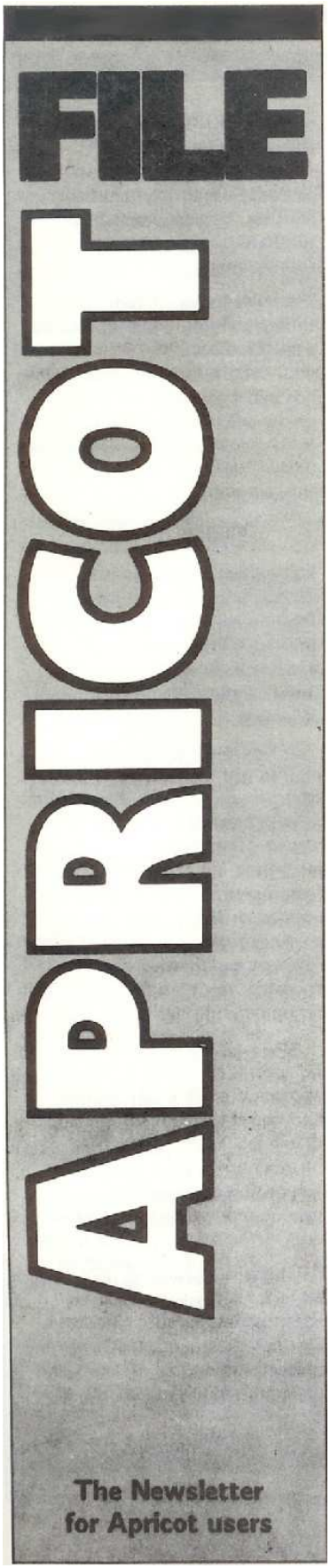
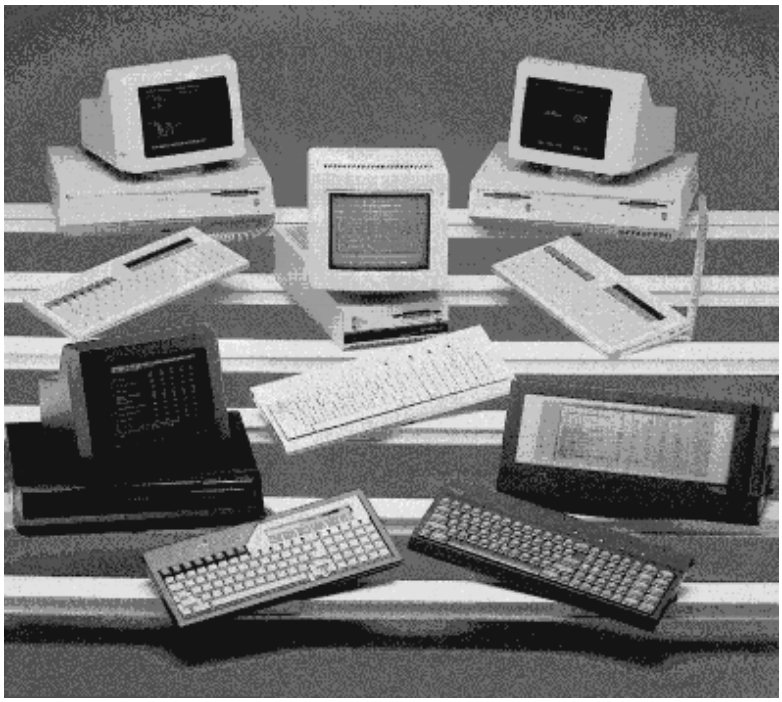


Apricot File 3.1



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 3.1

APRICOT FILE

The Newsletter for Apricot users

October 1987

No business like (computer) show business

The Tenth PCW Show reviewed

by Garret Keogh

With APRICOT FILE squashed in at the last moment, the Personal Computer World Show at London's Olympia last month was saved from being almost completely uninteresting. Still, between the usual bizarre items like Jeffery Archer launching a game version of one of his tomes and Chelsea Football Club opening the Commodore section at the Shed end, there were some things of genuine interest to the serious computer user.

For APRICOT FILE it was extremely useful. TP GROUP and Sierra Systems were practically alone in showing Apricot-related products, and there were a lot of forlorn-looking Apricot users at the Show. (Apricot itself doesn't go to many shows any more, and is waiting until the *Which Computer Show* to shower the world with goodies — IBM-compatible, of course).

Sierra Systems had its usual collection of cut-price Apricot software, the results of a pre-emptive strike on the remaindered stocks of Apricot's abortive North American adventure, and Sierra's **Geoff Wilkinson** reported business as being good — even for the mountain of US-oriented **pfs:** packages. Get his current price list from **03727 22890**.

Apricots in a corner

Our own star attraction was an exclusive preview of the **PC-286** upgrade we reviewed in APRICOT FILE Release 2.12. That attracted a lot of interest from subscribers some of whom came along specifically to see it. The PC-286 behaved reasonably well, once we'd replaced the curious gender-converting dongle that plugs into (and drops out of) the monitor socket. The general opinion? Most people were much impressed that it would read their old Apricot disks, not so impressed with the price — the Show was filled with sub-£1,300 286-based hard-disk machines — and the cost of converting to IBM-format software.

From the people who found our stand but hadn't been in touch before we got the same story we've been hearing for three years now. "I've got an F1/PC/Portable and it's a great machine, but Apricot/my dealer/software suppliers aren't very helpful". To which we could answer that their worries were over now that they had found APRICOT FILE ...

CONTINUED ON PAGE 2

In this issue

A Dream Shattered — A Riposte to Acid Drops 5
Batch File Recipes — Instant DELete, Tidy RAMdisk 6
Turbo Pascal Primer — Part One: A history 8
Comms on the Apricot — Communiqué in action 10
Writers in this issue 13
Readers' Writes 14
DEBUG for Beginners — Part Two: Advanced editing 16
Deals For Readers 23
WordPerfect's MicroScreen — LEDs and labels 24
Batch File Recipe — Disk tidy 25
Doodles by David — the occasionally essential column 26

PCW Show Report

CONTINUED FROM PAGE 1

We did however notice an interesting new approach in the self-effacement department. When we bounded forward eagerly with "Hello, do you have an Apricot?" the common response was "Well no, *I* haven't got one *personally*, but my friend (or lover, or workmate, or pet turtle) has". No need to be embarrassed — we've got them too!

Even without too many directly relevant products on show for the non-IBM-compatible Apricot user there still was a lot to see in terms of how the computer industry is progressing (or not, as the case may be) and along what lines. Olympia was divided into two sections for the home and business user, as before, and this ploy was successful in segregating the two largely incompatible audiences that have bedevilled previous PCW Shows.

