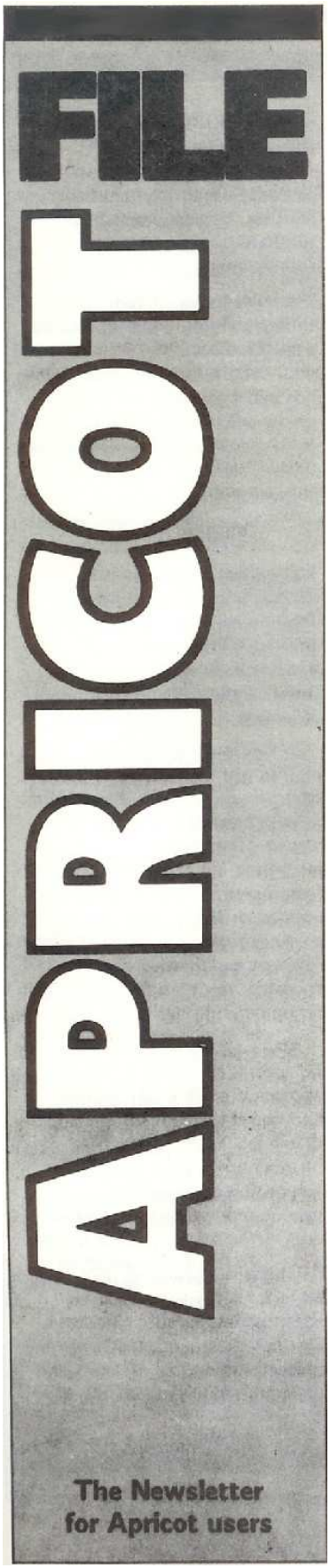
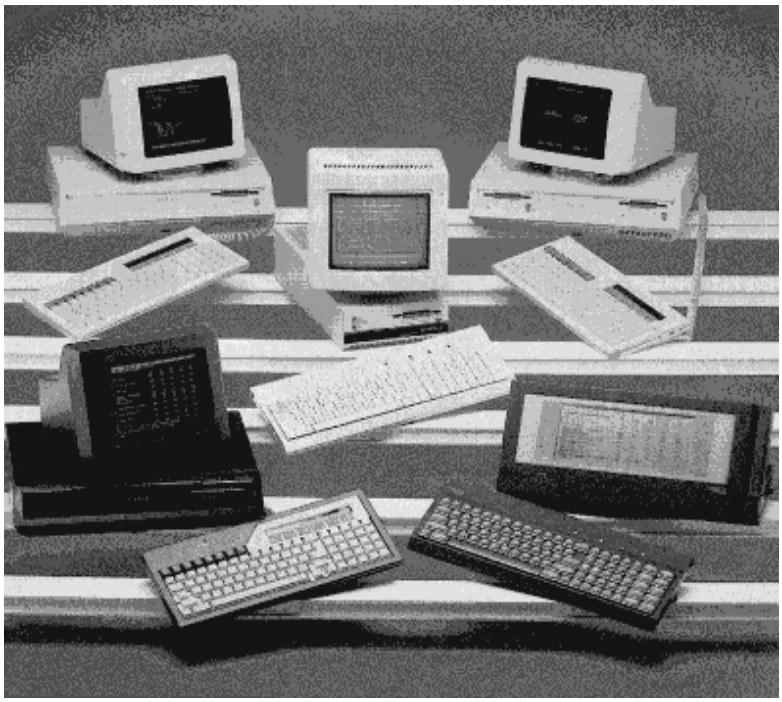


Apricot File 3.3



Scans courtesy of www.actapricot.org

Copyrighted content republished with thanks and permission from:

> **Dennis Jarrett of TP Group**
> **Stuart Bunt**
> **Eddie Bromhead**
> **Mark Whitehorn**

Many additional thanks to Eddie Bromhead for scanning all of Release 3, 4 and extra inserts.

Release 3.3

December 1987

FILE APRICOT

The Newsletter for Apricot users

Apricot news:

Interim results, new products — and the return of voice control!

By Henry Deckhand

Apricot's interim results show the return to profits we anticipated — and, more unexpectedly, a return to dividends as well. The six months to the end of September showed pre-tax profits of £3 million on sales of £37.6 million, with a 0.5p dividend to be paid to shareholders in January. That's the first dividend in two years.

The half-year results and the dividend news didn't do much for the share price, which was somewhat zapped in the crash and at the time of writing is still down at least 20p below its Autumn 1987 high. But the figures do represent a reasonable increase over the year-ago profits of £2.5 million on sales of £32.8 million.

City option

According to Apricot sources, the growth is coming primarily from Apricot Financial Systems Ltd. Like other consistently-profitable parts of the group, this subsidiary doesn't get much publicity; but it does claim to have market leadership in investment management software (the QUASAR suite handles £100 billion worth of funds for some 70 financial institutions) and it reckons to have "the largest installed base of systems in the financial sector".

Certainly it had the most interesting product on view at the recent Computers in the City exhibition, Apricot CITYDESK. It's an information monitoring workstation for analysts, fund managers and traders; it takes input "regardless of the source, location and technology of the incoming data" (data or video feeds), displays it in concurrent windows, and massages it (via the recently-launched and well-reviewed Microsoft EXCEL spreadsheet) for presentation to clients and users.

The turnkey package is a small network with per-station prices from around £5,000 — though something above £8,000 per user is more likely. It includes Windows-386, 'fault-tolerant' software, up to four colour monitors per workstation, and the XENTEL telephone management facility. Xenix and existing DOS applications are of course supported, as are most common network gateways in addition to Apricot's own.

The 'fault-tolerant' bit is Apricot's own 'shadow disk', a potentially helpful but inevitably pricey facility whereby a second hard disk replicates all operations on the primary Winchester.

The system looked very good at the show, fast and flexible. The main alternative to CITYDESK is IBM's not dissimilar RADIX system, based on PS/2. This was announced in September, but at Computers in the City only a prototype was shown — and multitasking Windows, which you might have thought would be *de rigeur* for an information-handling system that takes data from different sources for selective merging, was mentioned only as a planned modification for the future.

CONTINUED ON PAGE 2

In this issue

The Disinformation Column: On more tiny COM files 3
The Workshop: Autodial from WordPerfect 7
Seasonal Cheer: If you own a computer, clap your hands . . 8
Readers' Writes: Letters to the Editor 14
Side-by-side Review: two inkjet printers 17
Pascal Primer — Part Three 21
Old friends: WordPerfect boobs 24
The Workshop: Turn and turn about — Epicycloids 25
Free Small Ads 31

Apricot news

CONTINUED FROM PAGE ONE

Apricot won't ship CITYDESK until next March, but the product looks pretty stable. Quotes from show visitors were excellent — "the best of its kind ... basically Apricot has got it right".

Apricot feels it can compete with RADIX on price as well as facilities: CITYDESK is expected to work out about 30 per cent cheaper than the IBM option. And the proven reputation of Apricot Financial Systems in real-world installation could also help counter the impact of IBM's name.

What's more, the City is a pretty introverted community and it is extraordinarily susceptible to its own publicity. A few high-profile installations for CITYDESK could do wonders for Apricot's share price ...

Xen-i Tower

About the same time, Apricot unveiled a Tower system with its 16MHz 80386 motherboard — and, thanks to the extra space available in a floor-standing box, a lot of expandability. The cabinet holds up to 200MB of hard disk and ten full-length cards.

The Tower also gets 2MB RAM, VGA graphics, the 'shadow disk' software, and monochrome prices for single-Winchester systems starting at £3,999. That buys a 45MB hard disk, so the entry-level pricing isn't too bad — it's not bargain-basement time, but it is competitive with other big-name 386 towers.

New head for Apricot DTP

Distributor Xitan lost its founder and MD Geoff Lynch three months ago following "management differences" with the new owner, Kode. Lynch's replacement, Keith Henley, has now followed: he's taking over Apricot's desktop publishing division. The PageMaker-plus-LaserWriter operation, originally set up under group marketing manager John Leftwich, has scored a couple of coups; but it could do with some fresh energy. Of Henley's appointment, Leftwich commented that the man's experience (in dealing with dealers, mainly) would help Apricot "advance its lead" (oh yeah?) in "industry-standard desktop publishing.

Hear all about it

You might have thought that Apricot's previous forays into voice input would have put the R&D off the business for life. But no: the larger memory, bigger disks and much faster processors that are the norm in today's IBM-compatible world have made the whole thing more feasible again. The software can be cleverer, the vocabularies can be larger, and (because the punters are more used to draining their wallets) the prices can be high enough to pay for a more realistic system.

So let's hear it for Apricot Voice Control, "usable speech technology" that again incorporates the work of Dragon Systems Inc. The software will recognise a thousand words now, with vocabularies of 5,000 and more promised; 'dynamic grammar' selects only meaningful parts of spoken input, and the system can offer up to six alternatives on screen if it doesn't understand the input; a kind of voice macro facility allows the equivalent of multiple keystrokes to be associated with a single spoken command; separate pronunciation files are maintained for each user (individuals have to train the system to recognise their voices, with five samples being offered for each input during the set-up).

The whole thing operates through Windows — it's selected as an icon — and the voice input facility can be added to any package by creating for it a 'grammar' of the command structure indicating the commands to be voice-activated.

Sounds fun, no? For £950 you can buy the basic package — speech card, Shure microphone, software, manuals — for use on a standard IBM-compatibles. Apricot users pay more, £1,150; to run it on a Xen-i you need an XP expansion box because the slots provided won't take standard long IBM-type cards. And for £4,300 you can have a complete development system with a 30MB Xen-386.

New network stations

"The world's first 32-bit network station" is how Apricot describes the new VX386 AWS; basically it's one of the 386-based Apricots without a hard disk, retailing at £1,999 as a 1MB monochrome-display system with a LAN card (£2,699 for an EGA system). The theory, of course, is to provide 386 performance in a network where hard disks are shared; seems to us, though, that paying your £2,000 and adding a third-party hard disk (40MB for £500 or thereabouts) could produce a decent 16MHz 386-based standalone ... ◆

About the newsletter

Our aim is to provide a complete editorial and information service to users of Apricot computers.

Except in the normal course of day-to-day contact, sub sales and the like, we are not affiliated with or supported by Apricot plc or any of its dealers. Nor for that matter are we sponsored by or directly associated with any user group or any other such organisation, though in principle our goals and guiding intentions are very similar to the declared aims of many such bodies.

The Publisher does not necessarily agree with the views expressed by contributors, and assumes no responsibility for errors of reproduction or interpretation in the subject matter of this newsletter or from any results arising therefrom.

APRICOT FILE is a publication of the TP Group — which until we get the name changed is the trading style of Permapoint Limited, registered in England no 1929195.

■■■ ISSN 0268-6066 ■■■

The Disinformation Column

On still more tiny COM files

Cuddly cursors, practiced prints

By David Langford

Perfect footnote

A quickie to start with: there is an intermittent bug in WordPerfect's SPELL.EXE dictionary maintenance program which we can't pin down for long enough to work on seriously.

Connoisseurs of dictionary twiddling will know it as "Writing the m's". What happens is that when a dictionary is rewritten after SPELL.EXE editing (this happens when you leave the Add Words or Delete Words menu and try to return to the main menu), everything goes into a peculiarly irritating loop. Instead of "Writing the a's ... Writing the b's" and so on, SPELL.EXE lurches straight into "Writing the m's" and follows that after an interval with "Writing the y's". There is then a very long pause indeed, during which all the available disk space gets filled with temporary files (even on a hard disk with 15MB free — that was a particularly long pause, as you might imagine). And then ... an error message flashes up, and the dictionary is magically exactly as it was before!

This only happens occasionally. Chris Priest devised a roundabout solution whereby you do your dictionary editing on a disk without much free space (less than the size of a dictionary); when SPELL.EXE quickly complains of lack of space, you then nominate an empty floppy or RAMdisk for the working files. This almost always cures the problem.

We've taken a lot of flak from one stroppy WordPerfect customer who apparently does nothing but reorganise dictionaries all day long. The problem was common enough in our experience to have become a byword amongst Apricot and even IBM users; so, full of fond optimism, we tried dear old Sentinel again. There we benefited from their legendary support, which comprised the information that (a) the problem had never been heard of there, and (b) no one could reproduce it, and (c) it must be a hardware problem anyway, and (d) "We sell 200 copies of WordPerfect every day and no-one ever complained about this before". Ah, what profound expertise one meets from the real professionals.

Curses!

The *Complete Escapes* epic I wrote a while ago (APRICOT FILE Releases 2.9 to 2.11) mentioned the distressing fact that in PC/Xi/Xen BIOS 3.1+ you could no longer set the dinky little 'underline' cursor which is so much more useful in certain word processors like WordPerfect. (The full-size inverse-video cursor creates confusions about the exact start or end of a WordPerfect block, since blocks are marked in inverse video.)

To the rescue came hero hacker Jost Riedel, who's been patching BIOS 3.1.2 yet again to restore lost screen controls and add some new ones. He also pointed out to me the existence of Interrupt EC, which in BIOS 3.1 and after allows even better cursor shape control than was available in the old days.

So if you like customised cursors but don't want to upgrade your operating system yet again, whip out your DEBUG.COM or DEBUG.EXE and join me in this newest epic production — from an idea by Jost, additional dialogue by Langford.

Begin by loading DEBUG and typing 'A' followed by Return to indicate that you want to Assemble a small program. As in past work on tiny COM files, the actual assembler commands only should be entered at the subsequent prompts: DEBUG provides the address numbers at the left. I have omitted segment numbers (chiefly thanks to a suspicion that our dear editor doesn't fancy paying for dozens of useless DEBUG-generated numbers as part of the wordcount) and you can ignore them too. All lines end with a Return, or CR as we obscurantists like to call it.

```
0100 B80C0F     MOV AX,0F0C
0103 BB0000     MOV BX,0000
0106 B90C00     MOV CX,000C
0109 BA0100     MOV DX,0001
010C BF0000     MOV DI,0000
010F CDEC       INT EC
0111 CD20       INT 20
0113
```

When in that final line DEBUG invites you to enter something at address 0113, dismiss it rudely with a Return and the normal DEBUG hyphen prompt will come back. Name the file with the 'N' command — for example, N CURSE.COM <CR>. Set its length by entering R CX <CR> and responding to the colon prompt with '13': the program length is 13 hex bytes (that's 19 in decimal). Press W and Return to write the program to disk.

Copy DEBUG to an empty disk for this whole operation, unless you are insanely confident of your abilities and mine.

(There's a special alternative COM-file creation method for patient readers whose DEBUG.COM happens not to have the A-for-Assemble function. Start with E 100 and enter the hex bytes in the lefthand column, one at a time, separated by spaces: B8 0C 0F BB 00 00 ... and so on. Stop after the final CD 20, hit Return and carry on with N-for-Name as above.)

When run under PC/Xi/Xen BIOS 3.1+, this midget program gives you the normal 'underline' cursor which the documented escape sequence won't (except in Jost's patched BIOS 3.1.2). But this is also a program you can tinker with.

For example, changing the first line to MOV AX,0F0D or MOV 0E will give you a narrower cursor line: the screen lines available for the cursor are numbered 0 to 15, hex 00 to 0F. The first (high) byte in AX sets the bottom of the cursor, the second (low) byte the top. Play around a little to find the most appealing style: hair-thin cursors, little square block cursors, cursors which float in mid-air — take your pick.

If you're one of those weird types who like a flashing cursor, this too is allowed for ... which is just as well, since the tiny program as described above will always turn off flashing. The BX register controls this. If you change the second line to MOV BX,00FF the resulting cursor will flash as well.

Another tip: there is no need to type all that Assembler code in again, just to make a small change. DEBUG masters will simply rush in with the E command; for example, E 104 FF to set flashing via the low byte of BX. (Remember that for historical reasons the low byte comes first in the machine code: compare the hex bytes of the first line with the normal-format hexadecimal number being loaded into AX.)

Those more timorous can simply assemble a replacement line at the relevant address. For example, A 0103 followed by MOV BX,00FF will change that part of the program code without affecting anything else.

Raising a demon

As long promised, here's a very simple example of a memory-resident 'demon' program which is tiny enough to be created with DEBUG and yet serves an occasionally useful purpose. It's a crude printer interception utility.

One problem which can arise with printers concerns whether they've been correctly 'installed'. Is your word processor sending the right escape sequences, or something remotely like them? Or is the printer faulty?

How can you find out? Some printers have a 'test mode' which churns out not text but all the hexadecimal codes which are being sent. Others don't, and it might be illuminating to find what control codes are reaching the printer. The following is a very rough and ready approach; it intercepts the MS-DOS print function and tampers with the results. Specifically, it stops at every Escape character and replaces it with an exclamation mark. Thus the ESC 4 character sequence which starts italics on Epson-compatible printers would not change the print style at all but would print out as '!4'. A messy but sometimes handy diagnostic tool.

