2300,3100,2200,2100,2400
,2600,2900,3700:RETURN
2040 '
2100 ' * Load
2110 IF nul THEN 2130
2120 CLS:PRINT"LOAD --> ";x$:GOSUB 5000:LOAD x$
2130 RETURN
2140 '
2200 ' * Cat
2210 RETURN
2220 '
2300 ' * List #8
2310 IF nul THEN 2340
2320 CLS:PRINT x$:PRINT:PRINT v$;imp$;v$;b$:chan$=",#8"
2330 GOSUB 4000:IF ref=0 THEN WIDTH 80:LOAD x$
2340 RETURN
2350 '
2400 ' * Run
2410 IF nul THEN 2430
2420 CLS:PRINT"RUN --> ";x$:CALL &BB00:GOSUB 5230:RUN x$
2430 RETURN
2440 '
2500 ' * Xdrive
2510 IF fdr=2 THEN LOCATE 7,caty1-2:IF COPYCHR$(#0)="A"THEN |B ELSE |A
2520 RETURN
2530 '
2600 ' * OpenAsc
2610 CLS:IF nul THEN 2670
2620 PRINT"OPENIN --> ";x$:PRINT:PRINT"Scroll = Any Key / "+ref$:PRINT
2630 CALL &BB06:OPENIN x$
2640 z=0:WHILE NOT EOF:z=z+1:LINE INPUT#9,buf$:PRINT z;SPC(1);buf$:fr!=FRE("")
2650 IF INKEY(18)<>-1 OR INKEY(76)<>-1 THEN 2670 ELSE CALL &BB06
2660 WEND:CLOSEIN:PRINT:PRINT TAB(25)STRING$(8,143);" <<  E  O F  >> ";STRING$(8
,143);b$:PRINT:GOTO 2620
2670 CLEAR INPUT:RETURN
2680 '
2700 ' * New
2710 CLS:GOSUB 5000:PRINT v$;" N E W ";v$:PRINT:NEW 'END
2720 '
2800 ' * Erase
2810 IF nul THEN 2860
2820 LOCATE 20,caty2+3:PRINT"ERASE :"
2830 IF bak THEN LOCATE 30,caty2+3:PRINT v$;" *.BAK ? ";v$;CHR$(18):x$="*.BAK"
2840 LOCATE 55,caty2+3:PRINT"( OK = ";CHR$(241);" )";b$:GOSUB 4500
2850 IF r$=CHR$(241)OR r$=CHR$(10)THEN |ERA,x$ ELSE PRINT b$
2860 RETURN
2870 '
2900 ' *.BAK
2910 bak=1:GOTO 2810
2920 '
3000 ' * Rename
3010 IF nul THEN 3050
3020 LOCATE 20,caty2+3:PRINT"RENAME :":LOCATE 1,4:PRINT CHR$(19)
```

```
3030 LOCATE 25,1:PRINT ref$:LOCATE 25,3:INPUT"New Name : ",n$
3040 GOSUB 4600:IF ref=0 THEN |REN,n$,x$
3050 RETURN
3060 '
3100 ' * Cat #8
3110 LOCATE 1,4:PRINT CHR$(19):LOCATE 15,1:PRINT v$;imp$;v$;b$;SPC(10);ref$
3120 GOSUB 4500:IF ref THEN 3180
3130 WIDTH 80:PRINT #8,STRING$(80,"-")
3140 PRINT #8,"Name:":PRINT #8:PRINT #8,"Side:"SPC(4)"Date:":PRINT #8
3150 y=caty1-2:FOR x=1 TO 40:LOCATE x,y:PRINT#8,COPYCHR$(#0);:NEXT
3160 FOR y=caty1-1 TO caty2+2:FOR x=1 TO 80:LOCATE x,y:PRINT#8,COPYCHR$(#0);
3170 NEXT:PRINT#8:NEXT:PRINT#8:PRINT#8,STRING$(80,"-")
3180 RETURN
3190 '
3200 ' * List
3210 IF nul THEN 3240
3220 CLS:PRINT x$:PRINT:PRINT v$;" Screen Listing ";v$:chan$=",#0"
3230 GOSUB 4000:IF ref=0 THEN LOAD x$
3240 RETURN
3250 '
3300 ' * EditKey
3310 CLS:GOSUB 5000:LIST 5020-5230
3320 RETURN
3330 '
3400 ' * SaveDos
3410 CLS:PRINT "SAVE --> DOS.BAS":SAVE"DOS.BAS
3420 RETURN
3430 '
3500 ' * CopyBas
3510 LOCATE 7,caty1-2:d$=COPYCHR$(#0):d$=d$+":":CLS:PRINT v$;" * Copy --> "v$;"
";x$:PRINT:PRINT ref$:PRINT
3520 PRINT"Other Name or Other Drive ?  (Y/N)":GOSUB 4500:IF ref THEN 3600
3530 IF r$<>"Y"THEN n$=x$:GOTO 3580
3540 PRINT:INPUT"What other name : ",n$:GOSUB 4600:IF ref THEN 3600
3550 IF fdr=1 THEN 3580
3560 PRINT:PRINT"Other Drive ?  (Y/N)":GOSUB 4500:IF ref THEN 3600
3570 IF r$="Y"THEN IF d$="A:"THEN d$="B:"ELSE d$="A:"
3580 PRINT:PRINT v$;" * WAIT for the Ready ";v$;b$;" then Change Disk ";"and"