Tour

Before looking at what was there, it is interesting to note who *wasn't*. As well as Apricot, those who deemed it safe to stay away included **IBM**, **Lotus**, **Compaq**, **Microsoft**, **Ashton Tate** and **Apple**. Some of the big-name companies will be waiting for the *IBM Show* in Islington (27 October to 6 November) and there's the *Desktop Publishing Show* too (15-17 October, again in Islington); but the major event, the *Which Computer Show* in January, will have them all — and us too, if we can sort out an economical stand. *Which Computer* is still *de rigeur* for business system vendors, despite PCW's success at keeping the games freaks out of Olympia 2.

In the main hall among the video-game nasties and the space-suited models, the **Amstrad** stand was continually packed for the 'new' **9512** word processor complete with 20cps daisywheel printer at £573.85 retail along with the enhanced **PC1640** IBM clone (EGA-compatible, 640KB, prices from £688.85 for the twin-disk monochrome-display model) and the competitive **LQ3500** 24-pin letter-quality matrix printer (160cps draft, 54cps NLQ, RRP just over £400).

This show being famous for games, Amstrad was also showing the latest versions of the **Spectrum** — yes, it's still soldiering on.

Which links us quite neatly with **Cambridge Computers**: Clive Sinclair's new **Z88** portable was on show — as a working model rather than the wooden mock-up reviewed in *PCW* magazine — and quite impressive it was too. Demonstrations showed Lotus 1-2-3 and WordStar files being downloaded into it.

Nearer the real world, the other big stands in the main hall included **Acorn** with the extremely impressive **Archimedes** RISC machine — blink and you would miss it. It needs some commercially produced software, but technology-wise this was the tops.

Also representing Real Life was the man from the **Data Protection Registrar**; with the cost of registration going up in November, the stand was suitably packed. Everyone came away with eight brand-new brochures explaining it all; and (not before time) there's a new registration form — simpler and shorter. If you think you might need to register, contact the DPR on **0625 535777**.

The **British Microcomputer Federation**, which you might have thought was an self-congratulatory industry body with little real aim in life other than to stomp on imports, also had a stand to introduce a new Code of Practice and a kind of computer-industry Kitemark. The seal of approval is apparently being tried by "most" of the big software houses on their products — Ashton Tate, Lotus and Microsoft being among the first to actually feature the symbol. What it will mean in practice will probably depend on how quickly it becomes industry standard and how important the companies themselves consider it. Hopefully it will offer some sort of fallback should you consider yourself screwed by some approved software.

The Code of Practice is intended to cover all aspects of the microcomputer business and adherence to the code will be a condition of membership to the BMF. Action is threatened against those in breach of the code. The Code is currently pending registration, though, and exact details are scanty. Watch this space for further details; or try BMF on **01-866 8339**.

Areas

This year's industry phrase is still **desktop publishing** and the PCW Show devoted a small section to it. We found this area rather unexciting; the industry still seems to be obsessed with the Macintosh and with page make-up rather than true 'publishing' (which should include editing, design and the like). It still looks rather immature, and that was reflected in the lack of originality in the stands on show — a pity, since this side of the computer business has a lot of potential. Most of the vendors are presumably holding back until the DTP show later this month.

As for the required components, there were no cheap laser printers with better than 300x300

resolution and generally the prices stayed high. Even the much-heralded **Atari** DTP was low key, buried in a corner of the hyperactive 'Atari Village'. Atari is still pushing cartridge-based games machines, though a package for the ST which emulates IBM is taking the Atari by storm. PC-DITTO does put the business possibilities of the IBM into the hands of the £299 ST user — but jeez, is it *slow!*

The ST also seems to be the machine for serious music users — some fairly impressive and all good fun. Everything from complete recording studios, samplers and sophisticated synthesisers were being run from the ST.

Of more general relevance was the first hand-held page scanner we've seen — again from Germany, but for the **Cameron Handy Scanner** there is a UK distributor. It really is hand-held, weighs 150g, and would enable you to scan and input documents and text for most IBM-compatible applications — directly to several big-name packages, but it also comes with its own input software. **Lightning** has the agency (**01-964 5555**) and the price will be around **£300**; that buys the scanner itself, a driver board (should fit a Xen-i), and the scanner package.

Back on more traditional lines, we expected low-cost networks to feature more prominently. But with IBM and Novell absent, it was left to the likes of **Nine Tiles** — which only risked a small stand for its multi-machine connectors. Still, **Sagesoft** was showing new low-cost LAN for IBM-compatibles under the name SAGENET; it costs a mere £299 for a two-station starter pack and £149 for each additional PC.

Sage is also trying to get in on the low quality of modern manuals with a series of PC learning software. Titles include PC Starter, PC Accounting, PC Touch Typist, PC Write, PC ChitChat, PC Planner and PC Retrieve. They all retail at £29.95. The astute will have spotted that these are all for Sage's own products; you might have thought that tutorial material ought to be included in the price of the package anyway ... Sage is on **091 284 7077**.

Sentinel Software was selling (or giving away — depends what time of day you arrived) an interesting tome called *A WordStar Survivors' Guide to WordPerfect*. Quite a good read, this, based around the kind of questions someone offered the choice of upgrading would ask. The show price was £1, though dealers will probably be giving it away. WordPerfect was running on the ST, Amiga and DEC MicroVAX as well as IBM-compatibles. Sentinel is on **0932 231164**.

Comms corners

For the comms user, **Micronet** was the only service on show — as enthusiastic as ever. That may have something to do with the rumours we've heard about Micronet finally achieving break-even, so its future now looks healthier after three years of 'market development' (which translates into 'making losses'). Micronet currently has over 20,000 subscribers, a new MD (John Tomany has just joined from a BICC subsidiary), and a new product — BIZZNET, an information service aimed at the small businessman. The service includes everything from on-line tax and legal advice to information on pension schemes and how to get those elusive shares. Along with the other services that Prestel offers such as Telex, e-mail and company search facilities, this does sound attractive; we'll be looking at BIZZNET in a future issue. Subscription to Micronet is £66 per year, which includes the Prestel sub. Details from **01-278 3143**.

The appeal of comms in general could do with some stimulus in the form of cheap V23 and V22 modems, but yet again we searched in vain for the £49.95 1200bps modem. A company new to us called **Dataphone** had some items of interest, though — its **Demon 2** has V21 V23 with auto-dial, auto-answer and automatic baud-rate sensing. There's a built-in power supply too, so you don't need a straggly lead to a clumsy 9V converter. And the price of **£75.95** looks very competitive. That spec plus a loudspeaker (or 'call progress monitor', as the manufacturers describe it) and a phone socket (the Demon has to have its own connection — you can't piggyback a phone from it) produces the **Designer** for **£99.95**.

The same supplier has a simplified **Dataphone** modem for the Amstrad 8256 at **£59.95**, and was also punting a low-price comms package for IBM-compatibles called KIWICHAT for **£39.95** (Amstrad PCWs) or **£49.95** (IBM-compatibles) — ASCII or XMODEM, with viewdata emulation including a clever split-screen Prestel facility that lets you leave one page on display while you open a window to look at another. Dataphone can be contacted on **0733 230240**.