Two warnings. First, to keep this within the confines of a single column, I've omitted twiddly bits like facilities for removing this program from your Apricot's memory. Once it's installed, you get rid of it by rebooting.

Second, it will have no effect on printing which doesn't use the standard MS-DOS 'print character' function and instead makes a direct call to the BIOS Control Device. This means it's unhelpful with screen prints via the PRINT key, COPY FILENAME PRN commands, CONTROL-P or >PRN redirection, and with word processors which unlike SuperWriter are smart enough to print via the BIOS.

Here's the program, entered into the DEBUG mini-Assembler exactly as explained above.

```
0100 EB1F       JMP 0121
0102 0000       DW 0
0104 0000       DW 0
0106 80FC05     CMP AH,05
0109 7405       JZ 0110
010B 2E         CS:
010C FF2E0201 JMP FAR [0102]
0110 80FA1B     CMP DL,1B
0113 7405       JZ 011A
0115 2E         CS:
0116 FF2E0201 JMP FAR [0102]
011A B221       MOV DL,21
011C 2E         CS:
011D FF2E0201 JMP FAR [0102]
0121 B82135     MOV AX,3521
0124 CD21       INT 21
0126 81FB0601 CMP BX,0106
012A 741D       JZ 0149
012C 891E0201 MOV [0102],BX
0130 8C060401 MOV [0104],ES
0134 BA0601     MOV DX,0106
0137 B82125     MOV AX,2521
013A CD21       INT 21
013C B06F       MOV AL,6F
013E CD29       INT 29
0140 B06B       MOV AL,6B
0142 CD29       INT 29
0144 BA2101     MOV DX,0121
0147 CD27       INT 27
0149 B007       MOV AL,07
014B CD29       INT 29
014D CD20       INT 20
014F
```

End with <CR>, name the program with N PRINTEST.COM or an alternative of your choice, set its file size with R CX <CR> and 4F <CR> (note how the last DEBUG address prompt reminds you of the file size, plus 100 since COM files start at 100 hex) and save with W <CR>— all exactly as before.

Now run it with PRINTEST <CR>. If your Apricot crashes at once, you did it wrong, har har. If a tiny 'ok' appears, all should be well. Running the program a second time will give a beep to indicate it's already been installed.

Now let's have a step-by-step look at how it works.

Dissecting the innards

I'll use the DEBUG addresses (lefthand column) for reference. This is what the various portions of PRINTEST.COM actually do, in logical rather than memory-address order.

0100:

Jump over a couple of words of storage reserved for data, and over the interrupt-handling routine which begins at 0106, to the installation routine beginning at location 0121.

0121 and next line:

Call MS-DOS function 35 hex to get the current 32-bit 'segment and offset' address of the current handling routine for Interrupt 21 hex. (We load the '35' and '21' into AH and AL in one simultaneous operation.) Interrupt 21 handles all the standard MS-DOS functions. The 16-bit 'segment' part of its address is returned in register ES, and the 16-bit 'offset' in register BX. (Because we use only standard MS-DOS functions and interrupts in this demonstration program, it can be transferred to any other MS-DOS machine — including the dreaded Xen-i range and IBMs. But it's less likely to work with IBM PC-DOS software because the IBM BIOS printer function is much better known, and has been known for much longer, than the Apricot BIOS functions: many more IBM programs will circumvent the slothfulness of MS-DOS and go straight to the BIOS. Of course, similar techniques let you intercept BIOS calls too ...)

0126 and next line:

This is a quick-and-dirty short-cut to check whether the program's already been installed. Our own interrupt handlers are at memory offset 0106 within the current segment, as noted above. If the program is already resident, and Interrupt 21 has been redirected to address our own handler, BX should return 0106. The flaw in this reasoning is that some other resident program might have the same offset address and confusingly return the same value. PRINTEST.COM isn't bright enough to conduct more thorough tests, although these are not only possible but advisable. Assuming the test says our program isn't already there, its execution will skip the next paragraph.

0149:

We jump to this error-handling code if PRINTEST decides it's already been installed. Nothing dramatic is done; we use the MS-DOS 'print character' Interrupt 29 to send a beep (ASCII character 7) to the screen, and then halt execution with Interrupt 20.

012C and next line:

We now save the old 32-bit interrupt address in those two 16-bit slots left blank for the purpose by the DW 0 statements at 0102 and 0104. The offset address precedes the segment address, because we're later going to make use of the whole address. This offset-before-segment convention is another of those little historical idiosyncrasies, like storing the low byte before the high byte of a 16-bit word.

0134 and next two lines:

This is the key operation, in which we change the Interrupt 21 address to point to *our* interrupt handler. Its offset address (0106) is loaded into DX, its segment is taken automatically from the current data segment (same as the program segment in this COM file) so doesn't have to be loaded explicitly. Line 0137 echoes line 0121, but sets up MS-DOS function 25 to reset the Interrupt 21 address to our own handling code at 0106. And the INT instruction does the deed.

013C and three lines following:

Just to indicate that the program has actually done something, Interrupt 29 is used to send a couple of characters to the screen. These are hex codes 6F and 6B, giving the terse message 'ok'. A posher version of the program would print out a whole string (neatly terminated with <CR> and line feed) to explain what's been loaded, what it's supposed to do, and what idiot to blame when it doesn't.

0144 and next line:

Finally we want to 'terminate and stay resident', leaving our interrupt handler in place. The simplest way, though not recommended by fancy manuals (a slightly more twiddly method is now preferred), is to use Interrupt 27 with the DX register containing the address of the first byte following the program we want to remain in memory. Since the interrupt handler ends immediately before the installation program starting on line 0121, this is the value to give. Note the economy; the installation program is run once, and apart from the initial JMP at 0100 it needn't leave anything of itself in memory.

The naughty bits

Now, how does the interrupt handler work? Very simple. When a program like SuperWriter tries to call any MS-DOS function at all, the call is now automatically routed through our own few lines of code beginning at 0106.

0106 and next line:

First we enquire whether the AH register (high bit of AX) is set to '5'. The MS-DOS programmer's reference guide tells us this is a 'print character' function call, the kind we're interested in. If it is set thus, the program skips to line 0110.

010B and next line:

If the program didn't skip and this wasn't a 'print character' call, we need to return control to MS-DOS pretty damn quick. Fortunately, having saved the former offset and segment at 0102 and 0104 in our own program segment, we have all the needed information to send execution straight back using an indirect JMP FAR call. The CS: gives the processor advance warning that the 32-bit address is to be looked for in the current program segment, i.e. ours. The JMP FAR extracts this full address from offset 0102 and vanishes with it into the mists of MS-DOS.

0110 and next line:

Look at the DL register to see if it contains Escape, hex code 1B. If not, we won't interfere: the JMP FAR at 0115 returns control to MS-DOS. Otherwise ...

011A:

We evilly move into DL an exclamation mark, hex code 21, to replace the Escape; and control again returns to the MS-DOS chain of execution, which will not suspect that our inserted routine has pulled a fast one.

011C:

Although this returns control to MS-DOS as already described, we have another option. We could substitute an IRET (return from interrupt) assembler command at this point, completely short-circuiting the MS-DOS print operation and returning to the user program — SuperWriter, say — which asked for the character to be printed in the first place. If an IRET were put here, SuperWriter would labour under the sublime conviction that it had sent an Escape to the printer; and MS-DOS would never know that anything was supposed to have happened.

Note that although adding an IRET to the program after entering it as above will be fine (A 011C followed by IRET), it wouldn't do to substitute a mere IRET for the CS: and JMP FAR lines when entering the program. An IRET converts to one byte of machine code; CS followed by the indirect JMP FAR occupy five. The two lines could be neatly replaced by this sequence:

```
IRET
NOP
NOP
NOP
NOP
```

... where NOP is 'no operation', a harmless one-byte code that does nothing. Of course any old rubbish will do for dummy bytes after IRET, since they'll never be executed — unless you include too many and they overlap into the installation routine, so watch it.

Last words

Phew. I seem to have been typing this description all my life. There are other variations possible to a relatively simple 'demon' program written along these lines. For example, you could give your pals heart attacks by suppressing every lower case 'e' they tried to print (insert IRET as above and change line 0110 to CMP DL,65, since the code for 'e' is 65 hex).

Does your word processor futilely try to send the pound sign to the printer as its 'raw' ASCII code, 9C hex? If you can't get a '£', test for this by making the program look for pound signs *en route* to the printer: simply change line 0110 to CMP DL,9C.

If you starting getting exclamation marks instead of the missing pound symbols, the most likely place for the elusive £ character to be is in the printer font's hash mark (#) position, 23 hex. We can easily substitute one for the other, converting '£' to '#' by changing line 011A to MOV DL,23 (in addition to the line 0110 change just suggested).

Suppose your daisywheel's pound sign can be invoked only by a multi-character escape sequence? Our PRINTEST.COM example is a basis for working out this one. On spotting the pound sign heading for the printer, we could send on the needed characters by a call to the Control Device printer functions previously revealed in APRICOT FILE and then suppress MS-DOS's own single-character print function with IRET.

And then, of course, there are dozens more MS-DOS functions — each with its own special identifying code which can be read in AH at the beginning of the interrupt handler, any and all of which could be intercepted, redirected, modified and generally perverted. The world lies at your feet, Assembler-minded friends.

The possibilities are endless, but my time for writing this column isn't, alas. A Happy New Year to you all. ◆

The Workshop

Autodial

A WordPerfect macro

By Tudor Wynn-Williams

This macro operates a modem as an autodialler for voice calls from within WordPerfect; it was written for a Thorn EMI VX-series modem, but the principle will apply to any autodialling modem.

The macro extracts a telephone number from an address list (which can be of unlimited size — mine currently has about 400 records) and 'prints' the number (plus appropriate codes) on whatever is connected to the serial port; that in fact will be your modem, and this makes the modem dial the number before automatically handing over to the telephone.

The address file should be written with telephone numbers as a separate field, beginning on a new line and ending with a Return — even if as in WordPerfect the next line would contain just CONTROL-R. There must also be some symbol that is used in the macro but never occurs in the addresses; I use the dollar sign.

- Switch screens and type '$' twice, then space back to between them. Insert the whole address file at this point. This provides a working area for manipulating the address file and deleting it; the macro can then be used while word processing without risk of affecting any other text.
- Start a search and insert a Macro Pause so that when the macro is run a string can be entered at this point to find the record wanted. On the Apricot the Macro Pause is entered as CONTROL-STOP followed by two Returns. When the macro is run, the search string entered can be anything that occurs in the record — such as a name, town or even street number.
- Enter another Macro Pause, so that if the wrong record is found the search key can be pressed twice to find the next occurrence. If I wanted 'Barclays Bank' and searched for 'Barc', my macro would stop at 'Barclaycard' but I could press the search key twice to arrive at 'Barclays Bank'.
- Move the record to the beginning of the address file. Do this by searching back for the previous end-of-record mark (CONTROL-E in WordPerfect), marking the start of a block, searching forward to the end of the record (no need to re-enter the CONTROL-E), cutting the block, searching back to the $ mark and retrieving the text.
- Move to the telephone number by searching the appropriate number of times for the end-of-field mark and then moving the cursor to the start of the line with the telephone number. I have some telephone numbers entered twice, with STD codes and without (for local calls), so here I insert another Macro Pause so that I can cursor to the other version of the number if required.
- Change the printer with Select Print Options (Print 4 1) to a serial printer. This is in fact the modem; so the 'printer' will have to be defined to suit the modem, as specified in its manual. For my Thorn EMI VX543 the default setting is seven data bits, even parity, two stop bits, 1200 baud. All eight possible fonts that could be used with this modem 'printer' in WordPerfect should be defined as standard ASCII.
- At the beginning of the line containing the telephone number, insert the three-character sequence ...

  ESCAPE D SPACE

  At the end of the line put a percentage sign. (Escape is entered in WordPerfect by typing the four characters <27> at the Insert Printer command.
- Move to the very beginning of the line — HOME three times, then Cursor Left — and mark the whole line, including the final Return, as a block.
- Lift the telephone receiver and 'print' the block. The sequence

  ESCAPE D SPACE number % Return

  will be sent to the modem, which will dial the number and then connect the telephone to the line.
- Insert a Macro Pause so that you can continue to display the address record during the conversation if you want — it might include your bank account number, for instance.
- Search back for $, cancel block marking, backspace one character, begin block marking, search forward to $, delete the block, change back to the default printer, and switch screens to revert to where you were before running the macro.

The example macro uses an address file called ADDRESS with phone numbers as the third field of each record. The modem is defined as printer 4, the normal printer as printer 1.

When you run the macro, it will pause for a search string to be entered (type the string followed by Return, not the search key) and for confirmation or action at the other Macro Pauses. You can cancel the macro and delete the address file without affecting the selection of default printer at any point until the block is printed. To save time, the record with the telephone number wanted is moved to the start of the address file before printing. Otherwise there would be a delay while WordPerfect checked through the preceding part of the file for formatting instructions. ◆

COPYRIGHT © 1987 TUDOR WYNN-WILLIAMS

AUTODIAL.MAC for Thorn-EMI modems

```
<DELAY 000> <SWITCH> $$ <LEFT> <RETRIEVE> address <Return> <SEARCH> <PAUSE> <SEARCH END> <PAUSE>
<SEARCH LEFT> <MERGE E> <SEARCH END> <BLOCK> <SEARCH> <SEARCH END> <MOVE> 1 <SEARCH LEFT> $
<SEARCH END> <MOVE> 5 <SEARCH> <MERGE R> <SEARCH END> <SEARCH> <SEARCH END> <SEARCH> <SEARCH END>
<RIGHT> <PAUSE> <PRINT> 411400 <PRINT FORMAT> a <27> <Return> <Zero> D <Space> <HOME> <HOME>
<RIGHT> % <HOME> <HOME> <HOME> <LEFT> <BLOCK> <DOWN> <PRINT> y <PAUSE> <SEARCH LEFT> $ <SEARCH
END> <CANCEL> <LEFT> <BLOCK> <SEARCH> <SEARCH END> <DELETE> y <SWITCH> <PRINT> 411100 <SWITCH>
```

If you own a computer, clap your hands

A Christmas Tale

By Alan Solomon

I was fast asleep, dreaming of a sort algorithm that would blow QUICKSORT out of the water, when a small tinkly sound woke me up. I reached for my alarm clock but it didn't seem to stop; so I heaved my eyes open, and bleared at the world. Then I shut them again, because obviously I was still dreaming, and it's always best to do that with your eyes shut. The tinkly sound continued, so I stuck one eye out of the covers and had a look around.

It was still there, tinkling. And I just had to be dreaming; the alternative was too horrible to contemplate. Well, what would you think if you saw something that glowed blue and tinkled, hovered on gossamer dragonfly wings, and looked like a six-inch human female? That's right — it's funny farm time.

But this figment of my imagination fluttered down to land on the pile of APRICOT FILEs I keep for those nights when nothing else will send me to sleep. Well, I've seen stranger sights (you must remind me to tell you some time what I found at the North Pole) and I've learned that nothing is as dangerous as a closed mind, so I said "Hello". "Hello", she tinkled back. "I suppose you're called Tinkerbell", I said.

She folded her wings and looked cross. "Yes, and you're Peter Pan, I suppose." Um. I suppose I deserved that. Start again. "No, I'm Doctor Alan Solomon." "Yes, I know", she said. "that's why I'm here."

I could see that I wasn't doing very well. Suddenly I saw what was wrong. The only small human females I know are my daughters aged 4 and 6, and I was treating this small human female like one of them. I decided I'd better start according her a bit of respect. I got out of bed and put on my dressing gown and slippers. "Can I get you anything to drink?", I offered. She smiled, and said "A drop of honey-and-brandy would be rather nice".

We sat across the kitchen table drinking coffee (me) and honey (her), and she started to explain why she'd come to visit me. Her name, apparently, was Titania, and she was the queen of the little folk. There was trouble in Fairyland, and only I could help. "Tell me what the problem is", I asked, swinging into my role as a professional consultant.

"The fairies aren't working hard enough, and I need some way to motivate them better.", said Titania. I was quite surprised. For a start, I hadn't realised that fairies worked at all — I thought life was just one long shindig in Fairyland. I suppose this is a common effect; life always looks greener on the other side of the fence, and people often think I'm asleep when actually I'm deep in thought.

"But I'm a computer consultant", I said, "What you need is a psychologist". "Yes," she said, "I tried that, but the trouble is, as soon as they see me they start mumbling about alienated realities. And anyway, I thought that we'd computerise everything, and that would mean greater productivity."

Well, I can see where she got that one — she's been reading APRICOT FILE and believing Dave Langford. Computers don't increase productivity, they just foul up your business so badly that you have to spend a fortune on consultants to sort you out. Still, I mustn't grumble, I suppose — it keeps the kids in Care Bears and My Little Ponies.

