April-June 1988 Volume 2. Issue 2.

Scorpio News

Scorpio Systems
P.O. Box 286 · Aylesbury · Bucks · HP22 6PU

Contents

Editorial

Nobody noticed ?

The last issue of **Scorpio News**, Volume 2, Issue 1, had a mistake on every inside page – and nobody noticed ! This is the trouble with computers – I only made the error once, the software turned it into 36 mistakes !

COMPEC

In the last issue I went on about how boring COMPEC was, and I also pointed out that Gemini had a stand there. However, I gave no details as to what products they were exhibiting. Their main stand was actually divided into a number of smaller stands, and there were a lot of new products being shown for the first time. So what were they ? Well, I think that Gemini were actually trying to keep that a secret from the visitors !

What I mean by that is that nothing had a label on it to say what it was. For example, on one stand there were 3 Gemini "ODIN" frame systems, all looking downright identical. A bit of investigation (i.e. asking a stand representative) revealed that they were all 80–BUS systems, one with the 64180 CPU board, one with the Z280 CPU board, and one with the 80286 CPU board (all prototypes, by the way). Floppy based ODIN systems start at £1795, 20MB Winchester ones at £2550. There was also a wall display showing most of the available 80–BUS boards.

On another stand they had a unit that looked like a Gemini Challenger (their 68000 based system), but in fact it was a prototype of their new AT–compatible, as briefly described by Dave Hunt in the last issue of **Scorpio News**. Complete monochrome systems start at £2470, up to 140MB colour systems at £4080. On the same stand there was a genuine Gemini Challenger, this one running SimTac, a program for training Air Traffic Controllers. This hardware/software bundle can be yours from a mere £25000.

Yet another stand had a number of terminals and an Atari–ST, all hooked to a 68020 based system running the Mirage Operating System. You were supposed to believe that they were connected to a system containing Gemini's new 68020 board, but this wasn't actually available in time for the show and so there was a certain amount of cheating taking place. The price of the 68020 board ranges from £1985 to £5870, depending on RAM and FPU options.

The DX–3 system had a stand of its own. DX–3 is the latest in the Gemini Multi–Format series of systems, DX standing for Data eXchange. This has been described in **Scorpio News** in the past. DX–3 prices are no longer published for some reason.

A further stand was dedicated to the GM879 Falcon graphics board. This is a largely Pluto–compatible board. However, it uses the Hitachi 63484 ACRTC, and has on–board genlock and B/W & RGB frame grabber. It also has an in–built SCSI port, meaning that Winchester drives can be attached (via a suitable controller) to give extremely fast picture load and save times. Falcon prices range from £1650 to £5065.

On the final stand there was a group representing an INMOS distributor. They were there to "plug" the Transputer, because Gemini are now into those. This is a project in which they are collaborating with Glasgow University in order to produce a number of Transputer based products. These start with an IBM PC style plug–in board with a Transputer and 4 Mbytes of RAM at £3385 (COMPEC price – see later for an update). Then there are various different triple–Eurocard boards that may be used together. Quoting from a Gemini data sheet:

"For example; four Gemini GM8200's (i.e. 32 T800s & 128 Mbytes memory) and one Gemini GM8201 (T800 & 64 Mbytes memory, graphics processor, 8 T800 graphics transformation pipeline) housed in a powered rack, a colour monitor, a host computer, and Occam, Fortran, C and Pascal compilers costs less than £120,000 delivered and installed !". I'm lost for words ! And the final line in the same data sheet, just to put things into true perspective: *"The GM8200 has eight light–emitting diodes, each of which light when the error line on its associated T800 becomes active"*. Wow !

Nascom anniversary

As regular readers will know, last November was the 10th Anniversary of the launch of the Nascom 1 micro–computer – one of the first micros available in the U.K., and the origin of the equipment that 99% of you are using. I asked for interested people to write in and say how they would like to celebrate this, and then arranged the get–together that we eventually had.

We all met at a conference hotel in Bath for the week–end (Bath being a reasonable average distance from everyone's home). We persuaded various 80–BUS dealers and manufacturers to subsidise the event, and in the end he cost to the attendees was virtually nothing (beyond the cost of getting there, plus a drink or thirty !).

Friday evening was just a free for all in the bar, with everyone getting to know each other, and all of us putting faces to names; **80–BUS News** and **Scorpio News** contributors present included Dave Hunt, Richard Beal, me !, Doctor Dark (alias Chris Blackmore), Clive Bowden, Rory O'Farrell, etc., etc.; dealers included Brian Wingfield from Arctic Computers, David Searle from Kenilworth; representatives from the relevant manufacturers; and so on. Various little sub–groups formed and some discussions went on into the early hours.

Saturday mid–morning was a more formal event. Yours truly did the chairing, and we had presentations of various new (and not so new) 80–BUS/Nasbus compatible products. Some of these were then made available to those present at very respectable savings, and a number of people eagerly took advantage of this.

Saturday afternoon was "free". Some people stayed around and chatted. Others had their wives and girl–friends with them and went off for a look at the local sights and (unfortunately and expensively for some !) shops.

After an extremely pleasant (and rather dopey making) meal, Saturday evening consisted of a few brief, but very interesting talks, on a number of different topics – the pros and cons of different programming languages, "did you know ?"s about various 80–BUS software, BIOSs etc., some hardware tips and hints, etc. There was then a question and answer section, and finally retirement to the bar and/or bed.

Sunday morning, after a rather tasty "brunch", was "set up your own equipment" time, with various attendees showing off their systems and software, and much software swapping (own–written and Public Domain I was assured by all !). There were some very impressive systems on view, and it was difficult to realise that they were all based on the same board range.

Mid–afternoon the event officially ended, and people said their sad farewells and disappeared into the distance to their homes and families. Many friendships had been struck, and I am sure that many will maintain contact for a considerable time.

Thank you to all that attended, and to all that contributed, either in speech or financially. It was a great week–end, and perhaps we'll all meet again in another ten years, or even sooner if the demand is there !

New material

In the last issue I made a request for more material for possible publication. Nearly every issue we seem to be reliant on "the faithful few". For this we are very grateful, but they've made their contribution time and time again. I therefore asked for new authors, and the result of this should start to be seen in this issue. Any more ?

I have to make an apology (this seems to be becoming a habit – see last issue) for two articles that should be here but are not. These were supplied on Gemini 96tpi format disks. Unfortunately the 96tpi drive in my Gemini system went faulty under warranty. At the time of writing I have been without it for 5 weeks. Without it I obviously can't read the disks, and thus can't include what promised to be very interesting articles. Sorry.

Printers

The last issue of **Scorpio News** was printed by a new print company. I'm afraid that, although produced quickly, the quality left a bit to be desired, being over–inked, and with bad alignment from page to page. I hope that this issue is an improvement.

Current Status

And finally, a few notes on the current status of various new Gemini products. This information is based on a Gemini Dealer Newsletter distributed in late December. Io Research have now ceased production of the Pluto 1, so Gemini has taken this over. The GM886, 80–BUS 80286 board, is in production. The GM880, 64180 board, will be "available in early March", but as far as I am aware it is still a way off. The GM890, Z280 board, will be "available in May". The GM891 MODEM, V21/V22 (but NOT V23), is in production. And the PC–compatible Transputer board, mentioned above, is now "only" £2995 (2MB), £4295 (4MB).

Letters to the Editor

Supporting Scorpio News

Dear Sir

When **Scorpio News** hadn't come by the third week of February, I began to suffer withdrawal symptoms, and to think "Paul must have made off with the subscriptions". Then a tiny envelope postmarked "Aylesbury" arrived. They are returning my cheque, I thought. Imagine my thoughts when I opened the envelope and found it contained the new issue ! But if we do not support the magazine, it will get so small that it will only exist in our imagination, which I for one would consider a great pity.

Despite the limitations of 64K, a Gemini/MAP/Nascom CP/M system is very adequate for most home use. The hardware is first class in quality, there is still plenty of hardware support (consider the new Z180, Z280 and 80286 cards for example), and one can "get at" the software and hardware in a way that is much more difficult with the newer PC compatibles.