Another new modem was from Swansea-based **Astracom**, a small firm which has had all kinds of hassles in waiting for BABT approval. The **1000 series** is a line of auto-dial auto-answer Hayes-compatible modems with automatic baud rate sensing and a Centronics printer port (which apart from anything else means the thing can be used as an RS-232-to-Centronics converter — there's a 6KB buffer inside too). The basic **AC1000D** V21/V23 model isn't dramatically cheap at £145, but the V22 version **AC1000T** is competitive at £245 and the V22 *bis* **AC1000Q** is £320. Chip-swap upgrades are

very good value by comparison with the likes of Miracle Technology — from 1000D to 1000T is £100, to 1000Q it's £175. Rather meanly, you're charged an extra £8 for a serial cable. More from 0792 473697.

Ahmos Computers, another company we'd never heard of, had some good-looking Taiwanese modems from Taiwan. The SmarTeam modems start with a V21/V23 model that isn't cheap (£225) but it is capable of Bell tones (so you can talk direct to the States). The V22 MiniTeam 1200 was at the same price, and a 2400 bps modem was £375. Expensive? Well, yes; but they appeared to be excellent machines. BABT approval is "imminent" and Ahmos is at 01-209 1300.

Also of interest to the comms user was **PCFAX-2** from **Basis**, a German company. This is a scanner, printer/plotter and facsimile machine all in one; the restriction is that it works only with the Atari ST. The whole device is portable and could be carried in a medium-size briefcase. Prices were not available but Basis can be contacted on 010 49 0251 719 975.

Dataflex Designs had the latest version of the Stradcom modem on show at various stands. In best Hollywood tradition it's known as **Stradcom 2**, but it does have some useful additions to the first and excellent model. It offers support for COM3 and COM4 as well as COM1 and COM2, has improved support for PSS links, and Dataflex says new chips and a degree of re-engineering make Stradcom 2 "the most reliable PC modem on the market, particularly for use in areas still served by the old Strowger-based exchanges" (where the amount of noise can befuddle less resilient modems). With the price maintained at £199 for a Hayes-compatible internal V21/V23 modem bundled with the ProComm-clone BAUDWALK, it's an excellent deal. Dataflex is on 01-543 7029; but our DEALS FOR READERS service will be offering Stradcom 2 (for Xen-i users only, of course) at £200 *including* VAT and P&P.

Miracle Technology was also there. No sign of the hinted-at LAN, but the internal **KeyCard 3000** modem has dropped in price (the V23 model is now £275, V22 is £425). More interesting is the **Modem Maximiser**; this beast sits by your modem and offers you EPAD error-correction, data compression, optional encryption, and full-speed buffering as well as a printer port. It's priced at £255 (or £324 with data encryption). Miracle is on 0473 216141.

Last words

As is often the case, the most interesting products were to be found on the smaller stands (those without Chelsea FC or leotarded models draped over the products). For instance, **Southdata** is a small British company we like. On show was the current version of its clever SUPERFILE database manager, a comprehensive data indexing system which offers speed and accuracy — and it's a generic MS-DOS program, so it can run on the Apricot. The RRP is £780, but we reckon this one is well worth looking into for file management applications; Southdata is on 01-995 7587.

And we found one of those Good Ideas that all computer users can use: **The Thingi** is just a strip of plastic with a catch on it. It sticks to the top of your monitor, and you can attach documents via the clip so that they're easier to read. It costs £6.99 from CPM Ltd on 051-647 8616. They were all over the place — and so was the **Plonker Box**, without a doubt the most obvious and talked about peripheral at the show — it's a small plastic tray that sits on the side of your computer and holds your current floppies. A masterpiece of modern technology, a wonder of modern science, and — since it was perhaps the most useful and innovative item at the show — a worrying sign for the state of the industry. (Even we're selling it; see the DEALS FOR READERS catalogue).

Part of the razmatazz of the show was the **British Micro Computing Awards**, presented at a famously boozy dinner for which we failed to wangle an invite. Perhaps Apricot might have won for customer satisfaction and courtesy? Maybe a special award for the designers of System X? Sadly not. *Business Micro of the year* was the **Tandon Pac 286**, a removable hard disk unit which could also qualify for the 'Least Available Micro' award. *Business Software of the Year* was **Adobe Illustrator** from **McQueen**, an add-on for the PAGEMAKER DTP package (is that *business* software?). No quibbles about the *Small Business Micro of the Year*: the **Acorn Archimedes 300** was the only real contender. And **Ability** probably deserves the acclaim of the *Small Business Software* award for the **Ability II** database package — neat, clean, cheap, versatile. Special awards went to **Ventura Publisher** and the **Amstrad PC1512** for no good reason we could see.

So there were no great surprises in the awards list; and that seemed to reflect the show as a whole. Perhaps the most interesting part of the show was just across from our stand in the Taiwanese section — copies of nearly every machine on the market, including IBM clones in Macintosh-style casing (very odd), some superfast 386-based machines, and 20Mhz 286s for a fraction of the price. For some reason, however, the wily Orientals have still not found the need to clone the F Series; I wonder why? ◆

A Shattered Dream

A Reader's Response to Acid Drops

By Brian Read

In what follows, all the lines of Author Langford are quoted from his *Disinformation Column* 'On acid drops' in APRICOT FILE Release 2.11.

Plain Reader: WordPerfect's had a good press lately, David. From you and others I had concluded it's a superb word processor. I've become a SuperWriter addict, but recently I'd been telling myself that one day I'd get WordPerfect ...

Author Langford: It's coruscatingly wonderful ... every previous APRICOT FILE reviewer of WordPerfect has supplied fulsome praise ... very superior indeed ... has vast merits ...

Plain Reader: That's what you say in your piece 'On acid drops' in Release 2.11. But you don't tell us in detail what it will actually do that SuperWriter won't.

Author Langford: We have a working thesaurus ...

Plain Reader: But who wants it? Roget's is easy and quick to use and comes in a small hard rectangular format known as a book. Very handy and cheap — and it can be slightly edited, unlike the WordPerfect effort.

Author Langford: Most standard functions are handled with such brilliance ...

Plain Reader: Go on, tell me more. How can SuperWriter's functions such as search-and-replace, boilerplating, underlining and so on be improved? They seem fine to me.

Author Langford: I'm still devoted to WordPerfect ... now here I am with the acid drops.

Plain Reader: Well, I've read them. You amaze me. You say it can't even write more than one page of parallel columns like this script I'm writing now. You will note that this is perfectly set out in script format. It's done entirely with SuperWriter. *(Well, as emulated by* NEWSWRITER *for the purposes of presentation in* APRICOT FILE *— Ed.)*

Author Langford: Gee whiz. Unfortunately (in WordPerfect) the whole system collapses when a soft page break falls ...

Plain Reader: Well, SuperWriter manages this easily. Mind you, it does have a few small foibles in some minor functions. I've had a dream that one day I'd get perfection in WordPerfect, but now you've shattered my hopes. I shall stay with SuperWriter.