"So what I've come to ask you, is which computer should I buy?" Oh no, not that one. Women come up to me at parties, strange men phone me in the middle of the night, old ladies accost me in supermarkets, and they all have the same question. Actually, that's an exaggeration — sometimes there's another question, which starts off "My printer doesn't ..."

My problem is, I haven't worked out how to deal with it yet. For a while I took it seriously, and sat down in the middle of Sainsbury's for a four-hour consultation, or spent hours advising some dewy-eyed female that the FIE would not be all the computer she ever needed. But then I got older and wiser, and I started charging people for advice, so at parties I'd come back with "I'm thinking of buying a car — what would you suggest I get", or "I'm thinking of baking a cake — what ingredients should I buy." But people didn't understand the sarcasm, and would start explaining the difference between a Ford and a Toyota, or why I should avoid sultanas. So I'm still looking for a snappy comeback to "Which computer ...", and if anyone knows one, I'll trade it for the line I use when people first hear about my Doctor handle, and start describing their symptoms. It works a treat. All you have to say is "Take off all your clothes".

I explained to Titania that I'd have to understand more about her problem before I could make any recommendations. "I suppose I'll need a LAN", she said and I knew that there was *big* trouble in fairyland. "I never pre-judge", I said pompously, but I was thinking that if I ever saddled these little people with a LAN, I'd never be able to sleep soundly again. "It depends on a lot of things." "So what do you need to know?", she asked.

That's a hard one to answer. When you go into a business to make recommendations, you need to know a lot about it. But what you need to

know, varies from business to business. The truth is, I didn't know what it was that I needed to know. This is usually the case, but if you're looking at a candle-stick-maker, at least you have a vague idea what he does before you start. Hands up anyone who knows what fairies do for a living? I needed to go "on-site" as we say.

"I need to visit your premises", I explained. "No problem", said Titania, and waved a black rod with a rusty star on the end. "Wait a min ..." I began, but before I could say it, I found myself standing in the middle of a field of shoulder-high toadstools. Titania seemed to have grown, and was now nearly as tall as me — I looked again, and realised that she hadn't grown — I'd shrunk. I had another look; don't let anyone tell you that fairies are sexless — this one was definitely female.

John Darling started it, and Arthur Dent brought it off beautifully. But take it from me, a dressing gown is just not a practical garment, and I soon began to feel the chill. My slippers weren't waterproof, and as I squelched after Titania, I began to wish I'd worn something more practical. Still, it could have been worse — sometimes all my pajamas are in the wash.

She ducked into a particularly large and poisonous-looking toadstool, and I followed her. We went down a spiral staircase for ages, and eventually emerged in a large room that looked like a gigantic British Telecom switchboard. There were complicated plug-and-socket consoles all round the walls, and fairies were running back and forth from the console operators to a complicated arrangement of wires and pulleys in the middle of the room. The wires ran overhead about six inches from the ceiling, and disappeared through doors. And suddenly, I remembered Harrods twenty years ago, as I watched the cannisters flying up and down the wires, presumably containing sales orders and invoices.

"What's going on here?" I asked. "This is the Wishing Well", she answered. Fairies were rushing around in all directions, far too busy to talk to an interloper. Titania beckoned imperiously, and a fawn ran up, panting. "This is Doctor Solomon — he's here to help us improve our productivity. Show him around, and give him any explanation he wants." Titania turned to me. "I have to go now — when you're ready, come and talk to me about your recommendations." "Er,", I hesitated. "We haven't agreed on, er, um..." It seemed wrong to talk about cold cash in Fairyland, but we consultants have to think about such things, otherwise the supply of Care Bears and My Little Ponies will dry up. "Payment will be as much gold as you can carry", she said, and momentarily Greed replaced Lust as my number one problem. I reckoned I could carry at least 80 pounds, and at $600 per ounce, that meant a lot of cat food and Vim.

I watched her move regally through the crowds, and with a sigh, turned to my new guide. "What's your name, then?", I asked. "Puck", he said. "Of Pook's Hill". With all these lovely fairies around, I have to get saddled with an ugly, weird-looking fawn called Puck. Oh well, I wasn't here to enjoy myself. "OK, Puck, what's going on here."

He started to explain. "The wishes come in on the switchboards," he said. "The what?" I interrupted. "Switchboards", he said. "No, before that", I said. "Before what?" he asked. Just my luck. Not only an ugly Puck, but also a thick Puck. "Start again", I said.

He gave me a look like I was stupid, and I knew that friend Puck and I were not going to hit it off. "The wishes come in ..." he said slowly and distinctly. "What wishes? What do you mean?" I cut in. "The wishes, the wishes", he said, as if that explained it.

You often get this failure in communication. Computer people use computer jargon to people who don't understand it, but accountants use accounting jargon, and solicitors give words very non-standard meanings, and here was this flatulent Puck who thought that anyone who didn't understand his jargon had to be an imbecile. "Show me one of these wishes", I asked, and he went and got one for me.

What he actually brought back, was an order form, just like you'd see in any organisation. Written on the form was the customer's name and address, an order number, and a space for the order itself. But I've never seen an order like it. It said, "I wish this rain would stop".

I stared at the form, wondering what goods or services the customer must be after. I run a mail order company myself, and I can tell you we get some pretty weird requests. But as I looked at more orders, I could see that they were all like that. "I wish he'd hurry up", and "I wish I hadn't eaten that". Then I realised what they were — they were wishes.

Okay, okay, I know I should have worked it out sooner. Don't forget that when I go into a company they immediately start bombarding me with the shorthand expressions that they use, and I always have to get them to slow down and explain. This consulting business seems to consist entirely of stepping into buckets of paste — the trick is to make sure you're wearing wellies.

Anyway, now I realised what was going on. The fairies were in the wish-granting business, and now that I knew that, a bigger mystery immediately presented itself. I've never seen a business that ran out of altruism, so since no-one was paying for the wishes, why were they doing it? You might think that this was none of my business, but it's my experience that you can't really help a client until you really, fully understand the business, and if there are any major mysteries, you haven't understood it yet.

I asked the fawn Puck. He said that fairies grant wishes because that's what fairies do, but I knew he was wrong. You often get that sort of answer — I hear things like "We're in business to provide a service". Nonsense. You're in

business to make a profit, and providing a service is a means to that end. But lose sight of the end, and you wind up losing your business. I decided I'd ask Titania, when I next saw her.

I took a closer look at the switchboards. The fairies manning (I suppose that should be fairying) them were being deluged with calls, and were scribbling the orders onto pads as fast as they could.

I could see that they were simply missing a lot of the calls, because they couldn't write the name, address, birthday and wish fast enough. And then there were a whole bunch of Wishrunner fairies who did nothing but grab the wishes from the switchboard Wishgetters and run with them to the dispatching contraption of wires and pulleys. They had to put the wish into a cylinder, put the cylinder on the correct wire, and pull the handle that worked the spring, which sent the cylinder flying along the wire.

I got faithful Puck to explain it to me. "We work by countries; the wishes are going to the country specialists". This made sense — after all, a wish for good weather in the Sahara has a completely different meaning from a wish for good weather in England. We followed one of the wires down a corridor and into the Room of the Spanish Wishbringers. Inside, hundreds of fairies were frantically waving their wands to and fro, in a desperate attempt to keep up with the flood of orders. One fairy dashed out carrying a cannister — "Probably the wrong country", said Puck. He explained that a lot of wishes got misdirected, some because in the frantic rush of orders, they were sent to the wrong room, but mostly because these days, a lot of people were not in their home countries, and Fairyland's information was rather out-of-date, as there was no time to spare to get it current.

I was horrified. The whole system was a total shambles. I have never seen a business that was so badly organised in my life. They used manual systems where a computer was obviously better — they didn't even know where their customers lived! Judging from the Harrods contraption, the last time they'd equipped the business was fifty years ago. It isn't often that you see an administration system so ripe for change, and it isn't often that it is obvious what was needed. But this one stuck out like a sore thumb. "You need a LAN", I said. I never thought I'd say that and mean it.

That evening, I was summoned into the royal banqueting hall, to dine with the Queen. Everyone was dressed magnificently in silk and brocade, adorned with flowers and bedecked with jewels. I began to wish that my dressing gown wasn't quite so old, and my slippers weren't quite so disreputable. Titania was the loveliest of all, wearing a gossamer creation in midnight blue, and I can tell you that fairies are quite definitely female, and I can't think how the slang term got started. She sat me down next to her, and I tried to think beautiful thoughts as she leaned towards me and asked me about the Wishing Well.

"All your problems will be solved by a LAN", I said, conscious of the sarcastic remark that I usually made when I heard this line. "We replace the wires-and-cylinders with a centralised file server, and each switchboard Wishgetter has a PC at her desk. When the wish comes in, the PC searches the database for the wisher's name, address and birthday, and puts it up on the screen. The Wishgetter accepts or amends it, keys in the wish, and then goes on to the next one. Each Wishgetter will be able to handle at least five times as many wishes, because of the time saved by not having to write down the wisher's details." I looked at her — she still seemed to be listening, so I went on.

"The computer running the file server can then route the wish to the appropriate Wishbringer Room, automatically taking care of details like the wisher being temporarily abroad. In the Room, each Wishbringer has a PC that is linked to a centralised Wish Server. The wish is queued on the Wish Server machine, which holds the wish until there is a free machine to display it on. As soon as possible, the wish is displayed, the Wishbringer executes it and presses the Enter key, or refuses it and presses the ESC key; either way, that signals to the Wish Server that the wish has been processed, and can now be deleted."

"It sounds complicated", said Titania. "No, not at all", I said. "A standard LAN will handle it, using dBASE III Plus. And the fairies that are released from Wishrunner duty can become productive Wishbringers, we gain both ways — the volume of work needed to process each wish is reduced, and there are more fairies executing wishes." "Brilliant", said Titania, "do it". "What?", I said. "Yes, do it," she said.

I'm not used to that. Usually you prepare a written proposal in multuplicate, committees get set up to examine your proposal, it takes three years, everybody has to have their say and add their three ha'pence, the plan gets twisted and watered down, and by the time you see the end result, you're horrified to think that you had triggered it off. I wasn't used to "Yes, do it". I guess I haven't worked for Royalty before.

Of course, it wasn't that simple (it never is). Each Wishgetter and each Wishbringer had to have a PC, and had to be linked into the LAN, and all this had to be planned, organised and installed. There are six billion people in the world, and they each make an average of ten wishes per day. Fortunately, most of them can be refused, as they conflict with someone else's wish, and in some cases a load of wishes can be batched up and executed as a block wish (especially the weather wishes). But those 60 billion wishes still have to be dealt with, one way or another. I calculated that Wishgetters and Wishbringers would be able to deal with 2000 wishes per day, so we would need about 30 million of each. But there were only about 3 million fairies altogether, so we would still be underperforming by a factor of 20. Still, with

the current system, only about half the fairies were working productively, and could handle a maximum of 400 wishes per day, so my system would be ten times better than the old one.

We had to buy the 3 million PCs. That's a story in itself — there are not many organisations that can supply that kind of volume. In fact there's only one firm that has that kind of manufacturing power. A lot of people wondered why IBM's profits stumbled so badly last year, and most analysts assumed it was the impact of the cheap clones on IBM's volume. In fact, volume was way, way up, as you now know; the reason why their profits were so bad was that we paid them in fairy gold — it's a hard life in the hardware business.

It wasn't easy setting up the LAN, either. I ran head-on into the limitation of 64 machines on a single LAN, and I needed to hang 3,000,000 on it. But there's more than one way to skin a cat, and some of them give you quite a lot of usable fur. I LANed the machines together in groups of 64, and then hooked each file server into a SuperLAN, so that I had network of 4096 machines. I connected the SuperLAN fileservers together into a SuperDuperLAN of 262,144 computers. Even this SuperDuperLAN wasn't enough, of course, and I needed to network six of them together into a machine I called the Master Server. That gave me the necessary 1.5 million Wishgetters workstations; in the Wishbringer rooms, it wasn't necessary to go to such lengths, and I could get away with SuperLANs in most cases, and I only needed a SuperDuperLAN for the Chinese and Indian Wishbringer Rooms.

You can imagine the problems we had with the Wishrunners — they didn't like the idea of retraining as Wishgetters or Wishbringers at all, and Titania had to use all her persuasive powers to avert a strike. And the Wishgetters said that they didn't like the new system, and kept trying to sabotage it by feeding in wrong information. Only the Wishbringers were pleased with the new arrangements, as it made their jobs more important.

I use the old wire runs to string my cables, and I used good old twisted pair — none of this newfangled optical stuff for me. What I say is, if you can't put a voltmeter on it, it isn't really trustworthy. And anyway, the fairies all glowed so much, an optical link wouldn't have worked. I needed a lot of hard disk to store all that data, but that was easy — I just put a couple of the CORE 260 megs in each machine, and that gave me a total of 780 terabytes online (a terabyte is a 1,000,000,000,000 bytes). All that hard disk needed backup, and we decided to use the traditional method, as it is cheap, fast and nearly 100 per cent reliable. But we improved on it, since the database was so important; instead of just crossing our fingers and wishing as usual, we had a leprechaun standing by at all times to bring good luck.

I suppose you're thinking that anything that complicated couldn't possibly have worked? Well you're wrong; it worked perfectly, first time, and ran as smoothly as silk, and never broke down. LANs *do* work, in fairyland. Unfortunately, it didn't seem to make the situation any better.

I was summoned into Titania's office, and she was not pleased, not pleased at all. "It's as bad as ever", she said, and I had to agree. The Wishgetters and the Wishbringers were as overstretched as ever, but they were still dealing with only 3 billion wishes per day, out of the 60 billion total. To put it another way, we were still losing 95 per cent of the wishes, and that was not very good. I had an idea on how to handle that, but first I had to restore client confidence.

With a theatrical flourish, I produced a bundle of paper. The Master Server maintained statistics on the wish flow, and I showed Titania that we were doing much better than under the old system. True, we were only handling 5 per cent of the flow, but in the old days, 0.5 per cent was the average score. "I agree a score of only 5 per cent is inadequate, but I've already improved performance ten fold, and if you give me a chance, I'll do even better." "How?", demanded the Queen. So I explained to her.

Since the Wishgetters were now only needed to key in the wish, we could aim at automating this process; then we could retrain them as Wishbringers, and capacity would be much greater. "How?", she asked. "Instead of writing down the wish, we sample the sound, digitise it, and store it in a compressed form. Then we transmit the wish over the network to the appropriate Wishbringer room, where the digitised sound can be reconstituted for execution by the Wishbringing fairy."

I could tell that she was impressed. I embroidered the idea a bit more, as clients always like things to sound complicated, otherwise they feel that they're not getting their money's worth, but I was going to use off-the-shelf hardware and software, and I knew it would be quite straightforward.

We installed the digitisers, and retrained the Wishgetters. The Wishing Well was now strangely silent for the first time in centuries; the Wishrunners had long since been redeployed as Wishbringers, and now the Wishgetters had gone. Puck and I strolled through the Wishing Well, and I saw a tear roll down his cheek as he remembered how it had been for all those thousands of years, and how it would never be the same again. "Cheer up, Puck old fellow", I said. "You'll be a Wishbringer, now." "I failed my registered Wishbringer exams", he said, sadly. Only fully qualified personnel were allowed to execute wishes, for obvious reasons. Puck wasn't up to scratch, and now faced redundancy. I thought about all the other fawns and fairies that were in the same boat, and for the umpteenth time wondered about this "progress" that computers were supposed to bring. But then I remembered an old saying. A bulldozer replaces a dozen men with spades, but don't sympathise too much with the spademen;

remember that each of them replaced a hundred men with sharp sticks.

My tour of the new facilities took me into the Spanish Wishbringer room, and here I saw immediately that I had overlooked something important. Or rather, I heard that I had overlistened something important. It was pandemonium. There were too many machines synthesising too much speech in too small a space. Each Wishbringer had turned up the sound in an effort to hear better, and the result was a stentorian cacophony. I couldn't even hear myself think in this din, so I hurried out. If you ever wondered what happened to last year's cotton crop, now you know — cotton wool is the best sound absorber I know, and it worked a treat.

Between doubling the Wishbringer staff, and improving their performance by using voice, we had achieved a 200 per cent improvement in wish execution. That was really excellent, but it meant that we were still losing about 85 per cent of the business, as Titania kept pointing out. I hate it when clients are never satisfied with what you give them — I'd put in two technically superb systems, and still she wasn't satisfied. She kept rabbitting on about those lost wishes, until I told her I'd have another think.