3590 com$="Save"+CHR$(34)+d$+n$:CALL &BB00:GOSUB 4040:PRINT:LOAD x$
3600 RETURN
3610 '
3700 ' * UnErase
3710 IF PEEK(&A701)<>229 THEN POKE &A701,229:GOTO 3730
3720 |REN,"0:#UNERASE.JOY",x$:POKE &A701,0
3730 RETURN
3740 '
4000 REM ** Key def list/save
4010 PRINT:PRINT"ALL =";f$;" /  Quit = Q":PRINT:CALL &BB00:KEY DEF 76,0,13
4020 INPUT"List Lines -?-?- : ",z$:z$=UPPER$(z$):IF z$="Q"THEN ref=1:GOTO 4070
4030 com$="List "+z$+chan$
4040 KEY 159,"key def 18,0,13:key def 76,1,88:"+com$+CHR$(13)
4050 KEY DEF 18,0,159,13,13:KEY DEF 76,0,159,88,88:GOSUB 5100
4060 PRINT:PRINT"Press ";v$;f$;v$;:PRINT:CLEAR INPUT
4070 RETURN
4080 '
4500 REM ** Responses
4510 r$="":WHILE r$="":r$=UPPER$(INKEY$):WEND
4520 IF r$=CHR$(13)THEN ref=1 'cancel
4530 RETURN
4540 '
4600 REM ** Check file name
4610 k=LEN(n$):z=INSTR(n$,"."):IF INSTR(n$," ")OR z>9 OR(z=0 AND k>8)OR(z>0 AND
k-z>3)OR k>13 OR k=0 OR z=1 THEN ref=1:PRINT b$;
4620 RETURN
4630 '
```

For a SAVE: d$ holds the drive letter and n$ the name to be used. A com$ here might look like:

Save"B:MYPROG.BAS.

Lines 4040 & 4050 : Adding a CHR$(13) forces execution of these commands. This is the payload we want to deliver by way of the "Return" key and the Fire Button. Adding some KEY DEFs setting these same keys back to their defaults, at the very moment our commands execute, will cover our tracks. See, no hands...
Of course, from time to time the whole string stays "stuck". You can then use the "Enter" key in the meantime and type CALL &BB00 from the prompt or run the EditKey option in DOS.BAS.

ON THE WHOLE:

A lot of useful bits and pieces tied together by fast repeat file picking should make for solid support when editing programs or managing disks. So, when it bombs please help out with a bug report. Any suggestions are welcome.

Remember to put the phone back on the hook and to get that door fixed :-))

```
5000 REM ** Key Defs
5010 CALL &BB00
5020 KEY 0,"locate "
5030 KEY 1,"print "
5040 KEY 2,"KEY 2,"+CHR$(34)
5050 KEY 3,"KEY 3,"+CHR$(34)
5060 KEY 4,"chr$("
5070 KEY 5,"AUTO "
5080 KEY 141,"LOAD"+CHR$(34)+CHR$(13):KEY DEF 36,1,108,76,141 '^L
5090 KEY 146,"CLS"+CHR$(13):KEY DEF 71,1,122,90,146 '^Z
5100 KEY 6,"RENUM "
5110 KEY 7,"LIST "
5120 KEY 8,"EDIT "
5130 KEY 9,"DELETE "
5140 KEY DEF 17,0,40 '  [ = (
5150 KEY DEF 19,0,41 '  ] = )
5160 KEY DEF 22,0,36 '  \ = $
5170 KEY DEF 26,0,34 '  @ = "
5180 KEY DEF 65,0,50,64'" = @
5190 KEY DEF 68,0,61 'tab = =
5200 KEY 144,"CAT"+CHR$(13):KEY DEF 62,1,99,67,144 '^C
5210 KEY 143,"RUN"+CHR$(13):KEY DEF 50,1,114,82,143 '^R
5220 KEY 145,"SAVE"+CHR$(34)+CHR$(13):KEY DEF 60,1,115,83,145 '^S
5230 KEY 158,"RUN"+CHR$(34)+"DOS"+CHR$(13):KEY DEF 9,1,224,224,158'^+copy
5240 RETURN
```

Copychr$
The Aftermath

from issue 9 of CPCO Online

Written by Frederik Leighton

You have unwisely let these pesky critters loose on your screen, running all over, getting under things and painfully stretching your drive belt.