Author Langford: No software is perfect, but this word processor comes closest.

Plain Reader: I find that hard to believe, but if you send me a free copy of WordPerfect and I am convinced that it really is better I'll start to use it. But I'd still have to use SuperWriter for scripts like this, wouldn't I?

Setting out a script like this is very quick and simple using SuperWriter, but if you haven't done it before you may get rather bogged down by trying to find the instructions in the manual. The following is the method I used: what SuperWriter calls an 'embedded formatting command' was inserted for an indent (17 in this case) to create the second column. To get the names to appear on the left you insert a negative paragraph indent of -17. So after the title I typed ...

```
in17.pi-17
```

At the beginning of each speech I needed to ensure that the first letter would line up with the rest of the speech, so I inserted ...

```
tab18
```

This wasn't so laborious as it sounds because you can speed everything up by using one character (I used a plus sign, but any spare character would do) and then use SuperWriter's search-and-replace function to substitute the correct command. And of course you can also economise with your typing labour by using search-and-replace to avoid typing the names in the first column in full at the beginning of each speech. ◆

Batch File Recipes:

Instant DELete, Tidy RAMdisk

By Simon Potter

This batch file automatically deletes all files on a particular disk, without giving you the chance to change your mind via an 'Are you sure?' message. It's obviously very dangerous, so make sure you're confident about using it — I wrote it to mass-delete from a heap of old floppies and to tidy up RAMdisks.

```
REM DELY.BAT
ECHO OFF
CLS
ECHO
ECHO
ECHO  Warning! This automatically deletes
      everything on drive %1
ASK   Are you sure you want to proceed?
IF ERRORLEVEL 1 GOTO END
DEL %1:*.* < Y > NUL
:END
CLS
```

DELY.BAT is used with a drive name; so DELY A deletes everything on drive A (don't try it!). Things start safely enough:

```
ECHO OFF
CLS
ECHO
ECHO
```

The first two lines are the best way to start *any* batch file; ECHO OFF prevents the batch file's messages appearing on the screen, CLS wipes the display and gives you a clean screen. Then there are a couple of blank lines — when you type each of those two ECHOes, follow the command with a couple of spaces.

```
ECHO  Warning! This automatically deletes
      everything on drive %1
ASK   Are you sure you want to proceed?
IF ERRORLEVEL 1 GOTO END
```

The Public Domain utility ASK is to be found on DEALS FOR READERS disk AP001; when the batch file is run, you're ASKed the question in the text following the command and you can then have your batch file take appropriate action depending on the status of the ERRORLEVEL flag — it's set to 0 if the user enters 'Y', to 1 if 'N' is typed.

So the ECHO puts the Dire Warning on to the screen (the percent-sign parameter is replaced by the drive name you entered when you ran DELY), you're ASKed if you know what you're doing, and if you answer 'N' the deletion is skipped and you go to the end. Otherwise, you proceed to the operative bit:

```
DEL %1:*.* < Y > NUL
```

This is the key line — it DELetes all files on the drive you've specified. The %1 is replaced by the drive identifier you gave, and a colon and the star-dot-star wildcard is appended.

Note the use of piping here — the 'input' for the command, which MS-DOS would normally expect from the keyboard in response to that message, is instead taken from a file called Y located on drive A in the root directory. Y is a one-liner containing simply the letter 'Y' and a Return; so it provides the 'yes' response to 'Are you sure'.

And the 'output' of the DEL command is directed away from the screen and into the non-existent dummy 'file' NUL. That's because the 'Are you sure?' message appears even when ECHO is set to OFF.

That's it. The program winds up with a label (to which it jumps if you decide against deleting everything) and a tidying-up CLS.

```
:END
CLS
```

Safe and tidy RAMdisk

This one provides a fairly secure method of using RAMdisk. Well, at least it forces you to save anything inadvertently left in RAMdisk. Run your application from a batch file that includes these lines — so instead of typing 'SW' to start SuperWriter, you'd have this batch file under the name SW-GO.BAT or whatever, and include the SW command in it. Then you'd run SuperWriter by entering 'SW-GO'; and because it's much faster, you would save all your work on RAMdisk rather than floppy.

```
REM RAMDTIDY.BAT
ECHO OFF
CLS
:YOURBITS
REM  Run your applications here, saving work on
     RAMdisk — assumed to be drive C
CLS
A:
CD A:\
IF NOT EXIST C:*.* GOTO FINAL
IF NOT EXIST C:*.BAK GOTO NEXT
DEL C:*.BAK
:NEXT
ECHO  You have these files in RAMdisk:
ECHO
DIR C:
ECHO
ASK  Do you want to copy them to drive B?
IF ERRORLEVEL 1 GOTO END
CLS
KNEW C: B:
:END
ECHO
ASK  Do you want to delete them all from drive
     D?
IF ERRORLEVEL 1 GOTO FINAL
DELY C
:FINAL
CLS
TYPE MENU.TXT
```

This one makes use of a couple of Public Domain programs and a few simple batch-file tricks. To take it line by line:

```
ECHO OFF
CLS
```

Well, all batch files should start with those two commands.

```
:YOURBITS
REM  Run your applications here, saving work on
     RAMdisk — assumed to be drive C
CLS
A:
CD A:\
```

Instead of the REM, you'd have the command 'SW' to run SuperWriter, or 'SC3' to run SuperCalc, or whatever.

I'm assuming that your RAMdisk is set up as drive C. If not, change all references to 'C' in the rest of the batch file. When you've finished your application — running SuperWriter or whatever — the batch file cleans up by clearing the screen, then makes sure you're on drive A and in the root directory.

```
IF NOT EXIST C:*.* GOTO FINAL
```

This checks to see whether there are actually any files at all in RAMdisk. If there are none, it skips to the end of the batch file.

```
IF NOT EXIST C:*.BAK GOTO NEXT
DEL C:*.BAK
```

These two lines will prevents you copying BAK files; I assume that you want to discard these, on the grounds that you presumably have more up-to-date versions of the files in RAMdisk, but if you'd prefer to hang on to them just omit these two lines. The IF NOT EXIST skips you on to the next section if there aren't any .BAK files, and the DEL line is executed to erase them if there are some BAKs present.