My first idea went down like a lead balloon. I suggested we route all wishes via Wishdealers, so that anyone wanting a wish would have to trot along to his local friendly dealer, and the dealer would then forward the wish to us. I explained that this would staunch the flood of wishes down to a trickle, but Titania was horrified, and said that was the last thing she wanted. "We have to keep the wishes flowing", she said. Something was nagging at the back of my mind — something I was supposed to be doing or saying, but I couldn't quite remember what it was, and since I don't carry my Filofax in my dressing gown, I couldn't do anything about it.

My second idea I didn't even bother to mention. Only a tiny percentage of the wishers had modems, and the thought of all those wishes trying to get down a BT line made me fall about laughing. And anyway, there's a six month waiting list for BT lines into Fairyland, so the bulletin board idea was a non-starter.

But my third idea was nothing short of brilliant — we would automate the last remaining part of the process. I used Turbo Prolog to write an AI system that would identify, classify and sort the wishes; the hardest part was to go from the digitised sound to an understanding of the wish. Full speech recognition is way beyond the state of the art, but wishes are by their nature very simple; most of them were things like "I wish I had more money" or "I wish he'd go away". So the usual problem of fitting the speech into context didn't apply, and I got up to a 90 per cent recognition rate. The 10 per cent that my AI system couldn't handle would be passed on to the registered Wishbringers, and that 10 per cent would be within their capacity — at last we stood a chance of handling all the wishes.

I used fairly simple robot arms, controlled from the PCs that were already installed in the Wishbringer rooms. The arms needed freedom of movement in all three axes, in order to make the complex passes and gestures with the wand that would execute the wish, but it didn't need any kind of feedback, and it is the feedback robots that cost a fortune. I was very pleased with this system — now all phases of wish processing were automated, and the fairies would be able to cope. Improvements in the speech recognition software would lead to a better hit rate, and therefore to a lower percentage being passed over for manual processing, so there was even scope in the system for growth in the flow of wishes. Titania approved the design, and Puck and I organised the installation. Eventually, it was all complete, and ready to be started up, so we arranged for a Grand Opening Ceremony, at which Titania would throw a switch that would load the wish recognition systems into the PCs, and power up the robot arms.

The great day dawned. All the fairies were really excited — at last they would be liberated from the 24 hour days of drudgery that Wishbringing had become. We used the old Wishing Well for the ceremony, as it was the only room that was big enough to contain 3 million fairies (everyone wanted to come). Great rows of trestle tables groaned under the loads of food and drink; the elves had partitioned off the Wishgetter PCs so that they were out of sight, and decorated the room so that it had a really festive look. There was a big brass band playing stirring music, and everybody was dressed up to the nines. Even I made an effort; I pinned a daisy to the lapel of my dressing gown and spruced up my slippers a bit.

Titania made a speech. You know the sort of thing, and I'm sure you've heard it a hundred times before, so I won't repeat it here. It was all about brave new eras, and opening vistas, and paths to the future, and all pulling together, and how everyone was a vital cog in the great machine that was Fairyland ... I got quietly sozzled on mead, and wondered what I'd inflicted on these defenceless little folk. Eventually, Titania stopped mixing metaphors and the band played a fanfare. There was complete silence, a roll of drums, and Titania threw the golden switch. Nothing happened.

In this sort of situation, you set up a big flashy throw-bar switch, and run two obvious wires from it, but if you know what's good for you, it isn't actually connected to anything. There are so many things that can go wrong, like some fool throws the switch too soon, or the wire breaks, or something, and then you have to go up on the podium in front of six million eyes, and start fiddling around with screwdrivers. No, the golden switch was a dummy, and that cretin Puck had fallen asleep at the real switch. I struggled through the crowd to where he was snoring, woke him up with a

crisp clout on his left ear, and while he was wondering who'd chucked the thunderbolt, I pulled the real switch. Lots of impressive whirring, clicking and buzzing sounds started coming from the Wishgetter PCs; all perfectly fake, but there had to be something to show the assembled multitudes that something had happened. In the Wishbringer rooms, I knew that wands were weaving their complex patterns, and all over the world, wishes that formerly hadn't a chance were now coming true.

The party went on all night, and into the next day. I hadn't realised that fairies could be such swingers, but wow, when they let their hair down, they really let it down. And they really had something to celebrate — Titania's speech may have been the usual boring management up-and-at-'em gung-ho speech, but I really had liberated Fairyland, and how many people can say that?

I was woken up next day by a grim-faced Titania shaking my shoulder. Fairies were running around in all directions. There were Wishgetter fairies fairying the consoles, writing wishes on scraps of paper and passing them to Wishrunners, who were frantically racing out of the Wishing Well and back again. "You're fired", she said. "What's wrong?" I said. She showed me, and I blanched. Something had gone wrong in the Wishbringer room, and the robot arms had been granting the wrong wishes to the wrong people. Can you imagine getting someone else's wish? Completely inappropriate people had been falling in love with each other, igloos were melting in the hot sun, thin people were getting thinner and fat people fatter, and the increase in the money supply was threatening to cause a world financial crisis.

"We've ripped out your computers, and we're back to the old manual systems", she said. And they didn't have their Harrods machine, so the Wishrunners were having to carry the wishes for miles to the Wishbringers, and they'd binned the order forms, so they were using any scraps of paper they could find -the chaos was total, and they were servicing maybe a tenth as many wishes as they had before I'd got there. I felt awful. It isn't often that I fail as badly as that. And then the thing that had been nagging at the back of my brain surfaced.

"Titania, just answer me one question". She looked at me the way a Frenchman looks at a plate of fish-and-chips from a Happy Eater. "Why do fairies grant wishes?"

Her expression changed; now she looked at me the way you look at someone who has just formatted a hard disk by accident. "Because if we didn't, people would stop believing in fairies". "Why should you care whether people believe in you or not?" "Because every time someone stops believing in fairies, a fairy dies. And it's got so bad now, that only a few, very young children, in very backward countries still believe in fairies. In the old days, there were billions of us, but now we've almost died out."

I should have discovered all this before, I know. The only excuse I can offer is that the technical problems were so interesting, I had become completely blinded to the primary rule of consulting — find out what the real problem is. "But surely the fairy birth rate must compensate for the death rate — can't you just have more baby fairies?"

Titania explained the facts of life to me. Fairies don't have babies. They don't have some essential piece of equipment — she didn't go into details. The only way that you get new fairies, is when someone claps their hands, and says out loud "I *do* believe in fairies", and means it. So you get a few each Christmas, when the pantomimes run, but hardly any during the rest of the year. And the fairies were slowly becoming extinct.

Now I could see that I'd been tackling the wrong problem all along. The real problem wasn't to make more wishes come true, it was to protect the population of Fairyland against suspension of belief. Granting more wishes wouldn't actually help, in fact, as people would not realise where their good fortune was coming from, and put it down to the deity in whatever religion they followed, or to a rabbit's foot, or good planning and judgment by themselves, or a four-leaved clover.

I explained this to Titania, and she nodded glumly, "What else can we do, though?" she asked. "Marketing", I said, "Targetted marketing. You've got to sell yourselves, and you've got to do it to the most receptive market segments". "You mean the young children", she said. "No", I said.

Young children today are *very* sophisticated. They're into Care Bears, and unicorns, He-man, and My Little Pony. Fairies are very old hat, and no self-respecting child would believe in anything as unlikely as a fairy. My two kids explain it like this. "Care Bears aren't real, Daddy, they're just pretend". And if the mighty International Care Bear Manufacturing And Marketing Company Incorporated can't dent their armour, what chance do fairies stand? "So who should we target?", asked Titania.

I thought about gamblers, and their fond belief in luck. I thought about economists, and the way they believe sixteen impossible theories at once. I thought about politicians, and the way they can instantly believe anything that it is expedient to believe. And then it struck me — I didn't have to look any further than my own nose. Anyone who has bought a computer is by definition gullible. Anyone who owns a computer has swallowed so many unlikely stories that they would have no trouble with another one. Anyone who is a habitual user of a computer is a living demonstration of the triumph of faith over adversity. "We go for the computer users", I replied.

So here's what I want you all to do. Stand up now, and look the world firmly in the eyes. Clap your hands once, hard. Put your hand on your heart, and repeat after me, clearly and loudly — "I *do* believe in fairies". ◆

Readers' Writes

Your chance to have a say

DFR updates

Thanks for publishing my letter in APRICOT FILE Release 3.1. On DEALS FOR READERS topics, I detect that there seems to be a continual upgrading of the contents of the disks. Well done, but what about those of us with an old version? How about an upgrade option on return of the originals?

The DEBUG tutorial by Simon Potter is wonderful — keep up the good work!

Andrew Glennie
Christchurch, New Zealand

Indeed, the disks are updated as and when an improved version of a particular program appears. Usually the fixes are minor and wouldn't warrant a replacement; but when it does make sense to upgrade, we usually signal that in the newsletter with a free update offer.

The next Apricot DEALS FOR READERS catalogue will include a revised two-disk Index set for £6, with date-stamping on the files so that you can check their currency. This will also be available in paper form for £2.

And yes, we will gladly update any public domain disk (those with a DEALS FOR READERS number starting 'AP') if you care to send us the original disk and a stamped self-addressed envelope.

Andrew Glennie's letter in October also alerted us to the existence of a product called SoftPC which emulates an IBM PC on any other Intel-based machine — including the Apricot. We tracked the vendor down as Insignia Solutions of High Wycombe, and a subscriber rang us with a phone number. We must have copied it down incorrectly; the number didn't work, and directory enquiries don't know the company. Can anyone help?

As for the DEBUG Tutorial, this has skipped an instalment because Simon Potter has been writing a book. He apologises, and so do we, but it should reappear in the New Year.

To PC or not to PC

I am very interested in the upgrade from the Apricot PC to the 286 as discussed in APRICOT FILE. But while I like the idea of gaining an AT compatible machine for less than £1,000, I do not like to lose some of my existing software, files, etc. I welcome the access to the tens of thousand software program which are available as IBM-compatible; but I also realise that there are few programs which use the 80286 microprocessor to the full.

As you can see, I have my reservations about the upgrade. And I find it ironic that Apricot users are being asked to consider an upgrade to an AT-compatible machine which will mainly run IBM PC programs, while the bottom levels of IBM's new range may *not* run them — *and* has a specification more like the old Apricot than the 'industry-standard' IBM PC. I suppose that is what is called progress. It seems like one step forward, two steps back ... What do you think?

D Blenkinsop
Tananger, Norway

No, it's called the computer business. Which, more than most industries, is driven by technology and the marketing weight of *de facto* standards than by what the users want ...

Content and cost

Thank you for the subscription renewal reminder — a cheque is enclosed. Maybe £50 is a little bit steep; certainly I tossed up whether to renew or not. This is not to say that I don't find it useful, but you need to move on as we upgrade. Certainly I am now thinking seriously about either buying a 286/386 machine or upgrading the current Xi10 (or even both). It is getting to the point where further investment in the old Apricot is becoming a bit doubtful.

No doubt you will be moving to address the new constituency of compatible Apricots as well as drawing sustenance from the old — I am particularly interested for instance in the compatibility between my existing 760K floppies and the newer 1.44MB 3½in disks (anybody who goes to a 5¼in on a new machine needs new distance glasses). Nevertheless, until everybody goes 3½in there are going to be transitional problems. Equally, one is worried about SuperWriter files (I am pretty well sold on WordPerfect on a new machine).

Still, there seems some life in the old Apricot dog yet — long may you continue.

Back to the £50. Can't you get some advertising revenue to help out?

Derek H Broome
Northampton

We would really rather not take advertising; as it is, APRICOT FILE operates on a highly marginal basis — no-one here is getting rich from the newsletter — and advertising paradoxically puts up the running costs (you have to invest in sales commission and carry a promotional overhead).

The other problem with advertising is that it requires a 'universe' of potential advertisers; and frankly there aren't enough Apricot-specific products and services to make it worth our while. We also have a relatively

sophisticated readership, relative that is to the average computer magazine reader; you're unlikely to buy something straight from a printed ad, as the advertiser would prefer. So the advertisers will place their business in publications that *do* have more of that kind of reader.

On the subject of moving APRICOT FILE towards the IBM-compatible world, we will of course reflect the increasing need for old-style Apricots to co-exist with IBM-compatibles. But we are again disinclined to change too much; there's no shortage of IBM-related publications, and Apricot's product offerings aren't sufficiently differentiated to justify a specific newsletter. What's more, we don't intend to break faith with those who regard APRICOT FILE as their last remaining lifeline.

Experience

With hindsight it was a mistake to buy the Apricot; but at the time (September 1984) it seemed to be technically the best buy, certainly when fitted with an 8087. The fact that the CPU is an 8086 seemed to indicate that the IBM-incompatibility was only superficial — and that excellent drawing program AutoCAD was there to prove it. This did most of what I wanted, with one minor shortfall expected to be fixed in an update.

We all know now how differently almost everything has transpired.

Having personally bought the machine in the UK, I discovered too late that it had a duff operating system and one duff disk drive. There was no guarantee, of course; but the Dutch importer did update the operating system so that it would at least handle double-sided disks! The drive (which had visible internal mechanical damage) was fixed locally at cost.

The next mistake on my part was to order — in advance and unseen — the Apricot colour graphics card and monitor. The Beta-test software simply does not work, and the screen is far too small for engineering applications. Nor does it respond to the AutoCAD driver for colour operation.

Perhaps someone with the necessary knowhow will make me a decent offer for the virtually unused monitor, card and disk set? The problems are probably trivial ... provided you have a manual!

In the meantime, engineering needs are pointing me in the direction of the 80286. I need to run programs for control system simulation and electronic circuit analysis that are unavailable for Apricot PC. I've also played with AutoCAD 2.5 on an IBM AT with an 80287 co-processor, and I am hooked. Having learned my lesson by now I will *not* use the new Apricot 80286 motherboard.

My Apricot PC still has some of its write-off to earn. It makes a decent word processor (ignoring SuperWriter's iniquities). But I'm not going to invest in new software (even Ansible's) unless it will port to a new machine.

Tom Holmes
Vessem, Netherlands

IBM emulators

In general, I have found APRICOT FILE quite helpful in getting to grips with the operating system in a practical way — having a copy of Van Wolverton's book is not sufficient in itself. The articles about standard packages are also of interest, because they highlight the shortcomings and advantages of them.

I do find articles written by companies with an interest in selling their own products rather a waste of my time. Indeed, I find that most of the articles are skimming over the surface. Notable exceptions to this are the articles by Messrs Bromhead, Langford and Doyle; that is not to say that others, at their brighter moments, are any the less notable.

What is needed, in my opinion, are articles about applications to real-life problems. But I suspect that as in my case, programs written over many years on different machines, need to be translated to be of any use on the Xi, and it takes the most valuable asset of all — time — of which there is none to spare. I suppose people like Mr Bromhead are the only ones who have plenty of spare time, and he is not in industry: what a pity.

I have just purchased Microsoft's MacroAssembler version 4.0, so a few helpful articles would be appreciated.

Keep the good work going. We may be thin on the ground, but the font alone keeps me with the Xi.

M P Thompson
Dartford

Greek WP

A reader recently requested information on a word processing package that is capable of doing Greek letters; I would also like to suggest that the writer look into WordCraft. I managed to produce a bilingual Thai/English version into this. The whole keyboard is switched using the CONTROL and MENU keys together, and shifted back again using Shift-MENU (this can be changed by the user if necessary).

The earlier versions had normal letters and then italics for the super shift. More recent WordCrafts now have IBM graphics characters in the upper range of ASCII codes, in line with the general trend in dot matrix printers; all of that rubbish can be taken out (as I have done) and replaced by characters using FONT and KEYEDIT. There is a bit of programming necessary with a boot-up file in WordCraft, but this is easily edited by the user. The necessary instructions used to come on disk in the earlier versions but now it costs and extra £150 (though you do get a lot of extra utilities as well). Printing though has to be done by downloading program. Hope this is of some use ...

R D T Griffiths
West Byfleet

Review

Quiet printing in your briefcase?

Two inkjet printers compared: Diconix vs Hewlett-Packard

By Terence Green

You know that rattling, screeching, vibrating dot matrix printer on your desk — you remember, the one with earthquake-emulation? What would you say to someone who offered you a neat, compact, practically silent printer in exchange for it?

You'd probably say "Yes, please". And who wouldn't?

The printer on offer will most likely be one of the 'inkjet' genre. Quieter than the more common impact dot matrix printers, inkjets can be more versatile as well — and they can produce both colour output and transparencies for overhead projectors.

So inkjets are well worth considering if you're looking for a new printer. There's another plus — a small but perfectly-formed specimen from the Kodak subsidiary Diconix is now being marketed as the answer to the hardcopy needs of portable computers.

In order to compare the Diconix to a more established inkjet printer the Editor tossed a Hewlett-Packard ThinkJet into my carrier bag as I left his office. The Thinkjet, the smallest inkjet printer from HP, has been around since 1984 without setting the world alight. The Diconix was launched but a few months ago.