This is what they look like. See how they're pestering Nabokov's friend again?

You should have just said NO, but you could not resist some virgin code, and now it is too late to stop...

No matter what your CPC or whether using latest versions of Arnold, Caprice & WinAPE, you - like so many others - have fallen victim to a nasty piece of code, passing itself off as some kind of OS front end, while ruthlessly destroying your hardware in the background! Emulators are melting heat sinks left and right. Only the fortunate users of the CPC 464 have been able to escape this onslaught as Brian Watson runs out of drive belts... The last ones are going for 50+ quid, 20 euro stamps, no checks.

I know, I know. But they are making me do it to fill out this summer issue. Now for the really important stuff.

THE FAQ:

Q: Why does it look so cool?

A: To attract women. John runs field tests on his Plus. Working late again? John?

Q: Wow, far out. So what is it?

A: Supposed to effortlessly snag a file name in the normal CAT & |DIR screens on any make CPC except 464. No preparation, instantly, any file name, any 3" disc, drive A or B. Also good in latest emulators tested. Any simple code will then work with the file name. Hands-off mucking about...

Q: Sounds good. Hard to set up, lots of work to keep going?

Oh: No. Easy as Pong.

Q: Been done before then... bit of a yawn?

Uh: Hope not. Anyway, the little ones like to play with it until they vomit. It's not as bad now with emulators. These wireless keyboards are great too. Under the shower, a good shake and upside down to dry out. The keyboard silly... not the kids. Oh well.

Q: Will the CPC NG be backward compatible with this new software?

Ah: Indeed, the CPC NG has been 100% reverse engineered in order to handle it. (We hear the waterproofing is much harder then expected but it is to be put together again by shipping day).

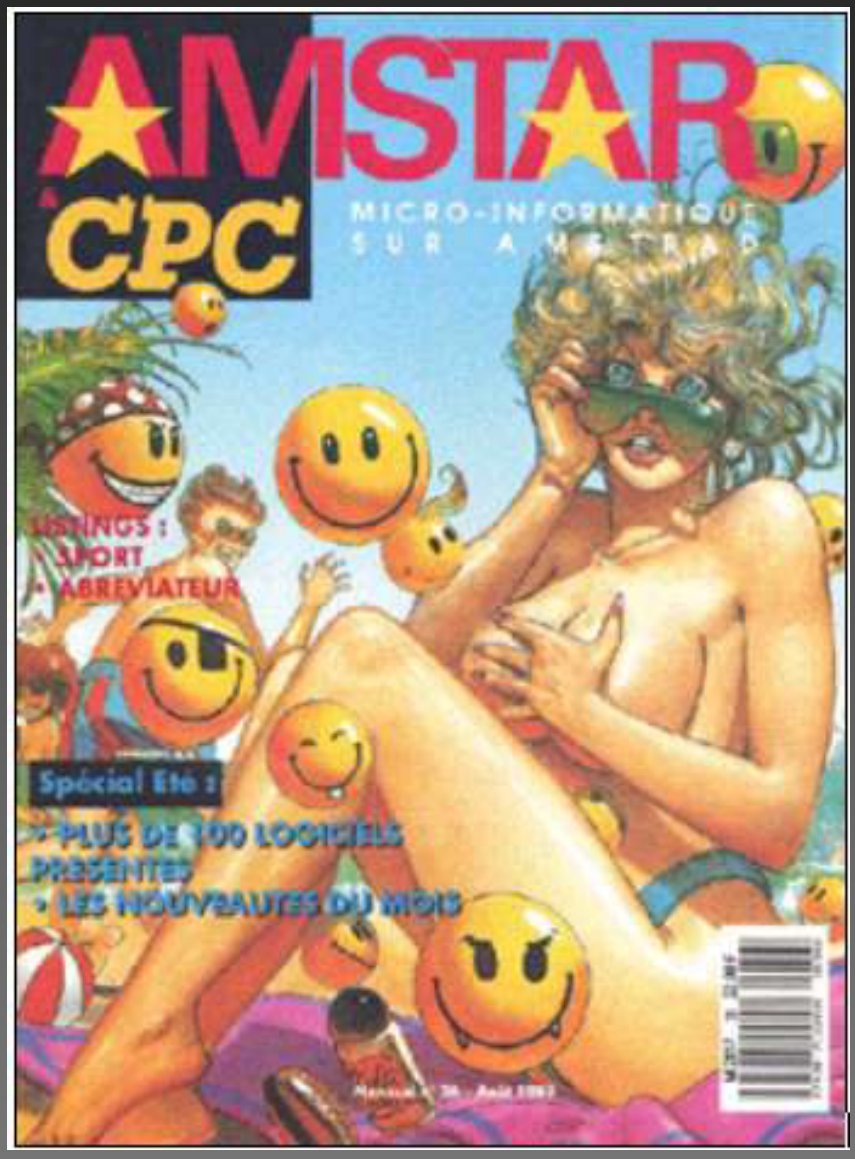
TO BUSSINESS:

In the CPCO Downloads section you will find an all new Copychr$.zip file. It contains a .DSK with all published programs ready to go together with Help files describing the keystrokes to use. The up to date Listings for each program are also available there and should print out exactly as on screen in MODE 2.