If you might want to retain the BAK files, an alternative to these lines could go something like ...

```
IF NOT EXIST C:*.BAK GOTO NEXT
ASK  Do you want to delete them?
IF ERRORLEVEL 1 GOTO NEXT
DEL C:*.BAK
```

This uses ASK to have the user confirm whether the BAK files should be deleted. The IF line checks the ERRORLEVEL flag, and skips the DELete if you typed 'N'.

```
:NEXT
ECHO  You have these files in RAMdisk:
ECHO
DIR C:
ECHO
```

Either way, you get to the section labelled :NEXT. This just tells you what you actually have in RAMdisk, preceded by an ECHOed message. I prefer to 'page' my DIRectory listings with the /P switch, such that the listing pauses after a screenful has appeared; you may prefer to use /W for a 'wide' display. The two spare ECHOes around the DIR command should be followed by a couple of spaces when the batch file is entered; the ECHO then 'prints' a blank line on the screen.

```
ASK  Do you want to copy them to drive B?
IF ERRORLEVEL 1 GOTO END
CLS
KNEW C: B:
```

Another ASK checks whether you do in fact want to copy them — and I'm assuming that you'll be copying to a work disk in the floppy drive. If you don't want to copy, the program skips to the end; otherwise it tidies up with CLS and runs the second Public Domain utility, KNEW (on DEALS FOR READERS disk AP002). KNEW copies only 'new' files; that means for each file in the source drive (the RAMdisk, in this case) it will check for the presence of a file of the same name on the destination disk (the floppy in drive B). It won't copy a file if the destination disk has one of the same name with a more recent date and/or time of creation.

Actually, KNEW probably isn't necessary in this case — since you've just been using RAMdisk, it's not likely that the floppy disk will have any newer files of the same names. But KNEW is a good habit to get into.

The KNEW line would then be replaced with something like ...

```
COPYQ C:*.* B:
```

That way, for each file in the RAMdisk you'd be asked whether the copy should be done. So you'd save only the files you really need rather than all the temporary work files you've been creating along the way.

```
:END
ECHO
ASK  Do you want to delete them all from drive
     C?
IF ERRORLEVEL 1 GOTO FINAL
DELY C
```

Yet another ASK (it's amazingly useful) invites you to scrub everything from RAMdisk. You might want to do that to prevent the RAMdisk filling up with dross when you run your next application. If you answer 'N', the program skips the DELete line; otherwise it uses my DELY.BAT utility to delete everything from the RAMdisk, drive C..

If you can't be bothered with DELY.BAT, you could replace that line with the essential bit:

```
DEL C:*.* < A:\Y
```

And that's just about it. The last few lines just clear the screen and put on it a multi-choice option menu something like the idea I discussed in APRICOT FILE Release 2.7.

```
:FINAL
CLS
TYPE MENU.TXT
```

Contribute!

Why not share your own useful utilities? Let us have your own home-brewed batch files, and we'll write them up for this *Recipe* series. Just us a line at APRICOT FILE, PO Box 509, London N1 1YL.

A Turbo Pascal Primer

Part One: A brief and biased history of programming and programmers

By Mark Whitehorn

In the beginning there was void. Then ... there were computers. Shortly after that there were computer languages. And finally there were programmers.

These programmers were initially recruited from the ranks of mathematics students because it was felt that all computer programs were going to be about sums. They had to program in excruciatingly abstruse languages; and their work was difficult and little understood by the people around them. Hence they became highly respected and young ladies threw themselves at their feet (female programmers were illegal at the time).

Later, languages such as Fortran and Cobol were developed. These were easier to use, but were still rather cranky; so programmers retained the respect and admiration of those around them.

Then came Basic.

Shock horror

This was really a bit of a blow. The new language was rather easy to use; worse, it was supplied free with the growing multitude of micros which was sweeping the country. This meant that ordinary people could get their hands on a computer language, and could write their own programs. And they did.

"Oh dear" thought the programmers. "There goes all our status." They were resourceful, however. They got together and decided that, for a variety of reasons, Basic wasn't a *real* computer language. There then followed a remarkably successful smear campaign in which Basic was thoroughly discredited.

Still, many people learned to program in Basic and were perfectly happy with the results, though they dared not admit this in public. The programmers thought this to be an excellent state of affairs and bragged about C (which few of them actually understood).

But time was running out for the programmers. Eventually Nemesis arrived in the form of Turbo Pascal. Here was a cheap, easy to use, easy to learn, compilable, structured programming language which was available to the general public. "Foul", cried the programmers, as the Basic-users came out of the closet clutching £50 copies of Borland's magic box. Suddenly the shame of producing BAS files was gone forever, replaced by the thrill of real COM files. Not only that, but the language was so fast that the old Basic-users had time for a social life; and they too began polishing their shoes before parties so that nubile members of the opposite sex would have a clean surface upon which to fall.

Such is the state of what we might call 'popular programming' today. There are many competent Basic programmers who are thinking of switching (or have just switched) to Pascal.

Should you really leave Basic?

Despite what some programmers say, Basic is a good language in many ways. It was originally designed to teach people how to program — hence the name (Beginners' All-purpose Symbolic Instruction Code). As such, it worked rather well; and eventually it succeeded beyond its creators' wildest dreams by becoming so popular.

Then why is it maligned? Well, apart from the paranoia described above, it does have some problems.

- **Compilation vs interpretation:** Basic is most commonly offered as an interpreted language. In practical terms this means that large programs can be slow in execution, and that it is difficult to produce standalone programs (normally the Basic language must itself be loaded before the program is run).
- **Modular design:** Many offerings of Basic do not allow a program to be built up in linked modules. Small programs in any language are usually easy to understand; but when programs become large, most humans have trouble understanding what is going on. Worse still, the programmer may be able to remember what is going on during the creation of the program — but understanding it six months later can be next to impossible. This is where modular design helps. Modular design means splitting the program into lots of little bits, each of which is small enough to be understood by the average brain. These bits are then strung together to form a whole.
- **Control Structures:** The flow through a Basic program is often controlled by the use of the instruction GOTO, usually allied with an IF. This is fine in small programs:

```
10 PRINT "The Penguin is a brave and noble bird"
20 A = A +1
30 IF A < 25 then GOTO 10
40 END
```

This program is perfectly readable. In large programs, however, the indiscriminate use of GOTOs can lead to so-called 'Spaghetti programs', where the flow through the program becomes so complex that a degree in cartography is required to unravel it all. Most implementations of Basic do not provide enough alternative commands for flow control, so the programmer is left with little alternative but GOTO.

The most recent versions of Basic from Borland and Microsoft, and the new BBC Basic 5, all overcome most of these failings. What seems to be happening is that Basic is becoming more and more like Pascal.

Still, Turbo Pascal is the language in which I choose to program. This is for a variety of reasons, the main two being that developing code is fast and the editor is so friendly (especially with David Langford's TURBOKEYS keyboard in control of the keys — that's DEALS FOR READERS disk AP003 at £15).

Under no circumstances should you believe anyone who tells you that if you have learnt Basic you'll have a tough time with a *real* language — "You see, you've picked up bad habits (unspecified) so you will probably never be a good programmer now". This is scared-programmer talk. My experience is that Basic programmers pick up Pascal more rapidly than non-programmers — and so they should. They already know about variables, saving files, input and output, disks, and all the things which are common to programming.

This series of articles is not designed to teach you Pascal; Borland provides an excellent tutor for that purpose as part of the Turbo Pascal package. Instead it is for those who have got past the initial steps but are still finding their feet. My aim is to provide worked examples of handy utilities which illustrate how the language can be used.