Let us spray

Inkjet printing *per se* is by no means a new technology, prototypes having been shown 20 years ago. So it has taken some time for inkjet printing to catch on. Hitherto the disadvantages have included nozzle clogging, spatter and poor character definition, speed limitations, and the cost of the ink (it requires a special formulation, otherwise it dries up after two or three passes).

But most objections are being met in current inkjet products. Inkjet printing has also received something of a boost thanks to the requirement for low noise levels (in offices and with travelling portables) plus the reasonable print quality and the use of plain paper. Most of the other more or less silent printer technologies demand special paper; apart from the extra cost, such stationery has not generally proved acceptable (or in some cases sufficiently durable) for correspondence and documentation.

The inkjet printing technology produces a dot matrix impression on the paper. There's a reservoir containing special ink under pressure; very small droplets are ejected from one or more nozzles on a moving head. As they squirt out, they pass through electrically-charged fields which attract the droplet one way or the other and so steer the drop into the correct position to form a dot on the paper.

The major difference between the inkjet and the better-known impact dot matrix printer lies in the former's minimal mechanical complexity. And because the mechanical parts are the tricky bits in printers, the theory is that breakdowns are likewise reduced to a minimum.

In addition the inkjets have a smaller footprint — a relevant factor on lots of desks where system boxes, disk drives, modems and all the other essential paraphernalia of the modern office are vying for space.

But the most obvious advantage of the inkjet is its very quiet operation. Almost all the noise comes from the mechanism that slides the print head along the platen, and that's not a whole lot. Even when sited right next to you on the desk, the inkjet is perfectly pleasant in operation.

The buts ...

The commonly-cited disadvantages of inkjets are that ...

- they are 'specialist' printers for specialist operations
- inkjet printers are expensive
- they require expensive paper
- they can't produce multiple copies
- the ink cartridges are messy to change

It's true that early applications *were* specialised — for a long while inkjets were restricted to customised label-printing machines and bar-code printers. That does not reflect the current state of the technology. however.

As for cost — well, maybe that's a valid point. Because inkjets are not widely used, there are none of the economies of scale permitted by the production levels of conventional printer types; and the lack of a mass market means there isn't much discounting. You may have to search a bit harder to find suppliers both for the printer *and* the consumables. Both the inkjets reviewed here retail at around £400 — and they are at the low end of the price range for inkjets.

And inkjet printers *have* been very fussy about paper in the past, as a result of the special ink formulation required. The ink needs to run freely in order to be squirted efficiently and precisely through the tiny jets, and it mustn't dry too quickly, or it will dry on the printhead and block up the jets. This makes for slow-drying free-flowing ink, requiring paper with special absorbent qualities.

Happily, both the models reviewed here can (with certain provisos) print on ordinary paper. In both cases, though, the manufacturers strongly recommend the use of special inkjet paper for top-quality printing.

The impossibility of producing multiple copies arises from the fact that the ink is sprayed on to the paper; there is no striking mechanism to create an impression through carbon, or whatever.

Print sample: Hewlett-Packard

HP THINKJET 2225C

Draft Mode is quite faint. Use **Bold printing for clarity** or,

Compressed-Expanded for quality and better presentation

Italics are not available on the Thinkjet

Compressed mode for Spreadsheets gives 142 characters per line

ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789 abcdefghijklmnopqrstuvwxyz

Print sample: Diconix

DICONIX Model 150

Draft Mode is quite faint. **Use Double-Strike for clarity or,**

the Quality font instead for better presentation

or the Enlarged-Condensed font as an alternative Pica
Italics are available on the Diconix 150

ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789 abcdefghijklmnopqrstuvwxyz

even tny SCRIPT for that Informal touch

Of course most offices have photocopying facilities these days. But carbon copies are still used for many purposes; there doesn't seem to be any way of getting round this deficiency in inkjets.

Both the Thinkjet and the Diconix printers use identical disposable ink cartridges, which are claimed to last for around 500 pages of A4. It's a clever cartridge and avoids a lot of the hassle formerly associated with inkjets; replacement is as simple as unclipping the used cartridge and banging in a new one.

Ink cartridges are available from Hewlett-Packard at £8.81 each for black and £11.12 for red, blue, and green. The Diconix supplier quoted £10 each, for black only, though the manual says the ink is 'available in colours' — because they're the same cartridges, you presumably have to go to HP for colour!

The boxes

The ThinkJet I looked at was the model 2225C — the full range includes battery-powered models for portable computers (said by Hewlett-Packard to have a 200-page life) and desktop models with mains power. The mains model I've been using isn't much larger than a good dictionary; but it's twice the height of the Diconix and weighs twice as much (6.8lb).

The tiny Diconix Model 150 weighs in at a minuscule 3.75lb, including batteries. Of course you also need its interface cables, AC adapter-cum-charger, paper, manual, and so forth — and you'll find that they outweigh the printer. Even so, you're hardly looking at a great burden.

The Diconix is designed to be portable. Its five rechargeable batteries, rather cleverly housed in the platen, should give 50 minutes continuous printing before a warning light tells you that you've got two pages left. Full recharging takes ten hours; during that time you can interrupt the cycle to do *some* printing, as long as you've given the printer the chance to recuperate sufficiently!

Paper paths

Both printers take single cut-sheet paper up to 9in wide and pin/tractor-feed paper up to the US standard (11x8.5in after the perforated pin-feed strips have been torn off — slightly larger than A4, which can be used instead). Approximately half an inch of movement on the right-hand tractor feed of both printers covers a small variation from the standard US paper. Neither have manual paper advance control, unless you count pulling the paper through by hand.

Paper is reasonably easily to feed into both printers. The Diconix scores on the output side, with a three-position top cover hinged at the back: it can be used closed, folded to 45 degrees, or folded back horizontally. In the last position the paper output path is held well clear of the feed, and clear of the interface and power cables too (they are plugged in on the lefthand side).

The output path on the Thinkjet is awkward. The paper separator is a clip-on affair, and doesn't in fact hold the paper clear of very much at all. The interface and power cables attach at the back of the box, and to make matters worse the power cable plug extends slightly below the base. Hewlett-Packard offers an optional printer stand to remedy this — otherwise the printer will have to go at the back of your desk with the cable overhanging the edge. If you need to be nearer to the printer, you'll have to stand it on a book: hardly satisfactory.

To be honest, I wouldn't leave either of these printers unattended during lengthy print runs on continuous paper until I was satisfied that I'd sorted out a very safe paper path! There just isn't enough positive control over the paper paths.

The Diconix also has an 'envelope mode' which amounts to sideways printing restricted to 20 lines of 100 characters each. This is quite clever but probably not very useful; special envelopes are necessary, too (the print tends to smudge on ordinary office envelopes)and these are available either singly or as continuous feed.

The Diconix takes transparent overhead-projection foils as well; that's a facility the ThinkJet doesn't have, though Hewlett-Packard is shortly due to release a complementary PaintJet — a colour Thinkjet which will be faster than a plotter and *will* take foils.

Compatibility

Compatibility is a wily creature. Used by manufacturers to cover both control standards and character sets, in practice it often boils down to what you make of it.

Diconix offers compatibility with Epson FX and IBM Proprinter; the manual explains Epson-compatibility in three paragraphs and mentions IBM in two lines. HP offers a choice of its own control set, or an alternative (based on the Epson MX-80/IBM Graphics printer) which is claimed to be compatible with most software (and indeed probably is).

The Diconix prints the Epson set (which includes italics) and should print the full IBM character set including the graphics characters. Unfortunately I could only verify this claim with a test print as my best efforts to produce extended ASCII characters were fruitless. A 'hex dump' mode to print control sequences like the Thinkjet's would have been a boon, as I could have seen whether the problem lay with the hardware or me, but isn't available.

By comparison the Thinkjet character set doesn't have italics, IBM graphics or mathematics characters; but it does have the International section (ASCII codes 128 to 175) of the IBM character set.

Control

Both printers have panel switches to request form-feed, line-feed, and power on/off. The ThinkJet has a top-of-form switch too.

The Diconix has an 'Operator Panel' which gives control over QUALITY, SCRIPT and DRAFT options for printing, and also over the 'battery recharge' and 'envelope' modes.

The printers compared

	DICONIX MODEL 150	HEWLETT-PACKARD HP2225C
Print method	Disposable inkjet cartridge	Disposable inkjet cartridge
Power	five C rechargeable ni-cad batteries mains via external adaptor	mains via external adaptor
Interfaces: parallel serial	 standard optional	 standard n/a
Paper types	fanfold single cut sheets OHP transparency foils [special] envelopes	fanfold single cut sheets
Performance claimed: draft mode NLQ mode	 150cps 50cps	 150cps n/a
Control set	Epson FX-80 IBM ProPrinter	Alternate (Epson MX-80) Hewlett Packard
Character set	standard ASCII International ASCII IBM (inc graphics)	standard ASCII International ASCII IBM (alphanumerics only)
Graphic modes: low resolution high resolution	 yes yes	 yes yes
Fonts	Draft Condensed Enlarged Proportional NLQ Script Italics	Normal Compressed Expanded
Size	2 x 6.5 x 11in	3.5 x 8.1 x 11.5in
Weight	3.75lb (inc batteries)	6.8lb
Price [ex VAT]	£399	£425

Diconix Model 150

Description: Portable inkjet printer, battery-powered or mains adapter, Epson and IBM compatibility; 2KB print buffer

Supplier: Systems Connection [and others]

Address: Chatsworth House
29-31 The Broadway
Maidenhead
Berks SL6 1LY

Telephone: 0628 70912

Price: £399 (parallel interface), £409 (serial interface) — both ex VAT but including carrying case, batteries, printer stand

Conclusion: Very neat, smart, compact — maybe even a bit trendy. Full Epson-compatibility and impressive portability.

HP-2225C ThinkJet

Description: Mains-powered compact inkjet printer; 1KB print buffer

Supplier: Hewlett-Packard [via dealers]

Address: Eskdale Road
Winnersh Triangle
Wokingham
Berks RG11 5DZ

Telephone: 0734 784774

Price: £415 ex VAT

Conclusion: Not 100 per cent Epson-compatible, fiddly paper path; but well-constructed and worth considering as a quiet desktop printer.

The DIP switches on the Diconix are under the front top cover facing backwards. That might sound awkward; but in practice it's as workable as the Thinkjet's switch panel, which is mounted above the parallel interface at the back.

The Thinkjet's eight-setting DIP switch covers just the default settings for the printer. All other settings and changes to the defaults must be done through software.

The DIP switches for both printers cover CR, LF, perforation skip, and page length, as well as the selection of International fonts (national variations to ASCII) and the IBM character set (IBM's extended ASCII).

Each printer has a self-test mode selectable from the control panel — use it to see the printable characters and as a rough speed test. It doesn't quite simulate normal printing but it gets closer than the manufacturer's claimed speed. A rule of thumb is to expect half the claimed speed, because the manufacturers test conditions don't emulate normal printing conditions.

The test pattern allows you to assess the quality of the printed characters too. The Thinkjet includes a useful DISPLAY FUNCTIONS MODE which treats control codes other than CR/LF as printable symbols. Very useful if you have problems in setting up.

Getting going

Printer setup depends on your requirement — for the most part you *won't* get everything you're expecting unless you spend a while in the setting up, but you should get 'compatibility' with your software by using the simplest default settings. Beyond this the subject requires a full *Workshop* series to do it justice.

Both printers print at 12 characters to the inch as standard. In expanded-compressed mode on the ThinkJet you can print at 10.7cpi; with the Diconix, enlarged-condensed mode gives 9.6cpi. Pica (10cpi) isn't available on either. Diconix' maximum printable line width is a fraction over 7in; the Thinkjet does 6.6in.

Performance

In general, inkjet printers compare well with impact dot matrix printers in DRAFT mode; and the QUALITY or EMPHASISED modes are quite good enough for office documents. Although business correspondence is still supposed to require daisywheel or laser quality, I would be more than happy to use this inkjet quality for chasing debtors and other basic correspondence.

I found the Diconix needs to be used in a double-strike or quality mode to produce readable print: the Thinkjet prints more clearly in both DRAFT and BOLD. Diconix DRAFT is sufficient if you are a normal reading distance from the paper, otherwise use NLQ mode for improved readability (albeit with the usual printing speed penalty). It seems that the SCRIPT and NLQ options were very much a marketing requirement.

Paper requirements might still be a disincentive to some. Both printers smudge badly on coated or machine-finished paper *unless* you allow for drying time. And the ThinkJet paper path *guarantees* smudging on unsuitable paper unless you leave the front cover open — thus increasing the decibel level, though not by much.

Is inkjet paper too expensive? Is it really necessary? Inkjet pin-feed paper tearing down to 11x9.5in is available from the manufacturers in 2,000 sheet quantities at £28 (around 1.4p a page); by contrast, standard 75gsm paper from a quality supplier costs around £22, 1.1p a page. Microperforated A4 costs a bit more.

It might be feasible to look for an uncoated standard paper which is slightly lighter, say 60gsm. The lighter paper has a more open 'weave' and may absorb ink better. Or try the other side if it looks to have a rougher finish.

But I would suggest that if you don't need to handle the paper immediately and can leave the printing long enough to dry, both printers would perform adequately. The only problem is the paper path on the ThinkJet; and if you can contrive to arrange this successfully the paper problems disappear.

Last words

An inkjet is faster than a laser for individual pages, and much quieter than an impact printer. It doesn't require a special ribbon as the other 'silent-technology' printers do; and, within limits, the inkjet can use ordinary paper.

For their size, these printers perform well. You should be able to use them with most applications programs within the restrictions imposed by the printer size. I found using both printers straightforward and, apart from the IBM emulation of the Diconix, there were no problems. Follow the golden rule and ask the salesperson for a test print before you buy.

The Diconix is totally portable, without any doubt. The Thinkjet *can* be if you don't need to use it too far from mains power; but it's bigger, heavier, and not as 'standard' as the Diconix. The Diconix's Operator Panel is convenient. Both manuals are usable, if terse (presumably as printers get ever smaller the manufacturers are increasingly pressured into producing even smaller manuals — and that's quite a lot of pressure in Diconix's case!).

But it's the advantages of super-quiet printing that can't be too highly emphasised. You may have adjusted to the noise of your printer until it's simply part of the background. But if you *can* remove another, albeit small, source of stress and strain, then why not do it?

Which to buy? I would probably pick the HP for late-night desktop printing and for quick 'n' easy dumps from comms sessions; I prefer its print styles. But if I needed a portable printer to go with a portable computer, it would have to be the Diconix. ◆

A Turbo Pascal Primer

Part Three: Text files

By Mark Whitehorn

Last month I looked at the advantages of writing in a structured manner. My friend and colleague Stuart Bunt (KERMIT wizard of the North) pointed out that a useful rule was missing from the list I had given about structured programming. He was quite correct, so here is another to add to your list:

> **There should be only one way into a procedure (which is by calling the procedure's name). Less obviously, there should be only one way out.**

Multiple exit points from a procedure can make debugging much more difficult since you are always that bit less certain where the program is going at any one time; and jumping out of a procedure can cause problems if performed repeatedly.

And so, back to the series. We'll start with a class of files recognised by Turbo Pascal as 'text files' — essentially those which contain ASCII characters. These are the sort of files that word processors for example will produce. Since most people use word processing, text files seem a good place to start. Turbo Pascal allows you to create, read from, and write to text files quite easily.

Writing to files

MS-DOS knows your files by their specification, a mix of filename, paths and drive identifier such as A:\SUPER\PENGUIN.TXT. Turbo Pascal must access files on disk via MS-DOS, so it has to use the full file specification outside the program. *Inside* the program, though, the language allows you refer to the file by a shorter name (known as the File Variable). So early on in the program Turbo Pascal has to be told to tie in the file variable you will use in the program.

This 'tying together' is done using the procedure Assign, as demonstrated below. The full file specification must be put in a variable before Assign is used.

The tiny demonstration program in the panel creates a file on the default drive called PENGUIN.TXT.

```
{----------------------------------------}
Var
 NameUsedInsideProgram  :text;
 FileSpecification      :String[30];

Begin;
 FileSpecification := 'A:\PENGUIN.TXT';
 Assign(NameUsedInsideProgram,FileSpecification);
 Rewrite(NameUsedInsideProgram);
 Writeln(NameUsedInsideProgram,'PENGUINS LIKE ');
 Writeln(NameUsedInsideProgram,'FISH.');
 Close(NameUsedInsideProgram);
End.
{----------------------------------------}
```

In order to write to a file, you need to ...

1. Declare the file as a text file in the Var section
2. Open the file
3. Write to the file
4. Close the file.

Step One is to choose the name by which the file is to be known within the program (the File Variable). Following the idea of using sensible names I have called it 'NameUsedInsideProgram'. Since we are dealing with a text file, the file variable is declared in the Var section as a text file.