With the "swing" to PCs and compatibles though, many of us probably have dealings with both systems, and it is inevitable that the magazine has to give support in this direction too. The recent very interesting articles on DRH on attaching various drive formats to an AT–compatible is a typical example of the way that we can continue to help each other. We should also remember that Gemini are now producing an AT–compatible series of machines, as well as the 8088 and 80286 cards for the 80–BUS, to say nothing of their Challenger and Transputer machines.

Ever since my first machine, I found all of the various Nasbus/80–BUS magazines invaluable as a source of learning and information. More recently I have felt able to contribute some ideas of my own. As the Editor said in the last issue, contributors don't get paid (boo–hoo). My reasons for sending in stuff is that it may help someone with a similar problem, and it helps to keep **Scorpio News** afloat.

There are two things that you, the readers can do to keep the magazine alive. The first is to send in articles. The second is to publicise the magazine to friends and get them to subscribe.

Yours, Clive Bowden, Nr. Truro, Cornwall.

Re. Drivetec drives

Dear Sir

I found Michael Hendry's article in the last **Scorpio News** very interesting and I believe that I may have answers to his two outstanding problems.

1. Slow writing to disk during disk copy

I had a similar problem to this when fitting 8" disks with the same track format that he is using. The problem was found to be in the BIOS. After a write operation and before doing a step operation, the drive's electronics require a small delay, so that it can empty its buffers and allow the flux around the head to collapse. In my BIOS this is implemented as a 2 msec delay after EVERY sector write. The fix is to delay 2 msec before each step operation if the last operation was a write. Disk copies will then write at the same rate as the read.

2. Lost sectors if motors have timed out before a write.

If I understand the drive manual correctly, it appears that the sequence the drive goes through is: –

(a) motor starts
(b) head load
(c) 2 index pulses sensed and READY (pin 34) enabled. (But drive may still not be up to speed.)
(d) fine stepper centres head on track and READY/SEEK COMPLETE (pin 4) is enabled

I suspect that Michael Hendry's BIOS is writing to a disk as soon as READY is active. This means that he may be attempting to write to disk while the heads are still moving or the disk is not yet up to speed.

I think that with these drives the BIOS should wait for READY/SEEK COMPLETE before initiating a read or write command. The motor start may also need up to a 2 sec. delay to allow the motor speed to stabilise.

From the above you may have guessed that I have managed to purchase 2 Drivetec 320 drives (from Matmos).

Yours faithfully, N.C. Back, Burgess Hill, West Sussex.

Modification of GM805 Disk Systems

by L.C. Waring

Old Gemini GM805 units, which use Pertec FD250 disk drives, can be converted to 40 track, 180K formatted capacity by a few simple modifications.

With reference to the diagram of the drive unit on page 22 of the *"Pertec Operation and Service Manual"*, these modifications are:

1) Removal of head loading mechanism (22). The head loading solenoid is secured by one screw at the side of the drive. This screw should be removed, the two connections to the solenoid cut, and the solenoid itself discarded.

2) The distance of head travel must now be increased. To do this the clamp support beam (10), secured by screws (1), must be removed. Care must be taken to avoid damage to the index photo transistor assembly (19). The carriage shaft (25) is held in position by a clamp, not visible in the diagram. The two screws securing the clamp should be removed. The lugs of this component are asymmetrical, and it is the broader lug which restricts head travel. This should be filed on the inside until it is slightly thinner than the other lug. The heads should now be able to travel inwards a sufficient distance to allow 40 tracks instead of 35 per side. The carriage shaft (25) and clamp support beam (10) should now be reassembled.

3) Adjustment of head pressure may be necessary if it is found that errors occur when accessing tracks 35 to 40 on either side. This may be achieved by placing some packing between the end of the head pressure spring and the head assembly (20). It was found that about 5mm of packing increased head pressure sufficiently to enable correct reading and writing of tracks 35 to 40.

4) The disk primitives must be altered to correspond to the increased number of tracks available. *[Ed. – With Gemini BIOS 3.2 or later this merely means increasing the number of tracks in the relevant .CFG file.]*

The writer has modified several disk drives in the above manner, and all have subsequently performed satisfactorily.

Notes on CP/M 3 Password Protection

by A.G.H. Sibley

Introduction

The CP/M 3, or CP/M–Plus, operating system for 8080/Z80 machines provides two levels of password protection, namely a disk restriction and a file protection system. Exactly how the passwords are stored in an encrypted form and how the programmer may access these areas to decode the password, in the event of a loss of memory faculties, is explained below.

In order to use password and date time stamping of files, a CP/M 3 utility, INITDIR.COM, is run to reorganise the directory area. This creates space in the disk directory for the date/time stamping and passwords but at the expense of halving the number of directory entries. The area which is created is known as the Extended File Control Block, XFCB.

Label XFCB construction

If the directory area on a disk (which has been label stamped) is examined using a suitable utility like DU.COM, then the following should be seen:

```
DISK UTILITY v8.6
Universal Version under CP/M 3.1

Type ? for help
Type X to exit

00 00534554 20202020 20434F4D 00000051 *.SET     COM...Q*
10 02030405 06070000 00000000 00000000 *................*
20 20544553 54202020 20202020 81440000 * TEST       .D..*
30 64646464 75757575 12070000 12070000 *dddduuuu........*
40 00444154 45202020 20434F4D 00000016 *.DATE    COM....*
50 08090000 00000000 00000000 00000000 *................*
60 00444952 20202020 20434F4D 00000072 *.DIR     COM...r*
70 0A0B0C0D 0E0F1011 00000000 00000000 *................*
```

A label directory entry has the first byte as 20H, rather than the normal 00H or user number. The password is encrypted in the area normally occupied by the block information as shown below.

```
   Label header       Label name     Password "key"

20 20544553   54202020 20202020 81440000 * TEST        .D..*
30 64646464   75757575 12070000 12070000 *ddddduuuu........*

                      Encrypted password
```

The important part, however, is entry 13 above which is the encryption "key" used to form the encrypted password stored in positions 16 to 23. In this case, the password was 1111.

File Protection

Password protection for a file has the user extent type 10H. An example is shown below where the file FRED.COM has been protected in the following way.

A>SET FRED.COM [PASSWORD=FRED]

The main directory entry for FRED.COM is unaltered, but the CCP must scan the rest of the directory area to find if an entry has the extent 10H. An example is shown below:

DISK UTILITY v8.6
Universal Version under CP/M 3.1

Type ? for help
Type X to exit

G=00:03, T=1, S=4, PS=0

```
00 00445520 20202020 20434F4D 0000003B *.DU        COM...;*
10 22232425 00000000 00000000 00000000 *"#$%............*
20 00444255 47202020 20434F4D 00000030 *.DBUG      COM...0*
30 26272800 00000000 00000000 00000000 *&'(.............*
40 00465245 44202020 20434F4D 00000016 *.FRED      COM....*
50 292A0000 00000000 00000000 00000000 *)*..............*
60 21000000 00000000 00000000 00000000 *!...............*
70 00000000 00000000 00000000 00800000 *................*
```


(XFCB)

G=00:06, T=1, S=7, PS=1

```
00 00534944 20202020 20434F4D 0000003E *.SID       COM...>*
10 50515253 00000000 00000000 00000000 *PQRS............*
20 00535745 45502020 20434F4D 00000056 *.SWEEP     COM...V*
30 54555657 58590000 00000000 00000000 *TUVWXY..........*
40 10465245 44202020 20434F4D 80A10000 *.FRED      COM.!..*
50 81818181 E5E4F3E700000000 00000000 *....edsg........*
60 21000000 00000000 00000000 00000000 *!...............*
70 00000000 00000000 00000000 00000000 *................*
```

Again, the XFCB structure is similar, but the header can extend from 10H to 1FH according to the user number. The "key" byte at entry 13 is again used to encrypt the password and store it in positions 16 to 23.

The Encryption

The "key" byte at entry 13 must be stored, along with the encrypted password, on the disk for the CCP to check if the correct password is specified. The key byte is constructed from the initial password specified in SET.COM, as shown below:

SET FRED.COM [PASSWORD=FRED]

PASSWORD = FRED^^^^ = 46+52+45+44+20+20+20+20 = 1A1H

A1H is therefore stored at entry 13 and used to encrypt the password in the following way:

FRED	=	46	52	45	44	20	20	20	20
KEY	=	A1	A1	A1	A1	A1	A1	A1	A1
RESULT	=	E7	F3	E4	E5	81	81	81	81

The string is then reversed and stored in the XFCB position for the encrypted password.