I shall be stressing the production of *usable* code. By this I mean programs which are readable and easily updated. This involves Modular Programming (see above) and Top Down Programming (see below), but it also means writing simply and clearly (see here). Some programmers rejoice in writing highly compact, frightfully clever code which is virtually unreadable. To some extent this attitude is a hangover from the days when micros were slow and had 8K of RAM, so programs *had* to be compact. Now we all have at least 256K and machines are much faster; we're no longer concerned in saving the odd 300 bytes, unless of course the program has to run in real time, with the accompanying real costs. The readability of our programs is more important.

With this in mind, I will try to avoid short complex solutions to problems when slightly longer but more comprehensible alternatives are available. I know this will offend some purists, but I remain convinced that I am right. No-one has yet complained about the length or speed of the programs that I write — but they *always* change the original specifications three months after getting the program, and want the update performed immediately.

What should we write?

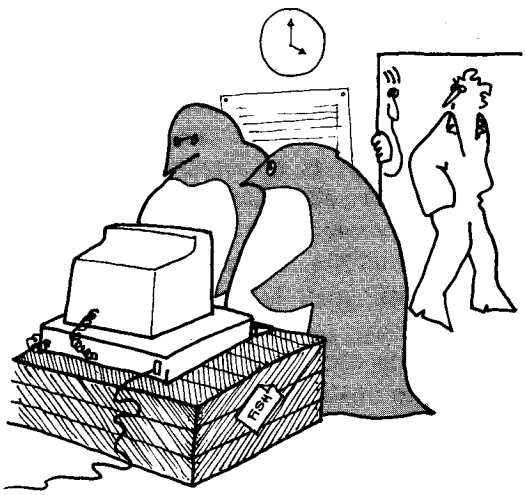
The ability to manipulate files has proved to be of great use to me. Changing word processor documents, stripping out control codes, encrypting files — all manner of doors open if you can play with the characters in a file. So a general-purpose file manipulation utility will do for a worked example. And one advantage of this sort of utility is that it can be modified time and time again (modular programming is a boon here) for different tasks.

The first illustration will simply write the characters to the screen, just like the MS-DOS command TYPE. This illustrates the power of Pascal — we can in effect write part of the operating system!

First, let's have a bit of Top-Down programming — in other words, planning what I want the program to do in easy stages:

1. Open the file for reading
2. Remove the characters one by one
3. Manipulate them in some way (in this case, write them to the screen)
4. Close the file

Now each part is gradually expanded until the program has written itself — well, that's the idea. I'll demonstrate this in next month's exciting episode, but try writing it for yourself in the meantime; and if you have not yet begun to learn Pascal you have a whole month in which to get started! ◆



"They aren't all that clever — they're still programming in MS-Basic."

The APRICOT FILE DEALS FOR READERS service can supply Turbo Pascal itself in Apricot format as Order No AS048 at £69 (VAT and despatch included, as on all DEALS FOR READERS disks); keen users might also consider VICAR (AS028, £56) to keep track of their programming.

We also have a new disk, AP023, which provides a Pascal Tutor and Compiler at £15; this is a double-sided disk with an eleven-chapter tutorial supported by over 60 illustrative programs and a full-scale Pascal compiler.

Communiqué in action

Automating Communiqué with the Apricot internal modem

By Paul Lavin

Comms programs generally fall into one of two categories. One variety is typified by CrossTalk and Mirror; they're designed for the user that wants to fiddle around comprehensively with the comms parameters. The other kind of comms package is for those who don't care about baud, parity and stop bits and furthermore may not want to know at all.

Apricot's Communiqué is the perfect illustration of that group. Originally cobbled together for the old-style non-IBM-compatible Apricots to provide a reason for using the Apricot internal modem, changes in philosophy at Apricot have lead to the package being made available to the wide and wonderful world of IBMs and clones.

Communiqué is more than just another menu-driven comms program: it's also a database of on-line services and their connection details. Currently, it supports a dozen services including Datastar, Extel Priceline, and Pergamon Infoline as well as the ever-popular e-mail services from Mercury Link 7500 (formerly more succinctly named EasyLink) and Telecom Gold. Updates for new services are provided in an accompanying folder as well as the usual deals for free or cheap subscriptions to on-line services.

Other features of Communiqué include a viewdata screen manager for Prestel addicts (not for Xen-i users, though), EPAD error correction as used by PSS, and encryption for those with secrets to keep. The current versions are 2.2 for Apricots and 3.0 for IBM-compatibles.

Modem digression

Communiqué is still tied specifically to the Apricot internal modem. The modem still sports the now-superseded Apricot bus connector but can be snugly accommodated in an IBM by the use of an adaptor card; Apricot threatens to bring out a Xen-i version that won't require the additional expenditure of fifty quid, but things move slowly in Brummieland.

Examples of the old-style Apricot modem are being flogged by **Sierra Systems** — phone 03727 22890 — for £75, against an RRP of £145 (*you'll have to get Communiqué itself from another source, though — Ed.*) The modem is worth considering at that price; my experience over the last year leads me to believe that they are reliable little beasts which rarely fail to latch on to the squeaks of the remote computer's modem.

Automating Communiqué

Communiqué is not however a complete solution if you want to keep your hands as clean of computer dirt as possible. A description of the personnel at Anonymous Ltd, a company that I consult with, may give you a little insight into why I invented the three-key method of PC communicating.

Richard is a Very Important Businessperson. Lunch with the Aga Kahn, condo in Thailand, big deals in the City. Name a pie and Richard has his finger in it. He has little time or patience to use a computer or comms. In fact, poor man, he is still using WordStar because he hasn't the time needed to learn anything better.

The Hon Mrs Olaf is the busiest person I know. She fits at least 26 hours on the telephone into every working day. She has endless appointments. And her management reports are always late.

Erik is an artsy-fartsy Luddite. After several close calls with various IBM and Apricot machines, he usually sticks to pen and paper unless provoked.

Sophie is pretty smart, ok? She can answer a telephone on the tenth ring as long as it's on the desk next to her. She can use the word processor, just, but often forgets to save her work. She lives to sail and go shopping.

In such an office, the comms package has not only to be menu-driven but must be largely automatic as well. By the artful use of batch files plus the macro facility of WordPerfect and the good offices of Communiqué, I was able to contrive a comms solution so quick to learn and so easy to use that it can be mastered even by large invertebrates including several species of giant squid.

What follows is an outline of the implementation generated to meet the specific needs of a small management consultancy company. If it's close to something you need, I will be glad to e-mail you the relevant details (less passwords!) if you contact me on Telecom Gold mailbox 72:MAG80012. Otherwise, you are free to emulate my method to achieve your own distinctive style of madness.

The comms facilities required by Anonymous Ltd centre around e-mail messaging to either or both of its US offices in New York and Boulder, Colorado; Telexes to Paris; and the ability to put messages into the company Mercury Link mailbox for each other. There is only the occasional need to send Telexes elsewhere.

The advantages of having Mercury Link call and download messages periodically was discussed; but no one wanted to be responsible for making sure that the system was ready — computer turned on, paper in the printer, etc. So messages had to be downloaded on demand.