Then a variable is chosen to hold the real name of the file as it will appear on the disk. Again a reasonable choice for a demo would seem to be 'FileSpecification'.

I have declared the variable 'FileSpecification' as a string 30 characters long because that allows me to use a lengthy filename such as A:\WORD\WORK\PENGUIN.DOC without fear of producing a string too long for the variable. In some ways this is a good example of the sort of programming I discussed last month. In the example following, the string held in the variable RealNameOnDisk is PENGUIN.TXT — a total of 11 characters. I could save 21 bytes of memory by declaring the variable as a string 11 characters long. But I'm not really interested in saving 21 bytes, I am more interested in avoiding hassle if and when I modify the program in later months.

Before opening the file, the file variable must be attached to the actual name on disk (which is held, as I said, in a variable). This is done using Assign. Now the file can be opened using Rewrite — used it with caution, for Rewrite will destroy any file of the same name which is already on the disk.

Writing to the file is similar to writing to the screen except that the file variable must be inserted before the information to be written.

Closing the file is performed using Close and the file variable. This makes sure that both Turbo Pascal and MS-DOS are aware that the file *is* closed; if it's left open, you may have trouble reading from it later.

Reading from files

Reading requires the same sort of process. The file variable must be assigned to the actual file specification, and the file then opened as before — except that Reset is used this time.

Reset does not destroy the file; it simply opens the file for reading. If the file is already open (which might be the case if you've been writing to it earlier in the same program) Reset will first close the file safely before reopening it for reading.

After Reset, Reading can be performed; and when that is complete, the file may be closed.

```
{-----------------------------------------}
{Note: this program will generate an error unless
PENGUIN.TXT already exists on disk.}

Var
 NameUsedInsideProgram  :text;
 RealNameOnDisk         :String[30];
 Temporary              :String[255];

Begin;
 RealNameOnDisk := 'A:\PENGUIN.TXT';
 Assign(NameUsedInsideProgram,RealNameOnDisk);
 Reset(NameUsedInsideProgram);
 Readln(NameUsedInsideProgram,Temporary);
 Clrscr;
 Writeln(Temporary);
 Close(NameUsedInsideProgram);
End.
{-----------------------------------------}
```

Standard files

Turbo Pascal treats *all* input and output as if it were being performed to or from a file. This for example means that output to the screen, input from a keyboard, and output to a printer are all treated as file I/O.

In order to save the programmer having to open and close these files continuously, Turbo Pascal looks after this automatically: you don't have to open the screen before writing to it. Turbo Pascal provides several of these so-called 'Standard Files', which can be used without being opened and closed. The most useful are:

- **Input** — basically, this is the keyboard at which you type
- **Output** — basically, the screen
- **Kbd** — the keyboard with no echo of characters to the screen and no interpretation of the characters
- **Lst** — the printer
- **Aux** — often used for the serial port.

Thus to output information to the printer you can use a line like this, without having to open the file Lst first or close it later:

```
Writeln(Lst,'Hi there.');
```

Turbo Pascal goes one step further in making our lives easier. The two standard files Input and Output (reading from the keyboard and writing to the screen) are used so much that they're treated as the default. So lines like ...

```
Readln(Temporary)
Writeln(Temporary);
```

... are much more common than:

```
Readln(Input,Temporary);
Writeln(Output,Temporary);
```

Tools for writing to files

You are probably familiar with the two procedures Write and Writeln, normally used to write to the screen.

Write will display a string of characters, Writeln displays a string of characters and then adds a Return (CR) and a Line Feed (LF). The effect of this is not apparent at the time the Writeln is used, because the CR and LF are written after the text. But any text that is subsequently displayed will appear on the lefthand side of the screen (because of the CR) and on the next line down (because of the LF). So the line ...

```
Write('Penguins like '); Write('fish.');
```

... produces the output:

```
Penguins like fish.
```

While this ...

```
Writeln('Penguins like '); Write('fish.');
```

... produces a second line of output:

```
Penguins like
fish.
```

Writing to files other than the default produces exactly the same result. If you use Writeln, a CR and LF will be added after the string of characters are written.

Tools for reading, file pointers

When reading, Turbo Pascal has a File Pointer which always points to the next character in the file — the character you're about to read. Take this two-line file:

```
Penguins like
fish.
```

When the file is first opened to read the information therein, the file pointer will point to the character 'P'. Suppose you read a single character from the file, using Read and a character variable called Char:

```
Read(FileVariable,Char);
```

The 'P' will be read and placed in the variable Char, and the file pointer is moved to the letter 'e'. If you had Read from PENGUIN.TXT into a four-character string variable, you would have extracted 'Peng' and the file pointer would be pointing to 'u'.

One characteristic of the Read procedure is that it stops when it reaches an EOL (CR and LF together are called an end-of-line — EOL — marker). It

refuses to proceed until a ReadLn (see below) is used. If you include a Read inside a loop, this can easily hang your program. One way to avoid hanging the program is to use the expression ...

```
While not Eoln do ...
```

You may also recall that last month's sample program has a Read procedure in a loop. This ought to hang the program, but it doesn't; this is the anomaly I mentioned at the time. It turns out that if the variable into which you Read is of type Char, the procedure seems quite happy to chew its way through the CR and LF and carry on to the next line. This undocumented feature is of great use in file manipulation.

The Readln procedure

When Readln is used to take information from a file, it reads as many characters as it requires and then advances the file pointer to the first character of the next line (the first character after a CR/LF).

Readln requires a little care. For example, if you declare a variable to be 10 characters long and then Readln from a file into that variable, you'll extract ten characters from the file. If you happen to have been reading from a line with 50 characters on it, you will never see the other 40; next time you read from the file, you'll start reading at the next line. If you wish to be sure of getting all the characters on a line, use Read — or Readln into a l-o-o-o-ng string variable.

Incidentally, although CR/LF is documented as the end-of-line marker, Readln seems to treat CR on its own as an EOL (but Readln ignores lone LFs).

End tests

Eof is a function which tests whether the file pointer is pointing at the end of the file. This is particularly useful for bits of program like this:

```
While not Eof(InputFile) do
Begin;
...
End.
```

In effect this means 'do the bit of program which is between the Begin and End until the file is empty'. The Eoln procedure is similar, except that it is true if the file pointer points at an EOL marker.

These functions and procedures are all you need to play with text files. Now you can strip control codes out from word processor files, create files, or write your own word processor! ◆

Contributors in this issue

Dennis Jarrett is Editor of the newsletter. His magazine credits include being founder-editor of *16 Bit Computing*, *Office Automation Report*, *Practical Computing* and *Which Computer*. Among his books are *The IBM PC Book*, *The Sirius/Victor 9000 Book*, *The Electronic Office*, and *The Good Computing Guide* (*really* thick and very cheap too at £2.95 — or £3.00 inc P&P from DEALS FOR READERS).

Henry Deckhand is a pseudonymous cynic whose role at the newsletter is to generate jaundiced opinions about the computer industry and the way it conducts itself in public.

Lindsay Doyle currently writes for APRICOT FILE on a variety of esoteric subjects. Over the years he has contributed under various pseudonyms over 75 articles to the likes of *Personal Computer World*, *Microcomputer Printout*, 16 BIT COMPUTING, the Epson Users' Group, *Pet User Notes*, and various other Pet, CBM 64, and Vic publications. (Most of these are now defunct — is there a message here somewhere?) In addition to pounding away on an Apricot Xi of long standing, on which all the real work gets done, he is now playing with a new Amiga and HP plotter.

Terence Green is an unemployed Apricot owner (currently looking for a job). Formerly an unlicensed truck driver in Lesotho, he started on IBM 370s and worked his way down to micros; these days he minds the kids and scrapes a living washing dishes (and writing occasionally for APRICOT FILE).

David Langford is a full-time author and computer addict, whose latest book *EARTHDOOM!* — a spoof of disaster novels, written with John Grant and scorned by Kingsley Amis — is "by no means" about Apricot BIOS variants. (The inclusion of a permanently drunk character called Apricot should not be misinterpreted.) He is inordinately proud of being a guest of honour at the 45th World SF Convention last August.

Alan Solomon is a writer and consultant who writes programs in Fortran, Pascal and assembler. They're marketed by his wife's software house, S&S Enterprises. He also runs a SIG in the IBM PC Users' Group.

Mark Whitehorn started life as a Cytogenicist but the Siren Song of the computer was too strong and he is now working as a programmer at a Scottish university, particularly on the application of computers (especially Apricots) to education. He says his hobbies include penguins, building 1960s racing cars on the cheap, and indoor underwater powerboat-snooker (of which only the last is apparently untrue).

Tudor Wynn-Williams spent seven years in Helsinki rewriting reports and publicity material written by Finns in English. Now back in Britain, he does similar work freelance, sending texts to Finland by electronic mail for direct setting.

Favourite things

WordPerfect does it again

WordPerfect is widely regarded as the best word processor available for desktop computers.

Even the December copy of the Consumers' Association magazine *Which?* included WP packages along which the 'which turkey' reports, and gave WordPerfect Best Buy status along with Microsoft Word 3.0 (admittedly from a very short list).

And the October issue of the estimable *What to Buy for Business* carries a much more sensible report on WP software that also names WordPerfect and Word as Best Buys — **Samna Word IV** (IBM-compatibles only) and the undersung **WordCraft** are rated 'worth a look'.

But while the package itself may be a thing of beauty, WordPerfect's *purveyors* don't seem as professional as you might expect.

Last month's APRICOT FILE mentioned the hassles of **Ansible Software** in dealing with WordPerfect Corp and its UK rep **Sentinel Software**; Ansible was taken to task for including the name 'WordPerfect' in the title of its WordPerfect utilities disk, which is why the same product is now called PARAGON.

Well, about the same time we received a press release from Sentinel quoting market research by **Romtec** for word processor sales in July. The figures had WordPerfect top for sales value again, a position it has occupied for some months; but for the first time it also had the lead in unit shipments, with just over **17 per cent** of the total. Hooray!

Or not. A a couple of days later, a retraction arrived from Sentinel. It seems Sentinel's PR people jumped the gun; the original press release quoted actual shipment figures — which are copyright to Romtec and confidential to Romtec's subscribers.

What's more, Sentinel isn't among those subscribers (Romtec would like to know who gave Sentinel the figures). And the July figures were distorted anyhow by a special bundling offer on Amstrads; Romtec's August results showed WordStar 1512 in the lead again for unit sales, with WordPerfect back at its 12 per cent level in second place.

WordStar vendor **MicroPro** took the opportunity to rub the egg well into Sentinel's face with a press release of its own, quoting MD **John Speller**: "I have great faith in the UK industry and its press and I don't believe that they will be taken in by some of the more dubious tactics of certain of our rivals ..." Now who could he be talking about?

Diary dates

On with the Shows

A round-up of potential relevance

January 1988

19-22 Which Computer Show *Birmingham.* The Big One — probably the show to visit if you want to visit only one computer-oriented exhibition during the year. Held at the NEC as usual. TP Group has a (small) stand at the Show — drop in and meet us! *Organiser: Cahners, tel 01-891 5051*

19-22 Office Update *Birmingham.* New NEC show for purchasing management runs alongside the Which Computer Show. *Organiser: Cahners, tel 01-891 5051*

12-14 Corporate Computer Security *Brighton.* Britain's first exhibition on the subject; associated multi-session conference and workshops. *Organiser: PLF Communications, tel 0733 60535*

13-14 Local Area Networks *London.* One of the IBM PC User Group's excellent series of economical and amiable conferences — speakers from the Group's SIGs and from vendors, admission £150 (or £200 for non-members). *Organiser: IBM PC User Group, tel 01-620 2244*

27-29 Videotex User '88 *London.* The only UK exhibition devoted to viewdata applications; run in association with the DTI and the Videotex Industry Association. Viewdata is at something of a crossroads right now, with its early identity crises being resolved just at the time when alternative communications media are looking attractive. This show and associated two-day conference (now in its third year) may help with a snapshot of the business. *Organiser: EMAP, tel 01-608 1161*

March 1988

3-5 CIM 88 *London.* First running for this Corporate Information Management conference, covering the acquisition, use and control of information — majors on on-line systems, laser discs and videotex. *Organiser: Effective Technology Marketing, tel 0462 57798*

22-24 Electronic Print and Publishing Exhibition *London.* The DTP business is moving so fast that all shows are interesting for the new products on view. *Organiser: BED, tel 01-647 1001*

22-23 Introduction to Data Communications *London.* Reportedly one of the best intensive seminar introductions for "newcomers with little background or experience". £530 per attendee. *Organiser: Frost & Sullivan, tel 01-730 3438*

22-24 Lotus World *London.* Second running for this introverted show — it was very good last year, and had some 5,000 visitors. *Organiser: Horne & Associates, tel 0256 83456*

Turn and turn about

Epicycloids

By Lindsay Doyle

Your humble scribe is a proud member of the Society of Ornamental Turners. What, you ask, is Ornamental Turning (hereafter referred to as 'OT') and what does it have to do with Apricots?

Whereas Plain Turning is the making of figures of revolution such as bowls and banisters, OT is the decoration of the surfaces of such pieces with patterns and the turning of more complex shapes such as helixes, spheres within spheres, and the like. Both are done on a lathe, but the OT lathe is considerably more complex.

OT was a hobby of the Victorian rich upper classes. Although some beautiful work of lasting value was done in that period, most amateur work which survives tends to be cluttered and overdecorated by current standards. But present-day practitioners who do not slavishly copy the Victorians are doing work which could grace any collection of articles of virtu. (Any reader who would like to learn more about the state of the art may consult references 1 to 4.)

The geometric chuck

One of the more exotic gadgets available to the OT enthusiast was the geometric chuck, with which he was able to engrave on a flat surface beautiful and complex designs. Some of these designs belong to the mathematical family of curves called epicycloids, and this article is about the reproduction of these on the computer screen, printer, or plotter. If you have cheated and looked at the illustrations before being told to, you know what I am talking about and can decide for yourself whether you would like to be able to reproduce any of these or the thousands of other possible variations.

A *cycloid* is the curve traced by a point on a circle as it rolls along a plane. A locomotive wheel provides a good way to visualise the process. If you like to drop names to impress your family and associates, I am able to reveal that a point on the part of the wheel which contacts the track traces a 'common cycloid'; a point nearer the hub traces a 'prolate cycloid'; and a point outside on the flange traces a 'curtate cycloid'. It is interesting to note that when a locomotive is moving forward, part of the wheel is moving backward and part is moving twice as fast as the locomotive! Queries to the editor. (*You must be joking — Ed.*)

An *epicycloid* differs from a cycloid in that it is the curve traced by a point on a wheel rolling on the circumference of another wheel rather than on a straight line, thus being a cycloid which curves around and meets itself in 360° (or multiples thereof, as we shall see). Prolate, common, and curtate varieties are found as before.

Some readers may have fond memories of a toy called the Spirograph, which consisted of plastic gears of various diameters with holes into which a coloured pen or pencil could be placed. One gear was to be pinned to the drawing paper while another was rotated around it. This was surely the cheapest way to get into the epicycloid business; but it was limited to prolate designs, and could not be adapted to three or more gears, as we shall do with the computer. A Spirograph has nothing to do with spirals: I suppose it was called the spirograph because 'epicycloidograph' was too much of a mouthful.

The Victorian OT nomenclature described curves generated with two gears as 'simple', with three gears as 'compound', and with a fourth gear as 'advanced compound'. I shall retain this naming system, clumsy as it is, so as to make it possible to follow the material in the appended references if anyone is so inclined.

The computer solution

I could never purchase or build a geometric chuck — existing samples are museum pieces. But it occurred to me one day that I would be able to simulate its action on the computer, and even carry the creating of patterns to a degree impossible to the Victorians due to the limitations of mechanism accuracy.