STORED ENCRYPTED PASSWORD = 81 81 81 81 E5 E4 F3 E7

Decryption

As an Xor operation is performed (essential in order to decrypt it), decryption is simply the reverse sequence. The LABEL password will be recovered from the earlier example:

```
20 20544553 54202020 20202020 81440000 * TEST        .D..*
30 64646464 75757575 12070000 12070000 *dddduuuu........*
```

STRING	=	64	64	64	64	75	75	75	75
ENCRYPT	=	75	75	75	75	64	64	64	64
KEY	=	44	44	44	44	44	44	44	44
XOR	=	31	31	31	31	20	20	20	20
PASSWORD	=	1111^^^^							

Program Solution

In order to simplify the somewhat lengthy decode process using DU.COM, a utility is available from the Author, by post only please, for decoding label and file passwords. A £5 handling charge covers the postage, packing and supply of a disk. Please specify what format you require and make cheques payable to the Author. The full commented Z80 source code, complete with documentation is supplied and most 5.25" formats, 48tpi and 96tpi including all Gemini/Nascom/MAP formats can be handled.

Mr. A G H Sibley
43 Winterbourne Road
ABINGDON
Oxon.
OX14 1AL

Private Adverts — 1

Disk Doctoring

by P.D. Coker

Over the past few years, I have accumulated a number of computers either as bits and boards or as more or less working entities. One of my more recent acquisitions was a strange beast called a Samurai (produced by the Kokusai Electric Co. of Japan). It was sold by Matmos as a 16-bit small business computer, has 8" drives and runs both MS-DOS and CP/M-86. Matmos supplied some software and a few disks and, by one of those strange quirks of fate (like getting a reply from certain 80-BUS manufacturers), I was offered from another source, a couple of hundred 8" disks, used once, one careful owner, etc. Without hesitation, I bought them, or rather did a deal...

So, what's all this got to do with 80-BUS machines ? Read on.

The usual type of 8" drive operates in single-sided single-density (SSSD) mode and has a capacity of about 243 kbytes. More modern versions sometimes occur in equipment or on the second hand market which are capable of operating in either SSSD or double-sided double-density (DSDD) mode. In the latter case, this gives a vastly increased capacity (round about 1.14 MBytes). Several 80-BUS FDC cards are available which cater for these, and BIOSs can be modified to take advantage of the 8" drives, so if you can get hold of cheap 8" disks, why not have a go !

A few years ago Dave Hunt wrote a very interesting article on disks in *80-BUS News* in which he pointed out that the disk base material was coated with magnetic material on both sides and that selection for single- or double-sided disks was made afterwards. This means that it is always worth trying to format both sides of a single-sided floppy disk — in my experience well over 90% will function quite happily as double-sided.

Standard 5.25" disks have the index hole in the same place, regardless of whether they are single- or double-sided but 8" disks don't. If an 8" disk is designed to be used in SSSD mode, the index hole is almost vertically above the read/write slot (in fact, a few millimetres to the right). If it is a double-sided, double-density type, the index hole is offset to the right by 20 mm or so. Another peculiarity of 8" disks is that the write-protect notch is on the bottom edge and has to be covered if you wish to write to the disk — devotees of this format say that the 5.25" system with its write-protect notch on the side is peculiar — but who's worried.

The Samurai system has two splendid NEC drives and can read both SSSD and DSDD disks. About 20 of my disk bargains were proper DSDD types and I decided to have a go at altering some of the rest from SSSD to DSDD. The technique is quite simple, and, if carried out as described, is unlikely to damage the disk.

With a ruler, mark a point 66mm down from the top edge and 84 mm in from the right hand edge of the disk using a soft pencil. Using a narrow–nosed single hole paper punch (obtainable from good stationers), punch a hole through the upper side of the disk jacket. Take care to avoid punching a hole through the disk itself or creasing it in any way !

Rotate the disk in its jacket so that the index hole appears in the new hole and, using a needle, punch a hole through the centre of the disk index hole and under–side of the disk jacket. Turn the disk over and using the hole punch as before; punch a hole centred on the needle mark on the under–side of the jacket. Finally, place an opaque adhesive sticker over the original index hole and check that the index hole on the disk is not obscured in any way (you may have to do a little trimming). The new hole should lie a little to the right of the original. Some of the disks which I acquired have no write–protect notch but since the 8" write–protect facility depends upon the notch being uncovered, these disks are not protectable. However, if you punch out a notch on the bottom edge, about 5 mm long and at least 2mm wide, centred exactly 4.4 mm from the right hand end, the write–protect facility is activated.

The disk should format correctly in DSDD mode if your software has been appropriately modified and the drive is designed to do this, but if your disk will not format, check that the new hole is in the right place. If, in spite of your best efforts, the doctored disk won't format in double–density, it can usually be formatted as a single–density single–sided disk – just transfer the adhesive patch from the old hole to the new one.

I have modified over 40 disks in this way with only one failure – due to over–enthusiastic use of the hole punch whilst drinking parsnip wine – there's a moral here somewhere.......

Next Issue

The dates for the next issue of **Scorpio News**, Volume 2 Issue 3, are as follows:

Letter/articles in by 01/06/88; adverts in by 08/06/88; newsletter posted to you by 01/07/88.

Doctor Dark's Diary – Episode 39

Artificial stupidity – a doctor writes...

As a doctor, I am often asked, *"Doctor, is artificial intelligence possible ?"* (I would just like to apologise to Private Eye for borrowing one of their better jokes, and to all the real doctors out there. I am not a doctor at all, so please don't send me long, whinging letters about your unpleasant condition, unless it is really hilarious and I can get a laugh out of it.)

I contend that AI is possible. I would be silly if I claimed it was going to be easy, or even soon. But it is possible. Machines can be made which will think. I nearly slipped up and wrote "think like you or I" then, but that would have led to a trap. The way we think has evolved over a long period, and is very complex, to say the least. The brain is not just an electrical circuit, it also carries out chemical reactions that are part of its processing.

As if it were not already complex enough, there is now evidence that the tiny "wires" between the cells of the brain reconfigure themselves as part of the way the brain works. So what was referred to in the science fiction of thirty years and more ago as "a giant electronic brain" is unlikely to be built to work in the same way as an actual brain. Even now, we don't have the technology to make a machine that reconfigures itself, is apparently all processor, and can build Nascoms for fun.

But we don't need to build anything at all. We already have computers that can carry out simulations of all sorts of things, including the operation of complex electronic circuits, and chemical reactions. Even the reconfiguring trick could be reduced to a self modifying program, or perhaps it would only be necessary to alter the data base holding the description of what was being simulated.

Research into neuronal computer circuits has shown that learning can be carried out by surprisingly simple machines. Simulating them in software on an ordinary computer takes a long time because of the von Neumann architecture of these machines, but more and more computers with many processors are appearing, and on these, the simulations would run rather a lot faster. Basically, the above explains why I believe that machines will one day be able to think.

In the meantime, machines with all the intellect of a stunned earwig can do a surprisingly good imitation of thought, and apparently hold conversations that some people find convincing. The program "Eliza", which was also known as "Doctor", and was written by Joseph Weizenbaum is the most famous example of this sort of thing. There was also a program by Kenneth Colby, called "Parry" which simulated paranoid behaviour. Douglas Hofstadter quoted in "Godel Escher Bach" from a "conversation" between two machines, one running each program, which went in part as follows:

Parry	: *I don't understand your motives.*
Doctor	: *Why don't you understand my motives ?*
Parry	: *Shouldn't I be ?*
Doctor	: *Why do you say that just now ?*
Parry	: *That's just the way it is.*
Doctor	: *Suppose that's just the way it were not.*
Parry	: *So ?*
Doctor	: *I am not sure I understand you fully.*

An enormous understatement at the end there. In fact, all the programs were doing was scanning the input for recognised words and producing a reply to fit them, from a selection of possible replies. A program to do just this sort of thing, called "Eliza", was printed in the ancient publication from the dawn of time, "Nascom BASIC Programs". It was very effective, but it never learned anything new.