So everything you need can be downloaded while the relevant comments are right here on CPCO, issues 5-7 and 9. Those you can of course Print or Save.

All three programs have been updated to some extent. Most changes made are adaptations for a 2-drive system and in that case ONLY these versions (v1.2) will do. Whenever practical replace all earlier versions you may have. Simply use the built in Save options to overwrite them.

Good news: if you did the type ins already and have only one drive, there is not much to change. Anyway, further down you will find the relevant line numbers.

Please let us know if you find any errors, as these files have been retyped. The same goes for difficulties in use, confusion as to what you are supposed to do or plain bugs. Consider yourselves expendable while testing this new software, but make sure to report back in :-) at cpcoxygen@hotmail.com



PRESENTATION OF MODIFICATIONS:

These are plain Basic programs and are limited to handling two 3" drives, 64 files each max. Within those limitations they should do well under Basic 1.1 on real CPCs and in emulation. In case of error message "disc missing" or "file not found" best use Cancel.

Each has a flag drive. Default is fdr=2 for a 2-drive system. Set fdr=1 for 1 drive.

Simply TOGGLE drives with the SPACEBAR or Joystick FIRE 2.

CAT.BAS and DOS.BAS will both restart from drive A or B when you press Control + A or B. This means you can instantly work with all other files on both drives, just by recalling a single copy of Cat.Bas or Dos.Bas between operations. The drive you are working on always comes up and the programs perform the same on both drives.

If Basic will load it, then the CopyBas option allows you to copy one file at a time. For the real thing you need |CPM or a utility. Please pay attention when copying files. There are two phases - Loading first, then Saving. You MUST wait for the "Ready", be it on a real machine or in emulation. Otherwise the file you want to copy may not have finished loading and you will be saving only the odd bit of this and that or getting funny noises! The larger the file, the more time it will need. Best is to always recheck your choices, by that time things should be Ready for saving.
If you have trouble in the middle of a copy operation you can back out with Control+A or B. Use the ENTER key if the Return key still has unwanted active commands.

A word of caution on manipulating fresh data in emulators. As long as your new data on .DSK has not been backed up AND verified, it's possible to lose complete sessions. Archiving successive edits is also more secure than overwriting. Having backed-up your work from A: to B: just gives you two crushable images on your HD. Both can turn into bad news in a flash. Moving a lot between emulators, while using the same set of .DSK images is very risky, so is restarting an already active emulator (sharing violations). All typical when editing and testing new programs, or extensively modifying data. Run of the mill for Dos.Bas ;-)

Following are most of the modifications made since the listings were first published.

ZIPCAT.CPC v1.2 in ZIP.BAS - CPCO #5

See ZIPHELP.BAS for current instructions.

Correction in Line 920 Added a ";" to read ...PRINT CHR$(7);:GOTO 910

If you have only one drive make none of these changes at all.

Modified lines or numbers:

100, 320
64000 added v1.2
64010 added fdr=2
64170 new, toggle drives with Spacebar
64180 number changed

CAT.BAS v1.2 - CPCO #6

Metaphysical quandary here. Wrote it to remain at 3K and to fit on one page as a type in.

So in order to get the drive switching code, the colour changing stuff has been whacked. You can decide to keep it or write it back from the listing of course. If you're typing in and only use one drive anyway don't change a thing.