Both paper and disk copies of incoming messages were required, and an archive of outgoing messages was also to be maintained.

Procedures

Communiqué was installed into a subdirectory on the hard disk using it's own install program. Company policy dictated that the messages and Telexes to be sent were located in drive B in a subdirectory called FILES. To kick the whole thing off, a batch file called TELEX.BAT was created to put this on the screen:

```
                Mercury Link Menu

Type A      TO RECEIVE MERCURY LINK MESSAGES
     B              TO SEND A MESSAGE TO NY
     C            TO SEND A MESSAGE TO BOULDER
     D      TO SEND A MESSAGE TO NY AND BOULDER
     E             TO SEND A MESSAGE TO PARIS
     F          TO SEND A MESSAGE ANYWHERE ELSE
     G      TO RECEIVE MESSAGES AUTOMATICALLY
     Q                    TO QUIT THIS MENU
```

WordPerfect was actually used to construct the text screens and allow the use of snazzy borders which won't reproduce here. You could use EDLIN or another WP package in ASCII mode, too (*SuperWriter and David Langford's* BORDER.XQT *file from Ansible or* DEALS FOR READERS *could be fun, too — Ed.*)

This method leaves the user with a functioning A> prompt. Some might consider this the moral equivalent of giving razor blades to babies; but this crowd of users wouldn't know what to do with an A> prompt, much less type the deadly FORMAT C: or whatever. Still, for safety FORMAT.EXE was renamed FORMATER.EXE. Format is called from a batch file that only allows FORMAT B: — which could be bad enough.

As you might reasonably anticipate, batch files perform the necessary actions. This is A.BAT, which sets up Communiqué to read the Mercury Link mailbox:

```
ECHO OFF
CD COMMUNIQ
CD B:\FILES
REN B:TLXRECD TLXRECD.BAK START1
```

Appropriate reminder screens are displayed about switching the printer on, verifying its on-line status, and checking paper availability. The message archive file name is changed in preparation for the new material. Then one of a series of START batch files is called; more on this later.

Batch files B.BAT to E.BAT are all pretty much of a muchness but take the following format:

```
ECHO OFF
CLS
TYPE TLXSCRN.NY
TYPE BLANK
PAUSE
CD WP
WP/M NY
CD \
CD B:FILES
CD COMMUNIQ
START2
```

The file TLXSCRN.NY displays this message in a form suitably customised for each destination:

```
IF YOU HAVE NOT PUT THE MESSAGE FOR THE NEW YORK
OFFICE IN THE \FILES DIRECTORY ON THE DISK IN
DRIVE B, HIT CONTROL-C TO ABANDON THIS PROCEDURE.
OTHERWISE...
```

The file BLANK, which is typed to the screen contains nothing more than a single Return — produces a blank line, geddit? This rather circuitous route was necessitated by the revision in MS-DOS 3 that disabled the good old ECHO [space] [space] method of generating a blank line.

The success of this comms adventure depends on the ability of the users to name their Telex file appropriately for the desired destination. If they got that screwed up, all hope was lost.

Macros

Batch files B to E continue in their execution by starting up WordPerfect using a macro unique to each destination. The sole purpose of the macro is to convert the WordPerfect format file into an ASCII file; this embellishment was added after several unconverted reports infested with control characters were sent to New York to the puzzlement and embarrassment of all concerned.

The contents of the macros can be summarised thus:

- retrieve B:/FILES/TLXNY (or as appropriate)
- hit Text In/Out
- save Text in File named b:/files/tlxny (or whatever)
- hit Y (to replace existing file)
- exit WordPerfect

This job could well be accomplished by an external program that strips off the high bit so often used by word processors to perform their *leger de main.*

After a little drive-logging and subdirectory-changing, batch files B to E all kick off a START batch file. F.BAT is a little different; because its job is to send a Telex to God knows where, no automated WordPerfect conversion utility is practical. The sender must therefore shoulder the responsibility of making sure it is in ASCII form before transmission, and TLXSCRN2 reminds them of this:

```
YOU MUST BE SURE THAT YOU HAVE CONVERTED ANY
MESSAGE TO BE SENT TO ASCII FORMAT USING
WORDPERFECT'S TEXT IN/OUT FEATURE. HIT CONTROL-C
TO ABANDON THIS PROCEDURE IF YOU HAVE NOT DONE
THAT.
```

Q.BAT merely displays the message that this feature is not implemented yet (it's awaiting personnel commitment to vital details). Q.BAT unceremoniously clears the screen; an important part of user friendly software configuration is providing a convenient exit, *n'est-ce pas*?

If Richard *et al* pluck up enough courage to press one of A to F and a Return, the computer will surge into life by running one of the START files located in the Communiqué subdirectory. These will clear the screen and run GSX and PRESCHRS in preparation for the main event. Here's a sample:

```
ECHO OFF
PRESCHRS
GSX
ACCESS %1 %2 %3 %4 %5 %6 %7 %8 %9
REM After exiting from Communiqué, the archive
file
REM is updated and the transmitted file is erased
COPY B:TLXNY +B:ARCHIVE.TLX B:TEMP >NUL
DEL B:TLXNY.*
DEL B:ARCHIVE.TLX
REN B:TEMP ARCHIVE.TLX
CD B:\
CD \
REM Returning to the TELEX menu ...
TELEX
```

Communiqué is started by ACCESS followed by a percentage-sign parameter. The intrepid user is then confronted by the Communiqué opening menu offering four choices. By careful tutoring, no-one dares to choose anything other than the first alternative, the one that gets them the connection menu.

Indulge me in a brief aside. There is nothing more software reviewers love to boast about than the discovery of undocumented features. Communiqué is a veritable goldmine of them; the manual is so abysmally written that literally half the stuff that Communiqué does is documented not at all or so poorly that the user on the Clapham omnibus is left completely in the dark.

In fairness, the people at Apricot Software will spill the beans. I had to call Apricot Software because my dealer couldn't answer my questions — since I was "pushing the limits" of the package I couldn't expect it to perform as documented. Thanks, Concept Computers of High Holborn. Good job. *(We hear they're no longer Apricot Dealers — Ed.)*

Because Communiqué comes pre-configured to give you access to a plethora of on-line services, you will need to hide the connection menu that Communiqué produces by default. In your installed Communiqué you will find a file called ACCESS.PAR. Rename it ACCESS1.PAR and create a new ACCESS.PAR by using choice two on the Communiqué initial menu, Change Parameters, filling in the blanks as required.

This allows you to present your busy/simple users with your own carefully contrived list of alternatives that will cater to their whims without their worrying about what the Dun & Bradstreet on-line database was saying about their corporate credit rating.

Once a choice is made from the connection menu, Communiqué will automatically download your mail or send Telexes/e-mail and then disconnect. A total of three keystrokes (not counting Returns) is all that is needed to accomplish a large proportion of Anonymous's business intercourse. To keep things simple and as fast as possible, you can't download and send in the same session.