Slightly more recently, I converted and improved the old "Eliza" program so that the user could teach it new replies. That program was written in EBASIC, a language that came with early copies of CP/M, and I once sold three copies of it. Our Editor has suggested that a disk library might be a useful addition to the services he provides, so I shall be sending him these ancient, creaking programs in the hope that people will want them, either as examples of pitfalls to avoid, or a foundation to build on. My version of the program makes the same "I/me" mistake as every version I have ever seen, and (I love it when I get to say this one) I leave it as an exercise for anyone who gets hold of the program to remove this long lived bug...

One application of these conversation imitating programs that I thought of a while ago, while planning what facilities to include in the Prestel software I was going to write, was a version of the program that would scan all the new frames appearing on a Micronet Chatline, and if they started with the word "Marvin", a reply would be generated, and sent to the Prestel computer. Alas, this particular software project has gone halfway to completion twice now, and the two halves are both first halves and won't work together. I have, however, learned quite a lot about how to plan large programming projects ! In the meantime, anyone who wants to amend their Prestel software to carry out this strange idea can help themselves. When one considers some of the conversations that actually appear on the Chatlines, it is possible that this has already been done !

If £ < > $ then we are being done.

Well, in spite of the Prime Monster's efforts, the Pound is not worth the same as the Dollar. But when you compare prices of programs and hardware offered for sale on both sides of the Atlantic, you will find that the number is the same, only the sign in front changes. Modula 2 for the Amiga is $99 in the States, and £99 here. Somebody is doing very nicely for themselves out of this, with the most annoying example being Metacomco, who are actually in Bristol ! They sell their stuff more cheaply to Americans than they will to us.

So what can we do to combat this phenomenon ? Well, it is obvious, really. Import things. I was initially quite worried about this, but the easiest way is to pay by credit card. You just tell them the number, and that you want airmail, unless you do not mind waiting six weeks. The catch is that you may end up paying Customs duty. So far, I have only bought a book this way (Sp*c*tch*r, if you must know) which the Customs kindly ignored. Perhaps no duty is payable on educational material ? However, I plan to obtain some Amiga software soon, and will let you know if this leads to any problems.

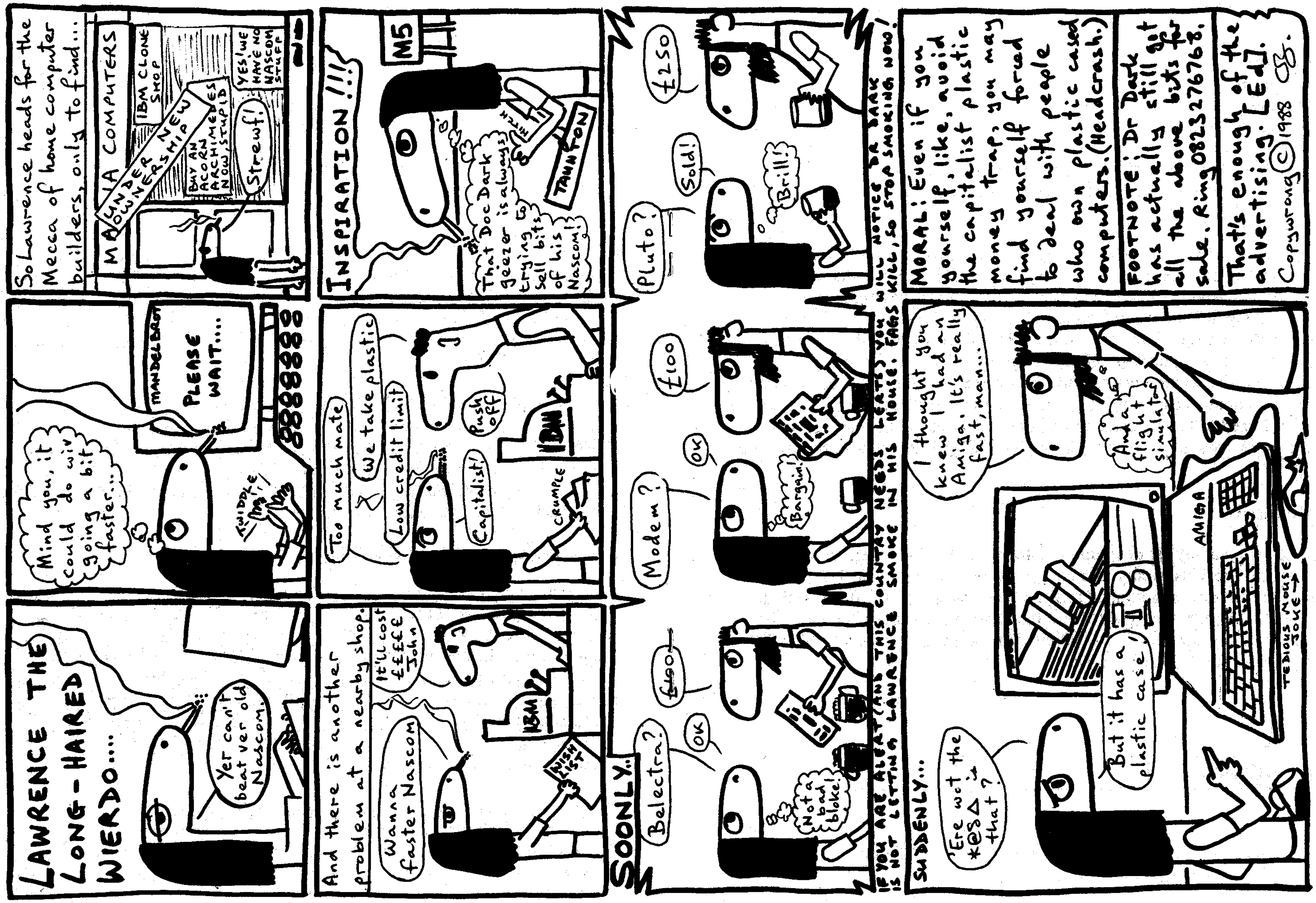
One thing it is easy to forget is that any hardware is going to expect entirely the wrong sort of mains supply, and you will blow things up if you give them 240 volts when they want 110. Less obvious is the fact that MODEMs from America don't understand our phone dialling system unless you are on System X exchanges and are able to use tone dialling. No doubt this is dealt with in the "Hackers' Handbook", but I haven't got round to buying that yet.

Oh gosh, not benchmarks again !

Well, only briefly. I benchmarked the Open University's VAX cluster the other day, using the UCSD Pascal they have on there for one of my courses, and got a quite accurate answer back in about 5 seconds. I have no way of knowing how many other students were using the machine at the time, so it may be an ordinary time, or it may be excellent. One catch I found was that the UCSD system has arctan, but does not have tan. Just divide sin by cos, obviously, but why on earth did they leave it out in the first place. Paddy Coker tells me that the Acorn Archimedes takes about two seconds, which is very nice if you can afford that sort of thing and don't mind writing everything yourself, using dear old BASIC...

Book – Algorithmics, The Spirit of Computing.