See CATHELP.BAS with current instructions.

Modified lines or numbers:

60000 added v1.2
60010 added fdr=2 and rr$ holding a file name to be re-run. You can pass rr$ when using Cat.Bas as a module.
60050 now use Control + A or B Keys to recall program rr$ (Cat.Bas by default) from drive A or B.
60260 added show version.
All "colour" lines removed.
After 60390 lines renumbered
60400 toggle drives with Spacebar and Fire 2.

DOS.BAS v1.2e - CPCO #7

DOSHELP.BAS has up to date instructions on keys to use and issue #7 a walkthrough of all available options.

10 added v1.2e
10 -100 moved some elements around, added flag kill edit, fke. If you set fke=1 then the EditKey option is bypassed and your keyboard will be set to its defaults (except Control+A and B).
120 -130 moved options around again.
370 added show version.

Reserving lines in Inkey Loop:

1000 Reserving lines 1100-1200 for more INKEY()s. This with a RENUM in the cursor section. Don't bother with this for the type in, manage your own line numbers when and if.

But be sure to catch:

1100 new and 2510 modified for the drive switching.
1110 Help... sets |USER,0 and fke=1 to insure a default keyboard for session only.

2030 line number sequence changed.

3500 - 3600 modifications to get a smooth copy dialog. Always double check you're writing to the right drive & did get a good copy and did not just WRITE BACK OVER the original on the same disc. No UnErase for the wicked!

4060 modified to print com$ & show what will happen BEFORE you hit "Return" or "Fire" to List or Copy. This is THE moment to double check for errors. You will clearly see what file is going to what drive and can simply Back Out with Control+A or B.

5000 - removed some assignments to free up buffer space.

If you change assignments or load and run other programs with redefined keys, and get an "improper argument" error with code that ran OK before, a full buffer is the culprit. CALL &BB00 cleans house completely. The lines with remapped characters in 5140-5190 have been REMmed out. Please see what you would like to do there yourself.

CONSUMER'S NOTICE:

Satisfaction guaranteed! Otherwise, send the product back to CPCO on an Amsoft CF-2 3" disc, where the staff will see it is properly recycled.

DOWNLOAD

Go to **www.cpcoxygen.net** to download a complete *.DSK version of the programs.

PRINTING FROM ARNOLD

Arnold allows for printing to a file and all LIST,#8 and PRINT,#8 commands work normally. The listings in the Zip file are output from Dos.Bas running in Arnold.

It is well worth setting up for (once) and should make all printing from Arnold routine. Please contact CPCO if you have more information on printing from emulators in general and maybe we can put together a How To.

Situation:

Arnold prints to the file PRINTER.TXT. It will append successive session print jobs to that file. The Output will show up AFTER shutting Arnold down. When you start Arnold again printer.txt will be blank and the print output gone. Printing the *.TXT files from Notepad is problematic because of mangled line breaks.

WYSIWYG or summat:

Here's how to work it for Dos.Bas in Arnold and at the same time obtain perfect listings for ALL your programs, exactly as on screen in MODE 2. The most important thing is to use the WIDTH 80 command.

For this result to show up on paper set up as follows:

" In Arnold select Settings.Amstrad settings.Printer Output.File from the menu.
" Find the file PRINTER.TXT in the Arnold directories. (Try Arnold\roms\cpcplus)
" Make a shortcut to this file. Make also a shortcut to its Folder; call that one Printer CPC. Make a shortcut to Word.
" Put all three shortcuts together, handy where you can see them after you shut down Arnold.
" In Notepad set the Font option to Courier New, Regular, size 10.

Now working these three links will make printing a cinch.

Run Arnold, run Dos.Bas and use one or more of the print options.
CLOSE Arnold. Then do one or both of the following:

" Simply drag the printer.txt link onto the Word link. Word should open with perfect 80 chars wide lines ready to print and save, all you have to do is insert page breaks as wanted.

" Double-click on the printer.txt shortcut to open the printer.txt file and use Save As... to name a "your_output.txt" right there in the printer folder. Later you can use Word to print those *.txt files anytime.

CLOSE Word to release printer.txt and repeat printing from Arnold as needed.

Use the shortcut Printer CPC to view or copy and move the output anywhere you wish, but be careful with any Arnold files present in that folder!

Please take into account that your code is now parsed into lines of text 80 chars or less wide and useful only for printing out nicely... or was that nicely useful?