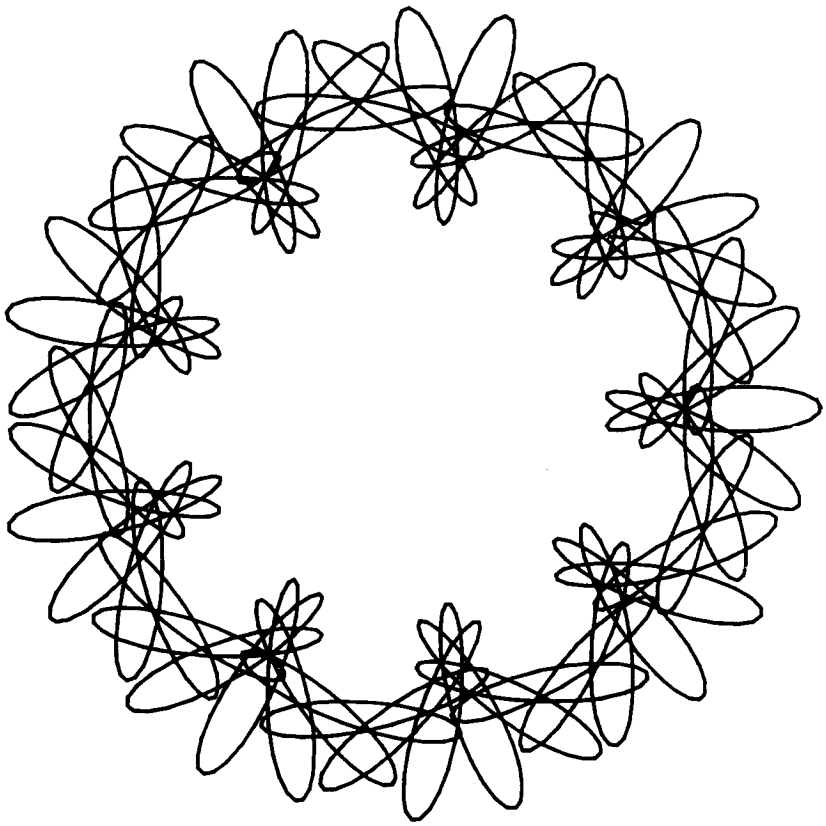
We can display epicycloids on the computer screen, print them with a dot printer, or plot them with an X-Y pen plotter.

The computer *screen* has the least resolution and usually needs to have its aspect ratio corrected. Curves look jagged due to the low resolution, but it is satisfactory for quick tryouts. A *dot matrix printer* such as the Epson FX-80 has far higher resolution than the screen; but a pattern must be generated line by line, unless complex reverse paper-feed problems can be catered for. The *plotter* is the best solution of all: the pen follows anywhere on the paper and draws a line between computed points, giving the effect of a continuous curve as can be seen in the examples accompanying this article. Multiple colours of pen can also be used to advantage. So I wrote my program to drive a Hewlett Packard 7470A plotter, which no home should be without. You should be able to modify the program without trouble to drive the screen from GW-BASIC if you wish.

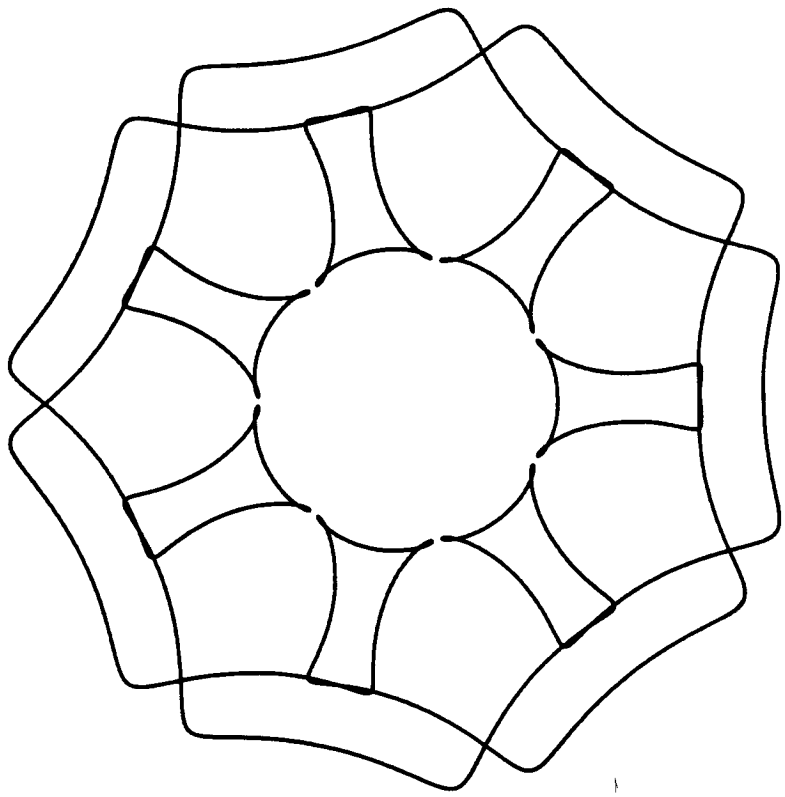
Great minds think alike

'Synchronicity' is the name of a quasi-science which seeks to explain coincidence by postulating a relation between events other than that of cause and effect. Perhaps it was behind the fact that I had no sooner completed my epicycloid-generating program than I was made aware of a modern book on the same subject called *Microcomputer Art*.

PATTERN NO. 592
11-03-1987
SC = 1
R = 50
S = 10
T = 8
N = 61
M =-50



PATTERN NO. 540
11-03-1987
SC = 1
R = 40
S = 20
T =-4
N = 9/2
M =-33/2



The title is misleading, since the book only covers epicycloids. It appears to have been written in Australia for youngsters who have had some minor exposure to trigonometry, as it is greatly simplified and spells out each step in excessive detail. There are some errors, and some points which the author does not fully understand; but I recommend it for the simple reason that it contains hundreds of illustrations of curves along with the tables of parameters required to duplicate them. Because some may refer to it, I have incorporated its rather oddly-chosen variable names into my program.

Circles on circles

A circle in X-Y coordinates is defined by the formula:

```
X = R * COS(A)
Y = R * SIN(A)
```

... where *R* is the desired radius and *A* is an angle varying from 0 to 360°. To get the circle in the centre of the paper we might write:

```
X = R * COS(A) + XC
Y = R * SIN(A) + YC
```

... where *XC* and *YC* are the coordinates of the desired centre. Now to add a second circle rotating about the first:

```
X = R * COS(A) + S * COS(N * A) + XC
Y = R * SIN(A) + S * SIN(N * A) + YC
```

... where *S* is the radius of the second circle and *N* is a factor which we shall discuss in detail.

The plot of these points will be an epicycloid of the simple kind. Its shape will depend on the relative values of *R*, *S*, and *N*. *N* may be positive or negative: pairs of curves generated by positive and negative *N*s of the same value are called 'companion curves'.

A positive *N* gives a pattern with *N*-1 internal loops; a negative *N* gives *N*+1 external loops. If *N* is a whole number, the pattern will be complete in one revolution. Such curves were called by the Victorian masters of the art 'consecutive'. If *N* is expressed as what I seem to recall used to be called a 'vulgar fraction' — one whole number divided by another — the denominator defines the number of revolutions; for example, when *N* = 9/2, the pattern will only close on itself after two revolutions and the range of *A* has to be modified accordingly. Such curves were called 'circulating'.

I will leave it to you, the interested reader, to develop other rules for yourself or to learn them from reference 5. Broadly speaking, you can get a very good idea of the type of pattern which will result from a given set of parameters simply by applying these rules to the parameters.

The third and future wheels

If a third wheel is to be added, to make 'compound curves', it has to have radius *T* and factor *M* specified. The Victorians called geometric chuck curves made with four wheels 'advanced compound curves' and were not able to go any further. We can easily do four or more; but the patterns tend to become uninteresting, being for the most part simply embellishments consisting of loops replacing the lines of compound patterns. For this reason I have restricted my program to the choice of two or three wheels, but the means of adding further wheels will be apparent to the reader. The equations for a three-wheel setup are:

```
X = R * COS(A) + S * COS(N * A) + T * COS(M * A) + XC
Y = R * SIN(A) + S * SIN(N * A) + T * SIN(M * A) + YC
```

What a program should do

Since the whole point of the exercise is to experiment to find patterns which have particular aesthetic value, I start by providing for an optional label in the upper left corner. This gives a curve identification number assigned by the user, the date, and a list of the selected parameters. I also provide for an optional title centred at the bottom of the page.

The program accepts values for ID number, radius *R* of first gear, radius *S* of second gear, and (optionally) radius *T* of third gear. Then it requests that the value of *N* be keyed in as a numerator followed by a denominator (defaulting to 1 if you just press the Return key). This is followed by a similar two-part input of *M* if a third gear has been specified.

The user is then given an option of step size (the smaller the step in angle, the better the resolution but the longer the time taken to plot). Full scale (the largest size which will fit on the paper) or fractional scale may be specified, and the user is prompted for a label and/or title.

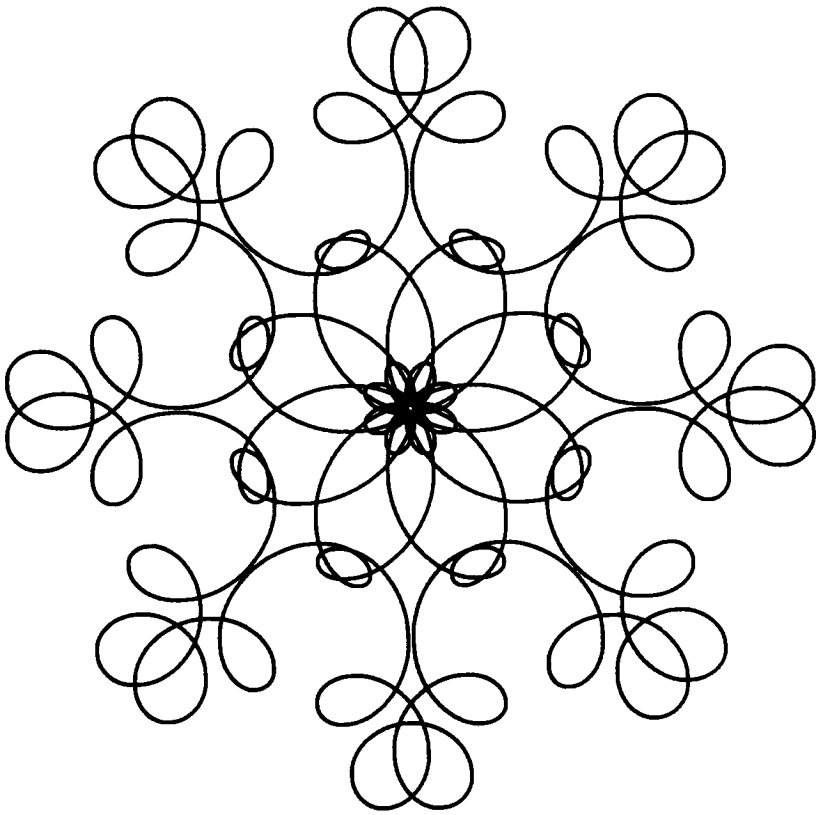
After these inputs, most of which have default values, plotting proceeds until finished. At that point the user is given a choice of adding to the present pattern, making a new pattern, or quitting.

I have found that these are the necessary and sufficient functions for my needs, but other users can easily add customising features of their own.

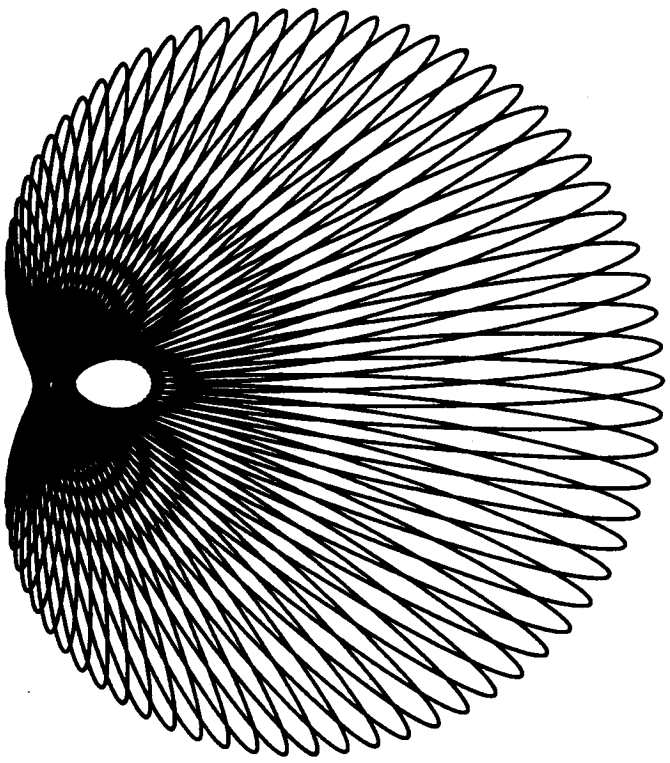
As the parameters of the curves are in arbitrary units, they are scaled by the program to suit the dimensions of the plotter paper. Because the plotter has equal sensitivity in the X and Y dimensions, no aspect ratio correction (such as might be needed if printing to the computer screen) is included.

My experience with plotter pens is that new ones can be used at the maximum speed the plotter is capable of, but that speed should be slowed so as to allow the use of old pens which otherwise might draw irregular line widths. Such a constraint is not incorporated but could be added by those who like to pinch their sixpence till it squeaks.

PATTERN NO. 551
11-03-1987
SC = 1
R = 30
S = 30
T =-8
N =-3/5
M =-51/5



PATTERN NO. 593
11-03-1987
SC = 1
R = 36
S = 28
T = 24
N = 56
M =-55



Types of patterns

While all the patterns which this program can produce are epicycloids, there are many individual types which can be generated by imposing certain restrictions on the parameters. I shall mention six of the most significant ones and submit a superfluity of examples from which the Editor can chose to reproduce those he likes and has room for.

1. One-line patterns are all those which consist of a single line ending up by meeting with itself, whether in one or more revolutions.
2. There are patterns consisting of a family of expanded copies of type 1. These break down into five types, all of which may be experimented with to taste:
   - vary R, keep ratio of S to R constant
   - vary R, keep $S = R$ - constant
   - vary R, keep $S = R/N$
   - vary R, keep S constant
   - keep R constant, vary S
3. It is also often fructitious to take a pattern of type 1, if not too complex, and reproduce it many times with each copy rotated about its centre by an incremental angle.
4. With care to avoid complex messes which have no redeeming value, you can also generate combinations of types 2 and 3 incorporating simultaneous rotation and expansion.
5. Clusters of radial straight lines where $R=S$ and $N=-1$ generate interesting starburst patterns.
6. You can also lay down patterns in other than a circle (along a straight line, for instance, or along the sides of a triangle.)

Reference 5 gives parameters for many patterns of each type if you are willing to be guided by the experiences of another. ◆

References

1. *Ornamental Turnery* by Frank M Knox, Prentice-Hall, 1986. Frank Knox is one of the outstanding OTers today and his book is an excellent introduction to the art
2. *The Principals and Practice of Ornamental or Complex Turning* by John Jacob Holtzapffel, reprinted by Dover Publications, 1973. This 656-page tome is the 'bible' of OTers and first appeared in 1884 as the fifth in a series covering the entire theory of turning
3. Samples of OT work and the special lathes on which they were made can be seen at the Science Museum in London
4. The worldwide organisation for perpetuation of the art is the Society of Ornamental Turners, whose secretary is Philip J Holden, 17 Chichester Drive East, Saltdean, Brighton, Sussex BN2 8LD. The SOT has an extensive library and collection of equipment and it produces a regular bulletin with articles discussing current-day techniques
5. *Microcomputer Art* by Ross Edwards, Prentice-Hall of Australia, 1985, is the text referred to in the article and contains extensive tables of parameters and examples of patterns