This was written by David Harel, published by Addison Wesley, and found in a sale at New Year, marked down from £15 to £1.50, and is an up to date survey of algorithmic methods. The author is clearly not a C fanatic, as he takes all his quotes at the start of the chapters from the Old Testament rather than anything by Lewis


LAWRENCE THE LONG-HAIRED WIERDO....
Yer can't beat yer old Nascom.
Mind you, it could do wiv going a bit faster...
MANDELBROT
PLEASE WAIT....
So Lawrence heads for the Mecca of home computer builders, only to find...
COMPUTERS
UNDER NEW OWNERSHIP
IBM CLONE SHOP
BUY AN ACORN ARCHIMEDES NOW, STUPID
YES! WE HAVE NO NASCOM STUFF
Strewf!
And there is another problem at a nearby shop.
Wanna faster Nascom
It'll cost £££££ John
IBM
WISH LIST
Too much mate
We take plastic
Low credit limit
Push off
Capitalist!
CRUMPLE
IBM
INSPIRATION!!!
M5
TAUNTON
That Doc Dark geezer is always trying to sell bits of his Nascom!
SOONLY...
Belectra?
£100
OK
Not a bad bloke!
Modem?
£100
OK
Bargain!
Pluto?
£250
Sold!
Brill!
IF YOU ARE ALERT (AND THIS COUNTRY NEEDS LEARTS) YOU WILL NOTICE DR DARK IS NOT LETTING LAWRENCE SMOKE IN HIS HOUSE. FAGS KILL, SO STOP SMOKING NOW!
SUDDENLY...
'Ere wot the *@$Δ is that?
But it has a plastic case
I thought you knew I had an Amiga. It's really fast, man...
And a flight simulator
AMIGA
TEDIOUS MOUSE JOKE →
MORAL: Even if you yourself, like, avoid the capitalist plastic money trap, you may find yourself forced to deal with people who own plastic cased computers. (Headcrash.)
FOOTNOTE: Dr Dark has actually still got all the above bits for sale. Ring 0823 276768.
That's enough of the advertising. [Ed]. Copywrong © 1988

Caroll. Still, I suppose *"Declare the things that are to come hereafter"* is rather appropriate in a book about formal approaches to programming !

There are three preliminary chapters introducing the basics of computing, and the book then moves on into discussion of algorithmic methods, correctness of algorithms and efficiency of algorithms. There is then a section on the limitations of computing, which is very interesting. It is concerned with the fact that some programs will take almost forever, some can not even be written, and some will run on incredibly simple machines. At the end of each chapter is a short description of recent (this was printed in 1987) research work on the particular subject of the chapter.

The final section concerns itself with getting answers to apparently intractable problems by using parallelism, probabilistic (coin tossing) algorithmics, and other fiendish methods. And finally there is a chapter on AI, which gives a good idea of some of the many difficulties that stand in the way of those who would like their Nascom 1 to explain the Universe to them. I found this an excellent book, as the mathematical aspects of many programming problems are explained very clearly. The chapter on what computers can not do is an excellent antidote to those who believe that "computers can be programmed to do anything that you want them to". I admit that I used to believe that, but it has been proved to be a silly idea. Even worse, the proof goes all the way back to Alan Turing, and ought to be common knowledge.

For Sale (still)

I was going to repeat my list of computer bits that are still for sale, but the lack of response to the previous version has convinced me that there is nobody left who wants a printer for almost nothing. Oh alright then. Belectra board with Pascal to work it, Gemini MODEM board with Prestel software, Pluto board with mini palette. Iotec Iona (contains switching psu), Ohio Challenger, Spectrum and VTX5000. Nascom programs on tape ! 0823276768.

end.

Private Adverts — 2

For Sale. Gemini GM837 Colour Card, GM816 I/O Card. With manuals, software, etc. Offers: C. Bowden, 0209 – 860480 (evenings).

IBM–COPY — A Review

by Kevin R. Smith

IBM–COPY is a program by TC Software which allows a Gemini computer to read and write files to IBM PC–DOS format disks. It costs £65 + VAT, and comes with a brief (11 A4 sheets) but adequate photocopied manual. This review is based on several months use of version 3.0 of the program.

To run the program it is necessary to configure at least one floppy disk drive on your system as having an IBM format. How to do this is described in your BIOS manual (version 3.2 or greater). It is unimportant which IBM format you decide upon, as IBM–COPY automatically determines the format of the disk being used, be it 8– or 9–sector, single– or double–sided. Only the high–density 15–sector disks used by the IBM AT cannot be used with this program.

On running the program, the screen displays a list of the available commands, and a reminder of the formats associated with each of your logical drives, followed by the program's prompt, IBMCOPY>>. The commands available are *COPY, DIR* and *INIT*. In addition, entering a "?" redisplays the initial screen display, and entering a "RETURN" by itself quits the program.

The most important command is *COPY*. This is used to copy files between CP/M and PC–DOS disks; it cannot copy files between two CP/M disks or between two PC–DOS disks, and can only access the current CP/M user area and the root directory of PC–DOS disks. In practice, these are rarely problems. The syntax used is similar to PIP, except that files cannot be renamed as they are copied. Ambiguous filenames are allowed for multiple file copying.

The *DIR* command is identical in function and syntax to the CP/M DIR command when used with CP/M disks. However, with PC–DOS disks, you are also told the size of each file, the amount of free disk space and the percentage of the disk which has been filled. Sub–directories and the volume label (if any) are also shown but cannot be copied (although no error message is generated by trying !).

The final command, *INIT*, allows you to clear the directory of PC–DOS disks, effectively deleting all the files on the disk. It can also change the PC–DOS disk ID byte, which determines whether the disk is treated as single– or double–sided and having 8 or 9 sectors per track. Attempting to use *INIT* on a CP/M disk produces an error message.

A number of options are available for each of the commands. These include suppressing checks when overwriting read–only files on a PC–DOS disk, and directing the output from a command to a printer. The latter is useful because control–P has no effect within the program, although it is still possible to get a screen–dump with EDIT and control–D. The *COPY* command also insists on being told the format (CP/M or PC–DOS) of the source disk as one of its options, even though the program knows the format associated with each drive.

The program's major limitations are its inability to erase or rename a file, or to overwrite a CP/M read–only file or disk (which generates that dreaded *BDOS Err on A:* message !). Perhaps someone will release a version 4.0 ! However, it is easy to use, so much so that it can be run without referring to the manual. Gemini owners who also use IBM PCs will find this an invaluable aid to their productivity. I found its main use to be transferring data files and source code for languages such as TurboPascal, which are available for both CP/M and PC–DOS.

[IBM–COPY is available from Gemini dealers, or from **Scorpio Systems** *by mail order for £65 + VAT (P&P free to subscribers).]*

"Real Programmers"

by V.A. Rious

Real programmers would rather use a bank of toggle switches and read a row of LEDs than use such cumbersome devices as keyboards and CRTs to talk to computers.

Real programmers don't write specs. – users should consider themselves lucky to get any program at all and take what they get.

Real programmers don't comment their code. If it was hard to write, it should be hard to understand.

Real programmers don't write applications programs; they program right down on the bare metal. Application programming is for feebs who can't do system programming.

Real programmers' programs never work right the first time. But if you throw them on the machine they can be patched into working in "only a few" 30–hour debugging sessions.

The DH Bit

by D.R.Hunt

Date–line February

Well I guess the topic of high capacity drives on IBM must be drawing to a close, but two further things have come to light. I recently borrowed a high capacity floppy disk (only) controller card made by DTK from Citadel Products of 50, Edgware High Street, Edgware (not to be confused with the Edgware Road), and surprise–surprise, it did the necessary. I could actually force it to format normal 96 tpi. disks to 730K and correctly in an AT clone. To my surprise it didn't even conflict with the existing hard disk/floppy controller card, though goodness only knows why ! I can only guess it used a different interrupt or port addresses or something, but the usual lack of information supplied, precluded my finding out.

Mind you, the need to squeeze the last ounce of capacity out of normal disks has eased somewhat, for me at least. New readers may like to have a brief resume of the story. My switch from Gemini with its 780K capacity drives to IBM ATs and clones brought a rude awakening. I could only get 360K of data on normal disks, and the high capacity disks actually cost a lot of money each – anyway I have a lot of normal disks, but object to using them at 360K knowing full well that they would format and work at up to 800K per. It finally came to light that the IBM AT disk controller hardware is so dedicated that using DRIVPARM to modify the format wouldn't work, so that left two choices. Use twice as many normal disks as I actually needed – or fork out anywhere between £2.80 and £4.50 per disk for high capacity disks.

I now have found another way round that problem but in the process, found another weirder problem than the fun we had deciding why the IBM AT controller wouldn't format normal disks at 730K in the first place !

The answer came in the form of a mail order company called *Mydisk* who sell disks at silly prices. The disks are unbranded but carry *Mydisk's* own label and a 2 for 1 replacement guarantee if they prove duff. Normal disks work out at about 37p each including VAT and postage (provided you buy by the box of 27 disks) and high capacity disks work out at a shade under a pound each by the box of 13 disks. Now 13 is an unlucky number, but it didn't stop me buying a box of each to see what made them tick.