THE ZILOG Z80

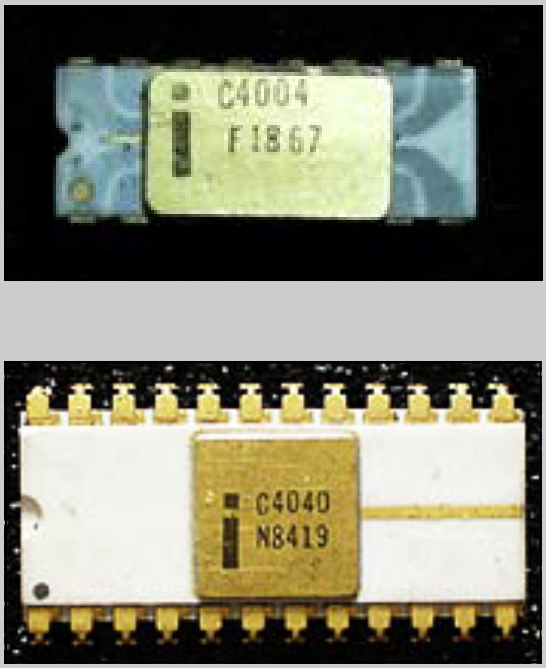
Written by John Kavanagh

John Kavanagh casts an eye back to a time when men were men and when processor clock speeds could be counted on one hand (in megaheartz!).

First comes History

To understand the origin of the Z-80 (the famous little chip that powers the CPC) you first got to know the history of Zilog, and indeed the history of the processor itself. But since this article is more about the Z-80 then processors in general, we won't dive into too much detail here. Basically the short version of the story is as follows:

In 1969 Intel (you probably heard of them!) was approached by a Japanese company to produce chips for an electronic desktop calculator. Intel had an idea of reducing costs by building the calculator around a single-chip, and so you have it, the microprocessor is born, the 4004. By today standards it pretty simple, with only 4 bit data and address buses, it input/output was handled by just 16 pins.
The years went on and Intel improved the design with the 4040 before moving onto 8-bit processors and in 1974 produced the Intel 8080, which was later used in the first home computer, the Altair 8800 in 1975. That's the year Federico Faggin who worked at Intel on the 4004 and later processors, left the company to form Zilog with Masatoshi Shima. On setting up Zilog they designed a new processor that was compatible with all the 78 instruction set of the 8080 plus a 120 more instructions on top of that, plus more registers etc! It was of course, the mighty Zilog Z-80.



Intel's 4004 (TOP) and the imporved 4040 (BOTTOM), both were 4-bit processors which were introduced in the 70's

The beast itself, the mighty Zilog Z80

The Z80 Specs

The Z80 microprocessor came out in July 1976 with a clock speed of 2.5 MHz, later versions had higher clock rates with the Z80A at 4 MHz (the version used in the CPC/plus), Z80B at 6MHz and the Z80H at 8 MHz. There were also further versions based on the original Z-80 (see box out).

Altair 8800, the first home computer.

As I already mentioned, the Z80 is an 8-bit processor, but has 16-bit addressing which means that it could handle up to 64 KB of memory, or a better term would be, to 'see' 64KB at any one time. Computers that had more then 64K gets around the limitation by using a system known as bank switching, which basically means swapping different parts of memory to be seen by the processor. The Z-80 also featured automatic DRAM refresh circuitry. Let me explain, computer RAM (Random Access Memory) needs to be refreshed every few milliseconds otherwise data would be lost. With some processors, computer manufactures had to use extra circuitry to refresh the RAM, but with DRAM refresh built into the CPU there's no need for all this extra circuitry which make computers easier to produce and cheaper too.

As any machine code programmer would know, registers are important and the Z-80 was not lacking. It had 8 main general registers plus very interesting, alternate registers. It also had two index registers (IX and IY) and relocatable vectored interrupts via the 8 bit IV register. For a full list of registers see box out - "Z-80 Processor Registers".

The CPU uses a separate 64k of addresses for input and output devices. The I/O (Input/Output) uses instructions called IN and OUT (surprise) to send or receive date from the selected I/O devices. The address for I/O uses the same wiring as the addresses for memory so the CPU has to use special control singles to recognise memory addresses from I/O addresses.

The CPU uses a separate 64k of addresses for input and output devices. The I/O (Input/Output) uses instructions called IN and OUT (surprise) to send or receive date from the selected I/O devices. The address for I/O uses the same wiring as the addresses for memory so the CPU has to use special control singles to recognise memory addresses from I/O addresses.