EPICYCLE.BAS

```
10    REM: EPICYCLE.BAS: draws epicycloidal curves on plotter
20    REM:
30    REM: MS-BASIC for Apricot Xi driving HP 7470A plotter
40    REM: Lindsay Doyle 7/4/87
50    REM:
60    REM: No screen display, no attempt to speed up computation
70    REM: Codes within LPRINT statements are commands to HP plotter
80    REM:
100   REM: CONSTANTS
110   PI = 355/113: C$ = CHR$(27): REM: ESCAPE
120   E$ = C$ + "E": U0$ = C$ + "0": U1$ = C$ + "1": B$ = CHR$(7): T$ = CHR$(3)
130   REM: ^ CLEAR SCREEN, UNDERLINE ON, UNDERLINE OFF, BELL, LABEL TERMINATOR ^
140   HE$ = E$ + SPACE$(32) + U0$ + " Epicycloids " + U1$: REM: HEADING
150   XMAX = 10300: YMAX = 7650: XCENTER = XMAX / 2: YCENTER = YMAX / 2
160   REM: ^ FOR HP 7470A PLOTTER ^
195   REM:
200   REM: INITIALISE FOR ANOTHER PLOT
210   LPRINT "IN": REM: INITIALISE PLOTTER
220   LPRINT "SP1": REM: GET PEN
295   REM:
300   REM: INSTRUCTIONS?
310   PRINT HE$
320   PRINT
330   PRINT: INPUT "Instructions"; I$: IF I$ = "Y" OR I$ = "y" THEN GOSUB 1100
340   PRINT
```

```
395   REM:
400   REM: GET PARAMETERS AND NORMALIZE
410   INPUT "IDENTIFICATION NUMBER..... ", ID: IF ID <1 OR ID > 700 THEN GOTO 410
420   INPUT "RADIUS OF FIRST GEAR..... ", R: IF R < 1 THEN PRINT B$;: GOTO 420
430   INPUT "RADIUS OF SECOND GEAR..... ", S: IF S < 0 THEN PRINT B$;: GOTO 430
440   INPUT "RADIUS OF THIRD GEAR OR <RETURN>... ", T: REM: MAY BE MINUS
450   INPUT "NUMERATOR OF N..... ", NN: IF NN = 0 THEN GOTO 450
460   INPUT "DENOMINATOR OF N OR <RETURN>..... ", ND: IF ND = 0 THEN ND = 1
470   N = NN / ND: IF SGN(NN) = -1 THEN LOOPS = ND - NN ELSE LOOPS = NN - ND
480   IF T = 0 THEN GOTO 520
490   INPUT "NUMERATOR OF M..... ", MN: IF MN = 0 THEN PRINT B$;: GOTO 490
500   INPUT "DENOMINATOR OF M OR <RETURN>..... ", MD: IF MD = 0 THEN MD = 1
510   M = MN / MD
520   INPUT "STEP SIZE (.001 TO .05)..... ", STP
530   IF STP <.001 OR STP > .05 THEN PRINT B$;: GOTO 520
540   INPUT "SCALE (DECIMAL <= 1) OR <RETURN>.... ", SC: IF SC = 0 THEN SC = 1
550   IF SC < 0 OR SC > 1 THEN PRINT B$;: GOTO 540
560   INPUT "0 = NO LABEL, 1 = SHORT LABEL, 2 = LONG LABEL...", LABEL
570   IF LABEL < 0 OR LABEL > 2 THEN PRINT B$;: GOTO 560
580   INPUT "TITLE OR <RETURN>..... ", TITLE$
595   REM:
600   REM: SCALE TO FIT Y-DIMENSION OF PLOTTER
610   RADIUS = (R + S + ABS(T) + 15) / SC: REM: T CAN BE MINUS
620   RS = R * YCENTER / RADIUS: SS = S * YCENTER / RADIUS: TS = T * YCENTER / RADIUS
695   REM:
700   REM: PLOTTER SETUP AND LABEL
710   PRINT "OK. PLOTTING": LPRINT "SP1"
720   IF LABEL = 0 THEN GOTO 880
730   LPRINT "PA 0," .98 * YMAX ", LBPATTERN NO." ID T$
740   LPRINT "CP, LB" DATE$ T$
750   LPRINT "CP, LBSC =" SC T$
760   IF LABEL = 1 THEN GOTO 880
770   LPRINT "CP, LBR =" R T$ "CP, LBS =" S T$
780   IF T <> 0 THEN LPRINT "CP, LBT =" T T$
790   LPRINT "CP, LBN =" NN T$
800   IF ND > 1 THEN LPRINT "CP-1,0, LB/" T$: LPRINT "CP-1,0, LB" ND T$
810   IF MN <> 0 THEN LPRINT "CP, LBM =" MN T$
820   IF MD > 1 THEN LPRINT "CP-1,0,LB/" T$: LPRINT "CP-1,0, LB" MD T$
830   IF TITLE$ = "" THEN GOTO 900
840   LPRINT "PA" XCENTER ", 0, CP" (-LEN(TITLE$) / 2) ",1, LB" TITLE$ T$
895   REM:
900   REM: ALGORITHM
910     FOR J = 0 TO 2 * PI * ND + STP STEP STP: REM: ENSURE CLOSURE
920       X = RS * COS(J + I) + SS * COS(N * J) + TS * COS(M * J) + XCENTER
930       Y = RS * SIN(J + I) + SS * SIN(N * J) + TS * SIN(M * J) + YCENTER
940       IF INKEY$ = "x" OR INKEY$ = "X" THEN GOTO 1020: REM: INTERRUPT
950       LPRINT "PA" X "," Y "PD;": REM: POSITION BEFORE PEN DOWN 1ST POINT
960     NEXT J: LPRINT "PU;"
995   REM:
1000  REM: FINISHED: ANOTHER?
1010  PRINT E$; B$; "PLOT COMPLETED"
1020  INPUT "A = Add to present pattern, N = make New pattern, Q = Quit"; I$
1030  IF I$ = "A" OR I$ = "a" THEN GOTO 400
1040  LPRINT "SP;": IF I$ = "N" OR I$ = "n" THEN GOTO 200
1050  PRINT E$: SYSTEM
1095  REM:
1100  REM: INSTRUCTIONS
1110  PRINT HE$: PRINT
1120  PRINT "First an identification number for the pattern will be requested. Then"
1130  PRINT "you will be asked to specify the radius of a first gear, a second gear, and"
1140  PRINT "optionally a third gear. Next you will be asked to input a value for N, the"
1150  PRINT "radius of the pen and, if you have selected a third gear, the value of M."
1160  PRINT "N and M must be whole numbers or `vulgar` fractions. For whole numbers, just"
1170  PRINT "press the Return key when asked for the denominator. Next a step size will"
1180  PRINT "be requested. The smaller the step size, the smoother the curve.  The program"
1190  PRINT "will scale the pattern to the size of the Hewlett-Packard paper, but you can"
1200  PRINT "make it smaller by specifying a scale less than 1."
1210  PRINT
1220  PRINT "You have the option to print a label in the upper left hand corner of the paper"
1230  PRINT "which can contain either just the date and ID number or all the parameters."
1240  PRINT "Finally, a title may be specified, which will appear centred at the lower edge."
1250  PRINT "Press X to stop plotting."
1260  PRINT
1270  PRINT "Parameters may be transcribed from tables in the book 'Microcomputer Art' by"
1280  PRINT "Ross Edwards, which also explains their significance. Or you can generate your"
1290  PRINT "own values."
1300  PRINT: PRINT SPACE$(27) + "Press any key to continue."
1310  IF INKEY$ = "" THEN GOTO 1310
1320  PRINT HE$: RETURN
```

Free Small Ads

Apricot Sales

■■■ APRICOT Xi — 10MB, 720KB floppy, 256KB RAM; includes Internal MODEM plus generic COMMUNIQUE; carry case, WordStar, dGRAPH and WordStar. Best realistic offer secures. Best realistic offer secures any item. Call 06512 2667.

■■■ APRICOT Xi 10MB, 12in green monitor — hardly used — plus some software. £775. Call 01-441 3269 (evening) or 01-889 2751 (day).

■■■ APRICOT Xi 10MB; usual software plus PD utilities, Microtrivia game, Apricot User mags, 40 or so used floppies — £600 ono. Contact Tony Bayliss, 0952 630036 (weekdays, 9am-6pm).

Apricot Add-on Sales

■■■ 512KB MEMORY BOARD for PC/Xi, surplus to requirements. £100. Phone D Loebl, 01-642 5380.

■■■ QED+ integrated package (diary/calendar, project planner, address book, telephone list/dialer). Runs with other programs, hardly used — only £150. Also GSX USERS/PROGRAMMERS GUIDE — £10. Phone Robin Griffiths, 09323 48148 evenings.

■■■ REGISTERED SOFTWARE with full documentation — DATAMASTER v. 2.11; superb database, cost £495, sell for £150. PEGASUS sales and nominal ledgers plus stationery, £100 each. PRINTER (Epson FX80 insides) as new, £140. Also address labels and other-size floppies. Call Tony Bayliss, 0952 630036 (weekdays, 9am-6pm).

Other Sales

■■■ AMSTRAD DMP4000 132-column printer — still under warranty. Also seven Epson MX-80 RIBBONS. Best realistic offer secures any item. Call 06512 2667.

■■■ WATANABE WX4675 six-pen A3 flatbed plotter, little used. £450 plus VAT. Call B Stanier, Demic Scientific, 0740 20818 any time.

■■■ IBM CLONES supplies - PCs and ATs, 8-12MHz; prices from £750. Martin Filmer Computing Services. 01-761 0425.

■■■ APRICOT Xi 10MB; usual software plus PD utilities, Microtrivia game, Apricot User mags, 40 or so used floppies — £600 ono. Contact Tony Bayliss, 0952 630036 (weekdays, 9am-6pm).

Help

■■■ F SERIES user wants to get in touch with others in the States. Contact Bob Mandel, Lewis & Clark College, Box 96, Portland, OR 97219, USA.

■■■ PORTABLE user would like to contact another in South London to exchange information. Contact Geoff Reade, 9 Taymount Grange, London SE23 3UH.

Wants

■■■ WANTED — dead/damaged/cheap Xis or PCs. Phone Derek on 01-650 4900.

■■■ APRICOT XEN (non-IBM-compatible) hard disk system. Also wanted — a good C INTERPRETER (to avoid having to link after every re-edit) or Apricot CODEVIEW debugger; and a TELETEXT TUNER/READER that can be connected to Apricots (is there such a thing?). Call Tony Bridges, 01-948 1603 any time.

■■■ F10 10MB hard disk system (with manuals if possible). Contact R Noakes, 051-336 6898.

■■■ IBM-COMPATIBILITY FOR F SERIES — either a fully-effective IBM emulator or a patch permitting the IBM version of Microsoft Word 3.0 to run. Also wanted — a WORLD MAP or GLOBAL MAPPING PROGRAM. I have scores of home-grown GW-BASIC programs to sell, exchange or share — including full documentation and sample programs for the infra-red Mouse. Contact Bob Mandel, Lewis & Clark College, Box 96, Portland, OR 97219, USA.

■■■ SECONDHAND OR EX-DEMO HARD DISK — internal or external, 10 or 20MB, for PC. Cannot afford £800+ for a new one. Write to T Orekoya, FAO (FIRI), Via della Terme di Caracalla, 00100 Rome, Italy.

■■■ MMS INSTRUCTION MANUALS AND INTERFACE BOARD required for MMS hard disk unit and tape streamer. Willing to pay for help (even telephone assistance) in setting this up. Call Roy Roberts, 0923 775381 (office) or 0494 775088 (evenings).

■■■ GOOD DEBUGGER wanted for PC/Xi. I hate using Apricot's; can anyone help before I'm forced to write my own? (Preferably with symbols and/or C source display). Phone 01-948 1603 any time.

■■■ 10/20MB EXTERNAL HARD DISK required for Apricot PC with BIOS 2.4. G S Jayaram, PO Box 2642, Doha, Qatar (phone 417282 6-9pm UK time).

■■■ APRICOT PC or 10MB XI wanted; also PRINTER if available. Fair price paid, immediate cash. Phone Mrs Morrison, 0636 704208.

■■■ WORD PROCESSOR wanted for Apricot PC with single-sided disks — MS-WORD 3.0 preferred. Also ANY OTHER SOFTWARE. Phone 01-908 4631

Services

■■■ DISK RECOVERY: problems with deleted files, scrambled hard disks, and the like? We can (probably) help — no fix, no fee! Call Roger Gann at TP Group on 01-833 3501.

■■■ FILE TRANSFER: we can convert files between Apricot single- and double-sided, IBM PC format, PC/AT format, and Sirius — cheap! Call Roger Gann at TP Group on 01-833 3501.

■■■ DESKTOP PUBLISHING: we can produce your newsletters, price lists, press releases etc in near-typeset-quality on our laser printer using our DTP software NEWSWRITER from your disks in virtually any format produced by word processor and/or computer. Fast turnaround, guaranteed quality, low prices. You're reading an example of our work right now ... Call Roger Gann at TP Group on 01-833 3501.

■■■ HELP for PC/Xi/Xen users — setting up systems, MS-DOS, directory organisation, comms, modems. ALSO utility and business software written (in C, assembler or Basic). Call Tony Bridges, 01-948 1603 any time.

■■■ REPAIRS guaranteed. Martin Filmer Computing Services. 01-761 0425.

■■■ A0 PLOTTING — time on a Calcomp A0 plotter available for rent: Sirius or Apricot disks plotted on your paper or ours. Terms negotiable, but cheap. Phone James McEvoy on 01-859 4014 or 01-852 7981; or write to 16 Dunvegan Road, London SE19 1SA.

Use the service!

The Classified Ads service is free to all readers of APRICOT FILE. Just let us have a sheet of paper with the words (not too many); your ad will appear at least once in the newsletter and for around three months in the DEALS FOR READERS broadsheet.

Send your ad to us via electronic mail (Telecom Gold 81:JET767) or by land mail to:

APRICOT FILE
TP GROUP
PO Box 509
London N1 1YL.

A blunder in the Production Dept meant that last issue's piece by David Ingamells on producing sound from the Apricot included a couple of errors in the Pascal listings. We apologise mightily; anyone who wants the correct version is welcome to send us a formatted disk, and we'll dump the original on to it.

Datasolve Info expanded

The *Financial Times* has acquired **Datasolve**'s online information services — full-text press cuttings from a variety of 22 international publications, available direct or via services like Telecom Gold — for £10 million. That makes sense; the FT is in the information business, Datasolve is more a computer services company. The new owners have renamed it **Profile Information** and added the FTBR database, which itself covers 22 publications including the FT's own FinTech newsletters for the computer and petrogas industries.

Epson revamps

Epson has announced a wholesale revision of its 24-pin matrix printer line. The new **LQ850** and its wide-carriage sibling the **LQ1050** replace the earlier 800 and 1000 nine-pin letter-quality models; they get faster printing (NLQ at 73 rather than 60cps, draft at 220 not 180cps) and a novel paper-handling system that allows single sheets and continuous stationery to be loaded at the same time, with switch-selection. Prices remain the same, at **£625** and **£835** respectively.

Then there's a revamp of the top-end LQ2500, with the new **LQ2500+** getting improved noise levels (now below 55dBA) and better printing for multi-part sets. RRP is again unchanged at **£1,045**. For next year Epson promises a **sub-£500** letter-quality printer called **LQ500** — 50cps NLQ, 150cps draft. And available now Epson also has a new low-scan print scanner option rather like the Database product sold for the Amstrad PCW8256: Epson's **Image Scanner** fits four of its printers (EX800 and EX1000, at **£180**, LQ2500 and 2500+ at **£195**) and comes as a scanner that clips on to the print head. This picks up graphics being output, digitises it and passes them back to the computer for storage and subsequent use in another application. Epson doesn't see the product as competition for the high-price high-resolution OCR scanners currently on the market — "we've produced a low-cost scanner to service our printer user's needs"; and though it scans at a decent 25 inches a second, which is fast as scanners go, it scans *every* pixel — so a full A4 page can take up to ten minutes to scan. Limited, but potentially very handy.

Happy birthday, Vanguard

Vanguard, the joint Government and industry initiative set up to promote value-added networks, celebrated its first birthday a few weeks ago. The main plug has been improved efficiency from the elimination of "whole tiers of paperwork and administration"; Vanguard's own administrators say more than **10,000** organisations have requested information and "several hundred" have taken up offers of individual consultancy sessions. The role of Vanguard lies primarily in education, but it can also do some pump-priming to help new user groups begin studies of VANS application — and all such approaches are welcome. More from **Clive Young** at **DTI, Room 937, Kingsgate House, 66-74 Victoria Street, London SW1E 6SW**, phone **01-215 2627**.

Cheap Citizen

Printer distributor **Norbain** has dropped the price of the attractive little **Citizen 120D** by a third. This 125cps IBM- and Epson-compatible 80-column matrix printer now retails at a very competitive **£159**. The 120D prints at 120cps in draft, with an acceptable but not brilliant NLQ mode at 25cps. Norbain attributes the price cut to sales volume; the Amstrad market is presumably the target, but it should benefit anyone who needs a cheap, small, up-to-date printer; Norbain is on **0734 752201**.

Smart move

Unix specialist **Informix** is taking over **Innovative Software**, purveyor of the **Smart** series of loosely-integrated packages.

E-mail opinions?

The **National Computer Users' Forum** is on another of its information-seeking treks, this time inviting electronic mail users to comment on their experiences — especially problems with transfer of large files. Contact **Patrick Raymont**, NCUF secretary-general, at the NCC on **061-228 6333**.

Greens linked

GREENNET, the electronic mail network run by the **Green Party**, has acquired a London end to add to existing nodes in San Francisco and Costa Rica (the latter is the site of the UN's 'University for Peace'). It's certainly cheap enough — "we're not in the market for profit, and our charges reflect this"; GREENNET charges £5 a month subscription plus 6p a minute online charges. Info from **01-490 1510**. ◆

Come up and see us

As last year, TP Group will be dominating the Apricot-compatible world at the Which Computer Show (NEC, 19-22 January). Or, to put it another way, we have once again secured a bargain-price last-minute space in Islington Council's small IT corner. Sorry, we don't know the stand number; but seek out the farthest, gloomiest recesses that are closest to the bins and we'll probably be there. Come and chat, pick our brains, match faces to names.

And in the meantime, best wishes to all our readers for a Happy Christmas and appropriate New Year.