Here comes the mystery. The normal disks seem perfect. They format faultlessly in anything; the Gemini at 780K per disk using Teac drives, a couple of Walters XTs using NEC drives, the IBM AT with Epson drives, and Walters AT clones with both 80 and 40 track NEC and Mitsubishi drives. Perfect in my anonymous AT clone which uses Korean Ho–Shin drives which appear to be a component for component Teac clone, even down to the positions of the screws securing the front bezel so the fronts are interchangeable with Teac. They even formatted in an old Shugart 48 tpi drive with notoriously low head current which usually refuses to format anything (although it will usually write to pre–formatted disks). They even work well in both the firm's old Commodore 8096s fitted with the infamously unreliable Commodore modified Micropolis drives.

How about the high capacity disks ? All I can say is the results are variable. All disks format perfectly in the IBM AT with the Epson drives. All but one formatted faultlessly in my clone with the Ho–Shin drives, and the one that failed came up with an inexplicable and undocumented DOS error message. 'Parameters incompatible' – remember this was a virgin unformatted disk – so where the parameters came from, I don't know ! I fixed that one by using a loudspeaker magnet which erased whatever was on the disk, which it read before it tried to formatted the disk. The Teacs in the Gemini couldn't be tested as they are only 'normal' drives with no double data rate ability. The oddballs were the NEC and Mitsubishi drives in the Walters. The Mitsubishi would format most disks most of the time, but occasionally, they would throw out error after error, flagging out anything between 10K and 500K in duff tracks. But the next time you formatted the same disk it would be perfect. The NEC was worse, only about 30% of disks formatted perfectly (and the same disks had worked perfectly in the IBM and my clone), the remainder would throw out anything between 100K and 200K of duff tracks, mainly on side '1'. The same disk would be alarmingly variable, one time round it would be 100K of errors, the next time, 200K. The only thing noted was that more times you formatted them, the number of errors on average was less. I suppose if you formatted them enough they'd end up Ok, but I don't trust these disks in the Walters.

Did I try *Mydisk's* 2 for 1 replacement – no I haven't yet. The disks are Ok in my machine and once formatted seem to read and write Ok in all the others. The errors seem to be only on formatting. I can only guess that the head currents of the drives differ and the disks are a bit marginal on side '1'. Sometime I'll think of someway of proving this. On the other hand perhaps I won't. The disks seem to work Ok and they are cheap. Would I recommend them ? or will I buy some more ? If you have a genuine IBM AT with Epson drives, I don't think you'd ever discover anything was amiss, otherwise you could have problems. Yes I think I'll buy some more.

Date–line March

It had to happen, didn't it ? The glee with which I used to receive customers at Henry's when they told me the sad story that their hard disk had gone down, and they hadn't kept backups because the machine had never given a hint of trouble. You'd think I'd know better wouldn't you ? Last Sunday night, having just finished this article (Paul says I'm late again), and about 8 hours of work–work over the weekend, I was just about to copy it all off when – do I have to tell you ? Well it wasn't actually the disk which went bananas, the controller card died. Perhaps it had got fed up with me tinkering with it or something. The error message said the controller was faulty, not even the floppies worked so a quick leap into the car and drive into work and I swiped the controllers out of one of the Walters and the IBM. Plugged one of them in and sat back praying I'd get my data back. No such luck. Both cards woke up Ok, the floppies were alive, but both reported a 'General Read Failure' on the hard disk.

Now "hard disks" is not a thing I've looked at, and I know very little of what actually goes on inside. The thought occurred to me that as hard disks are definitely fixed in the computer and allied to the controller card that drives it, they don't all have to be formatted the same way. Some spurious mention in the only book I'd got about hard disks said something about the 'length of the ECC burst' and implied that different controllers set different lengths. Now I deduce the ECC is the Error Correction Code, like the one and two byte CRC's used on floppies, but does each brand of controller do it's own thing, and vary the format to suit itself ? Well, it looks that way. The hard disk is originally formatted with the controller on board, so it guarantees the two work as a pair. Certainly both the Western Digital controller out of the Walters and the IBM controller from the IBM weren't going to talk to my hard disk.

That left the problem of possibly a duff hard disk as well as finding a new controller to replace the dead DTK one. Monday was out, I'd got to make a trip, so I stood no chance of getting in touch with any suppliers that day. Tuesday dawned, and acute computer withdrawal symptoms were setting in. A few phone calls and no DTK controllers in sight, plenty of offers of others, but no help as to whether they'd exactly replace the old one, *"Who knows ?"*, was the usual reply, *"You usually reformat the disk on replacing a controller, that's what backups are for !"*. Another thing was the going price, this is my computer we're talking about – the Company aren't going to pay. Apart from anything else, I've had an expensive couple of months, and the thought of £150.00 for a new controller didn't please me. In the end I gave up, I'd have to format the disk, if only to see if it was still alive. In the meantime I'd returned the two controllers I'd swiped as the owners of the machines might have got a bit cross to find someone had been vandalising their working tools, so I couldn't even do that. Late Tuesday afternoon brought me to Technomatic at Burnley Road, Neasden. I know the lads at the Edgware Road branch, but I'd never been in the

Neasden branch, but the relief, I was known and greeted like an old friend. Yes, they had some controllers, would I like to borrow one to see if it works ?

It was totally anonymous, no manufacturers markings at all, just enough to guess it was Taiwanese. It was a totally different design, no recognisable controller chip numbers, but incongruous chips from the Z80 chip set lying about (but no SDLC or HDLC controllers as I might expect), it even came with a manual, although that turned out to be a little more than useless, as it contradicted itself every other page. I rushed home and plugged it in. This was worse, the drive didn't even select. I spent the rest of the evening trying to get some sense out of the manual and moving links about. Simply no joy. Back at Techno's on Wednesday, this time with the drive. I took about two minutes moving the links about (in an undocumented fashion) to make the drive wake up, but of course, it wouldn't read it. Ok, format it ! Fine, it formatted and no drive faults, it's all back in order.

Back home again, screw it all back together, and fired it up. It works ! I've lost two days work on the disk, and spent the best part of another day chasing around trying to get it fixed. But it works and it cost a lot less than a new Western Widget one. My thanks to the guys at Technomatic for the trouble, effort and advice. Not even the DH condescending glee at the thought of another guy who doesn't keep backups, just commiseration.

I've put the new card through it paces now, it's very slightly slower, it only shovels data at about 140K/sec. rather than the 167K/sec. of the old, but I'm not complaining. Oh, and the good news, give it the right DRIVPARM statement and all my old DDDS disks can be used at 730K – no hassle.

Back to what I intended to write.

So on with the original article, this was written for someone else, I say this unashamedly, and if the 'Dodo' approach shows, then I don't apologise. Fortunately, the article itself wasn't lost in the disk crash, only the 'tweaks' (mainly deletions of the obvious) done for this reprint, I've had another go at it today, but if you find bits which don't add up, or look like they've been badly edited, it's because they have – sorry.

Now, I've almost completed the latest project for work, the largest yet and it looks the most saleable. It's a Digital Imaging System. The latest idea for replacing paper in the Corporate office, or small office for that matter. So I guess it's about time I gave you some idea what imaging systems are about. As I said, this article is the basis of a small booklet (with pretty pictures, when I get round to doing them) which we will publish at work to help the numerous but ignorant visitors we've had looking at the system. So keep in mind that it is aimed at the Office Manager, Senior Filing Clerk and the 'Computer chap' they inevitably have in tow when they visit.

Digital Imaging Systems – Part 1

by D.R.Hunt

An introduction for the uninitiate and the unwary.

There's a lot of hype going around information management circles about Digital Imaging Systems and about Optical Disks. So much so, that it's beginning to get me annoyed !! It takes a lot to get me annoyed, but of late I have been party to receiving so much specious, doubtful and down right incorrect information, that I feel in my own way I should start to put the record straight.

What this is about.

This particular paper is about imaging systems and the technical side of it at that. I find that it's on the technical side that potential users feel most ignorant, and at the same time are at the mercy of equally ignorant sales people (not that this paper is going to be heavily technical, just enough so's you know what people are talking about, sales persons and potential buyers alike). I don't propose to look at the alternatives, only to warn you that imaging systems aren't God's answer to the filing clerk and are designed to satisfy certain fairly rigid criteria. It most certainly does not mean that imaging systems are the all embracing answer to all your filing problems. Just because it's the newest approach, doesn't necessarily mean it's the best answer to your particular problems. Just a thought – I wonder why salesmen don't like me, could be because I say things like that.