Looking back, the Z-80 was a very simple piece of technology by today standards, it lacked stuff like a MMU (Memory Management Unit) and pipelining (to performs tasks in several steps) but in the day it was untouchable in terms of power to price ratios. Having said this, the Z-80 do have a lot of quirkiness as a lot of processors had in the early days of the microcomputer. For example, while I said there were an extra 120 instructions over Intel's 8080, not all instructions worked on every Z-80 manufactured. Basically a lot of the CPU's off the production line in the 70's and early 80's had a good chance of having the edge of the chip die to be infected by foreign bodies which cause damage to the chip. To fix this in the Z-80, Zilog put the least important operating codes at the edge of the chip but since Zilog could not be sure that those instructions at the edge of the chip die will work, those instructions became undocumented hence the name undocumented op code (operation code).

There is a lot to know about this 'simple piece of technology' and we only scratch the surface in the article. Future articles will cover Z80 topics in more detail, and of course, the new eZ80 will get a mention too.

Z-80 Processor Registers

Program Control		Flag Bits	
PC		S Z - H - P N C	
SP			
General Registers		**Alternate Registers**	
A	F	A'	F'
B	C	B'	C'
D	E	D'	E'
H	L	H'	L'
Index Registers		**Hardware Control**	
IX		I	R
IY		IFF 1	IFF 2

Zilog Wonders

Z-180

There is not much information on this processor apart from the fact that it based on the Z80 and I'm pretty sure it had a MMU (Memory Management System).

Z-280

Introduced in July 1987, a 16-bit version of the Z80, 256 byte cache,

features of multitasking, loads of new op codes, in fact, a total of over 2000! Sounds like programmer headache to me, if he/she was using machine code that is. Another interesting feature was that the processor runs at either two or four times faster then the system bus, just like today's modern Intel/AMD processors.

Z-8000

Another 16 but processor which predates the Z280, introduced shortly after Intel's 8086, and what a lot of people would consider more superior. Unfortunately it came onto the market with a few bugs that hindered its acceptance. It probably would have been the processor of choice for the PC otherwise.

What's interesting, is that the sixteen registers were exceedingly flexible with options of either sixteen 8 bit registers, sixteen 16-bit registers, eight 32-bit registers or four 64-bit registers.

The Z8000 also featured two modes, a operating system mode and a user mode which prevented users from messing around with interrupts and other stuff that could crash a processor. It also had automatic DRAM refresh like the Z80.

Z-80000

Same as the Z8000 but it had a 32-bit core instead of a 16-bit one. It also featured a 6-stage pipeline which just says that it can do a lot of stuff per clock cycle.

eZ80

Zilog's new baby, to find out about this great wonder, refer back to Issue 1 of CPC Oxygen.

The First Issues!

John Kavanagh writes from two viewpoints as he explains what got him into making CPC Oxygen (http://cpcoxygen.digi-alt.net) what it is today.

CPC Oxygen Online Magazine coming soon! check out the link at **http://www.pcpages.com/cpco** and remember to keep coming back for updates as soon as it takes off.

Oh and please give comments :-)

comp.sys.amstrad.8bit—Date: 2001-11-12 16:23:00 PST

From the 'CPC Oxygen' viewpoint

Wanting to be involved in the CPC scene was always a interest of mine since near the beginning of using a CPC. Being introduced to the micro computer world has surely changed many people lives, people who wouldn't have being interested in programming, gaming or the computering lifestyle became interested in one way or another. For me it was the programming, reading in computer magazines about current and upcoming games, programmed often by one man teams who became famous almost over night. Then there was the programming articles, be it the simplistic BASIC language of many 8-bit micro-computers of the time or complex machine code routines, it was all fun. There was nothing like typing in a program, editing it to find out how to works and actually running it.

For me it was the magazines of the time that made this happened, only for that I would have been stuck with a BASIC computer manual and stuff that I figure out by myself. There was no internet (not as we know it today) and books were expensive. I got very involved in computer magazines, even to the point of wanting to be involved, to bring the same amount of enjoyment to the readers that I got, but of course, at my young age, it was just a dream.

Years past and PCs came and go and the internet has exploded with popularity. One of the first things I searched for was CPC sites and to my delight I found many. Straight away I wanted to start my own site but with the lack of time I had, it didn't happen, well not until early 2002 when the first issue of CPC Oxygen was launched. I'm not sure what made me go for it but I think it was the amount of free time I had on my hands and the large amount of interest for the Amstrad CPC that I'd seen from postings in the **comp.sys.amstrad.8bit** newsgroup. Both of wish prompted the newsgroup posting above.