A few technical purists who read this are going to be upset. If, as I suggest, I know so much about it, why isn't this paper presented in a precise and formal way, without the sweeping generalisations, the screaming approximations and the simplified explanations, and why isn't it written in formal English, instead of the appalling informal style I've chosen ? Well, all I can say to you Sir, is that this isn't written for you, you probably know it all already, having read those awful wordy boring reports and papers I've had to read to understand what it's about. This has been written for those who wish to know generally what's going on, not the precise details. I know I use generalisations, and approximations. It's fine in its place, for guys like you and me who have to actually understand precisely what makes a thing tick, but it's still boring reading. I've written turgid technical stuff before, and believe me, I like writing it even less than I like reading it written by others. So if you don't like my writing style – tuf !

Beware of wrong definitions.

Where to start ? Well let's get one definition straight to start with. Perhaps my pet hate are those who refer to Digital Imaging Systems as 'Optical Disks'. Now this is simply wrong. The term optical disk simply describes a device which reads or writes information from/to a special disk using the light from a small low–power semiconducting laser — hence optical. It so happens that optical disks are usually used by imaging systems as the means of storing the images once captured. But they are only a small part of the system and that use is not the sole use to which optical disks are put. The converse is also true, in that optical disks are not the only way of storing the images. Storage to tape is quite practical, although admittedly, less convenient. Images could be stored on the hard disks of computers with considerably less technical complication than optical disks, although the quantity of images stored this way would be tiny by comparison; none the less, it's possible. So, please — less emphasis on the storage medium and more emphasis on what the systems are about.

What is an Imaging System ?

Ok, now let's start at the beginning. The idea of imaging systems is to store images, be they pictures, text, drawings, documents — or what have you, in some way that they can be pulled back and looked at in the future. At the same time, an imaging system must satisfy other criteria. The storage medium must take up less room than the original, it's no use replacing hundreds of files in cardboard folders with something which takes up the same or more room — that costs more. The information must be easily accessible by those who need it, and the cost of storage and access must be less than the cost of storing and accessing the information in its original form. Substantially cheaper if the high capital cost of any alternative storage medium is to be offset by the savings in cost of storage and access.

Mind you, many storage systems are unnecessary anyway. Very often, the custodians of storage systems don't ask the obvious question, *"What are we keeping this for anyway ?"* But that's not my problem. I think about alternative storage systems, it's not my job to philosophise over why other people should need them.

An imaging system consists of a number of discrete components: a scanner for getting the images into the computer; a display system for having a look at the images; a compression system for reducing the raw size of the images to something more easily stored; a storage system to keep the images; a hard copy system to reprint the images if desired; and most important of all a computer system to control what does what, and to find things once stored.

What does 'Digital' mean ?

Now the whole thing is 'digital' — is this another bit of hype from the promotions boys, as 'digital' doesn't seem to mean digital these days, it means a sort of woolly association with computers, latest technology and of course, is good for you whether it's got anything to do with being digital or not. Like the word quartz, another piece of 'word specialisation' which the promo boys have foist upon us. Quartz is a silicate mineral — a lump of rock, nothing more, nothing less — it's certainly not an adjective to be used in place of the phrase 'precision timekeeping'. That's not how the promotions guys would have it.

No, digital imaging is nothing to do with promotions, it is truly digital and a brief explanation of 'digital' is necessary. Digital information consists of binary codes, and binary codes are often referred to as a system of counting in '0's and '1's. This is true but confusing, better to say binary codes can represent almost anything if applied properly and the business of '0's and '1's is better presented as the presence or absence of something rather than anything to do with counting. Binary codes imply something to do with computers — not true. Binary codes could be written and manipulated using matchsticks. Line up a row of, say, twenty matchsticks all equally spaced apart. In this case we'll say the matchsticks represent '1's. The row of matchsticks represents a binary code. Now take some of the matchsticks away, may be just a couple perhaps all of them, perhaps none. The absence of a matchstick will represent a '0'. The code you are left with is another binary code — totally meaningless in this case as the original code didn't represent anything in particular in the first place, but a binary code none the less, and you didn't need a computer to manipulate it — to remove the matchsticks. Another couple of quick definitions — each individual '0' and '1' is known as a 'bit' and a collection of eight 'bits' is known as a 'byte'

The computer handles digital information, but as we'll see later, there's digital information and there's digital information. Important to our case is the difference between the handling of an image in a computer and the way in which a computer handles text. In our own minds there's a clear distinction between text in the form of written words on one hand and on the other hand, pictures. You read words — you look at pictures. The process is totally unconscious, your mind says *"This is text, read it"* or presented with a picture, your mind says *"This is a picture, look at it"*. But that's not to say that a piece of paper covered with words is not a picture. Think about that, stand back from any printed page and the text takes on grey shapes, you can't make out the words, but the shape of the blocks of text is quite distinct. Can you see that a page of words could be a picture ? Is the converse true ? A picture consisting of words is not necessarily text. Looking close up at a picture made up of words and the words might read as gibberish. So we have established a simple rule: a picture is a picture; text is a picture; but a picture is not necessarily text.

A computer, say when used as a word processor, normally stores text as the actual letters that compose the text but if it has to store pictures this is done completely differently, as a pattern of dots. An imaging system can't take chances, nor is it intelligent enough to make the decision between text and pictures, and anyway, even the best character reading software in not infallible and makes mistakes, so we can assume a computer can't read well. (I always thought the name Palantir for the supposed best OCR system around was either chosen by mistake or as a wry joke perpetrated by the designers, on the users. No I shan't explain that, for a definition of Palantir see 'Lord or the Rings' book 2, 'The Two Towers' by Tolkien.) To play safe the computer is sure of one thing, whether the image is text or a picture it can always be treated as a picture. That's what it does — hence imaging system — it always treats anything it sees as a picture.

The scanner.

Enough descriptions of what goes on, starting at the beginning, how do images get into system ? I've already said that the image is stored as a pattern of dots. The binary system is ideal for dealing with dots, it means that for a monochrome image, we only have to think in terms of black and white — '0's can represent black and '1's can represent white. The device used to convert the image on a piece of paper into a digital representation is called a scanner. These can be flatbed, like most photocopiers, or rotary, where the paper is popped in a slot and re–appears out of another. In a flatbed scanner, the paper stays still and the scanner sensor moves (or at least a mirror reflecting into the scanner sensor moves), in a rotary scanner the scanner sensor stays still and the paper moves. There are two common sizes of scanner, A3 and A4. A3 scanners are always to be flatbed, at least, I haven't seen a rotary A3 scanner.

The way a scanner works is simple. I guess you've an idea how a television picture is built up on a TV screen, it's drawn as horizontal lines, 625 of them in Europe, from the top to the bottom of the screen. The same in reverse for the scanner. As the image moves past the scanner optics, the image is projected on to the scanning sensor, the sensor is usually a device known as a CCD (Charge Coupled Device). An A4 size CCD consists typically of 3,500 minute discrete light sensors side by side in a strip. The 'Charge Coupled' bit describes the way they work and is of no concern at the moment. As the image falls on them, each sensor will either see light reflected from the paper, the white bits, or no light reflected from the paper, the black bits. They can also make some decisions about colour and half–tones. Some scanners don't use a CCD but mechanically scan the document with a single laser using a system of rotating mirrors. These are much faster, the resolution is very much better, but they have real problems with half–tones and colours and cost the earth.

Want to try it ? Take a piece of paper with a picture on it, a newspaper picture will do nicely, lay a ruler graduated in millimetres horizontally down on the paper so the

ruler is someway down the picture. Use a pencil or run your eye along the edge of the ruler. At each millimetre transition make your mind up if the area you're looking at is black or white. Ok, so some points will be difficult, about half a millimetre will be white and half black, either way, make your mind up, it's either white or black, 'don't knows' are not allowed. So as you 'scanned' along the line, you end up with a pattern of blacks and whites. Move the ruler down a shade and do it again. This is what the scanner sees. Black and white dots, and its output is electrical, in binary form representing the blacks and whites.

Now I've heard a lot of twaddle about scanning resolution. If you tried it yourself using a ruler, then that was at a pitch of 1mm. That's about 25 dots to the inch. Scanning resolutions are at worst 75 dots to the inch up to some real hairy scanners which work at 2,400 dots to the inch or more. Typical for imaging work is about 200 to 400 dots to the inch, scanned along the line, and the paper or scanner advances down the paper at the same rate, 200 to 400 lines to the inch. So for each inch square, there are some 40,000 to 160,000 dots. 300 dots to the inch is commonly used as being a trade off between resolution and speed. It's good enough to reproduce the finest detail the human eye is capable of seeing unaided and the amount of information produced is not so large as to be unmanageable. Of course, the higher the resolution used, the more detail scanned, but what's the point, you don't normally read books or drawings or documents with a microscope. So 300 dpi is adequate, yet even so, the information scanned at this rate is represented by over 8,000,000 dots to the A4 page. Pretty big.

The Display system.

The data from the scanner is temporarily stored in the computer's memory, and there are three choices from here: throw the data away and start again because it was the wrong document or something; have a look at it and then decide to save it or throw it away; or save it without having had a look at it.

It's usual to have a look before deciding what to do next, and that involves the display system. There's not a lot to say about the display system, except you get what you pay for. The simplest and cheapest is to use the computer's original display screen. This is usually the wrong shape (or at least for portrait displays) and it's resolution is low, typically 50 to 75 dpi. Don't worry about the loss of resolution in the picture you see, the image in the memory is fine and the resolution is unchanged, it's just some software reduced the resolution of the displayed image because the computer's display couldn't cope with the full resolution. Next type of display is the portrait shaped (approximately) A4 size screen. These usually display the whole of the image, but usually the resolution is reduced. These displays are typically resolve at about 100 dpi. This is entirely adequate for looking at documents and drawings to check the scan quality, but text can be a bit tiring to read. It's a trade off again, cost versus speed in this case. There are some A4 size displays which work at the full 300 dpi, but they are comparatively slow and cost a bomb. Lastly come the big (approximately) A3 size displays. Resolution varies, but is usually reduced from the original, more on speed grounds than on cost. Those which show the full resolution look superb but, brother, are they slow !!

Manipulating the image.

A point is sometimes raised as to whether the image can be manipulated. That depends on what you mean by manipulation. Sure, you could expand the picture to look at a part of it in full detail. If it was a small display screen you could shrink it to see the whole image in reduced detail, you could turn it sideways or upside down. None of these things change the image stored in the computer, they simply manipulate the way you see it. Now if your idea of manipulation at this stage is to enhance the image, that to my way of seeing things is naughty. It's only too easy for the software writer to provide tools to allow you to 'tidy up' the image. Perhaps not a bad idea if it has scanned badly and areas are smudgy or indistinct, but these self same tools could also be used to add or delete things and at this stage any modifications would be invisible on the final stored result. The whole idea of imaging systems is to store documentation and this is best stored warts and all. There's always the touchy subject of 'legal admissibility', where documents might have to be produced in some legal proceedings. Now the way the law stands at present, documentation submitted must be 'best available evidence'. If the original paperwork has been destroyed, then the best available evidence is the images, and if it can be proved that it is not possible to tamper with the original images before storage, this stands a far greater chance of being legally admissible than something that has been 'tidied up'.

On the other hand, 'tidying up' an image is not a bad idea and might be vitally necessary to enhance obscure detail or something. So the imaging system could allow 'tidying up' after it has safely and irrevocably tucked the original away unaltered. That way the original is untouched, but a second copy could be enhanced in any way that was required, that's why automatic date and time stamping becomes an important feature of the system.

Legal Admissibility.

Just a few points about the legality of documents stored in an imaging system. In law the only legally admissible evidence is the original, even carbon copies of an original are doubtful and any form of machine generated facsimile of the original is even more doubtful. However, to work the law must be 'reasonable' and as I have said, actually works on the basis of 'best available evidence'. For instance, it is assumed in the microfilm world that microfilmed documents are legally admissible, this is not so, simply that microfilmed documents may be the best available evidence under given circumstances. There have been test cases to prove it on the grounds

that microfilm is difficult to tamper with. To date, as far as I'm aware, no such test test cases exist for documentation stored in an imaging system. However, the same criteria apply. Images stored in an imaging system are equally if not even more difficult to tamper with and would certainly would be accepted as the best available evidence under similar circumstances.

The Compressor.

The image is stored in the computer, which leads to the next stage. Vast though the capacity of an optical disk is, it's not going to hold many images if each takes up in excess of 8 million dots. A typical 12" disk would only hold perhaps a couple of thousand images at that rate. So something must be done to compress the data – to make it smaller.

Now if you tried the do–it–yourself scanner, above, using a newspaper picture, you will have noticed that in any one line there are a lot of black dots and considerably fewer white dots. Imagine, if you will, scanning a piece of paper with text on it. In this instance, the situation would be reversed, with considerably more white than black dots. Also, with typescript the number of black–white transitions and white–black transitions (that's where one dot was white and the next was black, or vice versa) would be fewer than the number of white dots. So as the whole lot is connected to a computer let's make the computer do some work. Make the computer count the transitions and the number of black or white dots from one transition to the next. In other words make the computer say (in computer language):

white – 10 dots, black – 3 dots, white – 6 dots .... etc.

instead of

white white white white white white white white white white black black black white white white white white white .... etc.

Even as I've written it, it's shorter. This is called compression, and although real compression systems don't work as I've just described, they follow the same sort of idea.

There's a number of compression algorithms, no one algorithm is perfect for all jobs (otherwise there would be no argument about which was best), each algorithm is best at one form of image or another. The main difference is the degree of compression achieved by different algorithms attempting to compress the same image.

There's also two ways of making the computer do the compression, either by software or by hardware. The end result is the same, but the benefits of one method against the other can be pretty dramatic. Software is very much more flexible than dedicated hardware, say, by being able to switch algorithm for best the results under a given circumstance, and software is always cheaper in quantity. But given the average PC type computer, dedicated hardware wins every time on the speed front. Even the latest super fast PC type computers using the 80386 and 68030 microprocessors can't beat, or even come close to speed of using dedicated hardware. Dedicated hardware is more expensive and once designed, is inflexible by comparison as each added algorithm adds substantially to the cost, but even this inflexibility can be an advantage. It means that all images compressed in hardware will at least have the same type of compression, so that when transmitting images to other systems, means the receiving system only has the one type of compression to contend with. In the main, software compression is left to the big imaging systems which use mini–computers which have the edge in speed, but at a cost.

Now the degree of compression leads to advertising gimmick number 2. How many images on a disk. Now I've seen all sorts of claims, here are a few:
Upto 40,000
Over 60,000
Between 30,000 and 60,000
Thousands
... and so on.
This is really like asking 'How long is a piece of string ?'. The number of images on a disk will depend on how much capacity the disk has and how well the images compress. The first is easily answered, ask how big the disk is ! The second – well – that's a dodgy one. It depends on the document itself and it depends on the particular compression algorithm used. I've compressed a good many images testing our system, and I wouldn't like to be drawn on exactly how well anything will compress. It's even theoretically possible for an image to compress bigger than it started, although I've tried make it happen – I've never actually seen it happen. Let's play safe. The worst I've seen is about 3:1, that's a compression to one third the original size, that would give a rock bottom minimum of 6,000 images on a 2 Gbyte 12" disk (definition – Gbyte is a Giga–byte that's 1 x 10^9 or 1,000,000,000 bytes). On the other hand, the best I've seen is over 100:1 which would give over 200,000 images on the same disk. Who knows. All I'm prepared to say is at an informed guess, the *'Upto 40,000'* is probably near right, given average typescript documentation, and the 'Upto' is an old advertisers ploy, as 'Upto' includes any number less than the number quoted. Ever fallen for that one ? Seen a sale sign which says *"Upto 50% discount on widgets"*; what that means is that they've got one widget with 50% off and rest aren't discounted at all ! Quite legal.

Anyway, that's all for this issue.
Copyright: D. R. Hunt. © 1988.