The first issue (January 2002) was received well by the CPC community with the only complaint been that the issue was very short. Which of course was true but nevertheless the issue got people interested in supporting the magazine.

The second issue set the layout for the year with its mixture of entertaining as well as technical content. Highlights in future issues were the CPC-NG articles written by CPC-NG's leader Christophe Guelf and technical articles of Z80 Programming and the Copychr$ series by Xavier Glattard and Frederik Leighton. Since CPC Oxygen was to cater for all CPC fans, there were also many game reviews and a constant news section.

Issues were appearing monthly and while not of any great size, the articles proved to be of great interest to the CPC community, which is what CPCO set out to do in the first place. While new issues no longer appears monthly due to lack of time on my part, the magazine still goes from strength to strength. Not only are we gathering content for upcoming issues, we're also working on a new improved website for the magazine. There's still a lot of stuff still to be written about the CPC so keep your eyes on **cpcoxygen.digi-alt.net**, you won't be disappointed.

A EARLY EXAMPLE OF THE CPC OXYGEN SITE

From the 'AA Online' viewpoint

Wanting to have the complete set of 117 issues of Amstrad Action displayed online was a desire of mine before the idea of CPC Oxygen came to mind.

Although I always wanted to do a site / fanzine about the CPC, I never had a complete picture in my head of what it should be like and what topics to cover, but with Amstrad Action it was different. The posting shown below came a few weeks after I got re-interested in the **comp.sys.amstrad.8bit** newsgroup. As I read postings over the coming weeks I came across a set about the desires of wanting Amstrad Action scans online which got my imagination to rush. What if I could host all 117 issues?

Great, I thought, I had enough website skills to make it happen, so did many other people on the newsgroup but I had the time to make it happen, or so I thought! The replies from the posting below were 'excited but cautious' which was understandable since Future Publishing wasn't interested before. My own emails to Future went unanswered as well, fair enough probably.

However the need to be involved with the CPC scene drove me to think up CPC Oxygen. Since I wasn't a good writer (still not as you may have noticed) I was more interested in getting people to write articles with myself being the person that links it all together, to be like a central hub for the magazine. So by January 2002 the first issue of CPCO went online (see previous page).

Moving on to February 2003 I email Future again, for some unknown reason to myself I was hopeful, perhaps believing if I stated not just my desire but the desire of the many CPC fans then maybe we would get the permission needed to display AA online. I also stated that the scans would not be used for profit.

Prompt reply came, excitement soon became disappointment as my request was rejected. Followed by another email, curiously I opened it. To my joy was what I wanted to hear all along, permission to have Amstrad Action content online.

Well actually Future Publishing offered me a licence to display Amstrad Action content with copyright still belonging to Future. After a phone call explaining to me that I would have to remove any content such as adverts and freelance articles if I get any objections from the copyright holders. I was also informed that they will not be able to help me with the project by supplying content such as past issues of Amstrad Action. Well I wasn't expecting any content, that would have been too good to be true, for sure ;-)

Within a short time, Future emailed me a contract which I had to print off two copies (the only time when my printer gave me trouble!), signed both copies with a witness signatures also. Posted both copies off to Future and got one copy returned with me with another signature on it. As easy as that.

It's basically a yearly licence that's updated automatically each year along with many other paragraphs that lasted for a few pages.

Many CPC fans helped with the scanning over the months that followed. My thanks goes out to them, without their help it would be near impossible to complete such a project. As I write this we're growing even closer to the half way point and already had a number of new people wishing to help with issue scanning.

Currently we're working on a new enhanced version of the CPC Oxygen website and will be looking at ways to increase the enjoyment of reading Amstrad Action online. Don't forget to check it out at **cpcoxygen.digi-alt.net**. I'm sure you agree, it's still a joy to read and a great historic source into the world that was the Amstrad CPC.

Hi,

I heard a lot about people wanting to scan Amstrad Action magazines and let them be accessed by anyone on the internet. This is something I would really be interested in, but surely it would take too much work for any one person to do.

This is where my idea comes in, all of us who uses this board pick an issue and just scan that one issue, gives credits to the person who scanned it (if the person wants credits) and then upload it to a website which I can easily put together (I have HTML and Java experience plus great software like Dreamweaver and Flash).

So ok, lets start, I'll scan Issue 100. I really hope others will follow with other issues, it would be great to have every issue right online, ready to read at will :-)

John Kavanagh

comp.sys.amstrad.8bit—Date: 2001-08-12 10:54:30 PST

cpcoxygen.digi-alt.net

CPCOXYGEN

cpcoxygen.digi-alt.net