On exiting from Communiqué, the remainder of the START batch files tidy up the archive files and return you to the first menu, TELEX.BAT.

Scripts

The facility that enables Communiqué to send your message to New York or Paris is built in via script files. These are no stranger to comms packages generally; they allow you to specify default settings and drive the software as though someone intelligent was seated at the console pressing the buttons. Communiqué has two flavours of script files: 'sign-on' (filename extension SGN) and 'control' (extension CTL).

The sign-on files, constructed with any word processor in ASCII or with EDLIN, are literally scripted dialogues that will take place once a connection has been established. Communiqué will wait until whatever is on the first line of the sign-on file appears on the screen before it transmits the contents of the second line.

So when Mercury Link prompts 'ID?', Communiqué can reply with the terminal type, user identification [I] and password [P] needed to sign on. The ID and password are read directly from the sign-on file created using the Change Parameters choice from the initial Communiqué menu.

The process is repeated alternating line received with line transmitted. There is a timeout set in case line corruption interferes with accurate communication; Communiqué will wait one minute then try again. Here is an example:

```
ID?
16 [I] [P]
PTS
[CNTL, EASYNY.CTL]
```

For the system I devised for Anonymous, the last line in each of the respective sign-on file handed control over to a unique CTL file for each connection menu choice once the 'PTS' prompt was received from Mercury Link. The control file turns on the printer (or dumps to disk in the mail-reading choice), inputs the address, and uploads the file indicated in the 'send message' options. As an example, this is the control file for sending a message to New York:

```
[PRINT]
/NTF
123456789+
[T3]
[LOAD,B:TLXNY]
LLLL
[T3]
[MENU]
```

Instructions to Communiqué in a control file need to be contained in square brackets. [T3], for instance, causes a three second pause. Upper-case letters are *de rigeur*. Characters to be sent to the e-mail service are entered just as you would at the console, Returns and all (and not as ASCII values enclosed in brackets, as the manual would have it).

The F.BAT file that allows a Telex to be sent anywhere triggers a START file that can sign on to Mercury Link but doesn't have an associated CTL file. Those daring enough to tread the path to exotic destinations will need to input their destination address, use the send function key and damn well type in their own file name.

If you try to do everything for someone, they'll never appreciate it ... ◆

Contributors in this issue

Dennis Jarrett is Managing Editor of the TP Group newsletters and a 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).

Garret Keogh is the staff writer for TP Group, a computer enthusiast and general dogsbody; since coming to TP he has acquired more knowledge about desktop micros and envelope-stuffing than he ever thought possible — at the grand old age of nineteen he is adding to his computer education through a long process of trial and error(s). A music fan and committed socialist, his dream is to become a successful journalist ... these are his meagre first steps.

Paul Lavin is an American who succumbed to the wiles of an English woman and the siren song of the long British raspberry season over three years ago. He is a legal consultant and has recently renewed his love affair with computeroid machines. Unable to get beyond the third attack wave in *Space Invaders*, he has become conversant with the world of business software and the excruciating application of the Data Protection Act to the unsuspecting small business computer user.

Simon Potter is a business systems consultant with a speciality in word processors, file managers and spreadsheets. He says he'd like to like computers but can't find one that likes him enough to warrant that.

Brian Read was once a public health inspector but is now a freelance writer, mainly of books for children. He says he is "an enthusiastic but naïve" Apricot owner.

David St. John-Wallis is a sort of psychologist who runs an export marketing consultancy and writes *enormously* thick marketing books about the developing world. He's a chastened man since he first leapt to journalistic prominence in these pages; vituperation, vilification, abuse, rejection and guffaws have nearly broken his spirit. So he now runs a sort of computer software company which doesn't do very well. He says that all contributions will be gracefully acknowledged.

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).

Readers' Writes

Your chance to have a say

In Re. Release 2.10

In APRICOT FILE Release 2.10 you asked about reader experience with modems. The **Pace Series IV** caught my eye as it has a display that many other modems don't and is reasonably priced: could you do an article on it?

Here's a useful trick: CD > FILENAME will save the current drive and directories to the named file, which can later be used to reinstate the same setting by running PATHSET < FILENAME.

There was an error in David Langford's *Complete Escape Codes* that applies to BIOS 3.1 or greater. ESC ? is not assigned to shifted CALC; in fact special key 06 is assigned.

I recently observed that the clock advances on the screen after having been in Activity while the keyboard clock remains correct. This could perhaps be highlighted as a note in a future issue. I believe it is due to the fact that the on-screen clock in ACTIVITY is done by the ticker interrupt and there is something not quite right about that; perhaps the coding is fine-tuned to the speed of the 8086 chip, and I have the NEC V30 so it may no longer be accurate.

G.M.Sturges
Trélex, Switzerland

We'll be looking at the Pace range — Series IV costs around £300 in its V21/V23 incarnation, which seems a bit pricey to us.

IBM emulators

I would like to congratulate you on the extremely good service from DEALS FOR READERS. I posted an Airmail order from New Zealand and received the disks 14 days later. Congratulations, team.

However, all was not well. I had ordered PC-FILE, partly for the new IBM emulator of which I had great hopes. Well, what a disaster. (I hasten to add that I am using BIOS 3.1).

First, the keyboard file was hopelessly inaccurate — mainly in its use of the keypad, which I have had to fix using the previous emulator keyboard as a model. I have still to replicate the PRTSCN and NUMLOCK keys — I probably won't bother with the latter.

Second, the new emulator has very poor screen handling compared to version 3.6, circa 1985. The main disadvantage of the new emulator seems to be the very slow writing to the screen, compared to the very fast writing with 3.6.

An incompatibility appears when trying to run PRINTMASTER, a simple graphics printing program for making Christmas cards, calendars, and so on. This program fails to respond to all the cursor movement commands, no matter where they are supposed to be situated, and fails to refresh the screen between commands. Is this unique to the package?

Has Apricot got it wrong somewhere?

An announcement in New Zealand of a British software package called SoftPC (a relation to SoftClone?) claims to emulate an IBM PC, including all hardware. If it becomes available for Apricot, maybe we will be able to run all those programs requiring a graphics card. On the other hand, has anyone thought of writing an interface using GSX or similar to emulate the IBM or Hercules graphics cards?

A W Glennie
Christchurch
New Zealand

Roger has now fixed the problem with the numeric keypad keys, though PRTSCN and NUMLOCK still aren't implemented (we decided they wouldn't be required very much, and the function of PRTSCN is duplicated anyhow).

As for the other points — yes, the new emulator isn't as fast at screen-writing as its predecessor. No, we haven't come across the dead-cursor problem (or PRINTMASTER, for that matter). Yes, Apricot did probably get it wrong. We don't know anything about SoftPC — all input welcome. And the Hercules/EGA emulator in GSX sounds a good idea, but we don't know how practical it would be ...

Of Kermit

I saw in APRICOT FILE Release 2.11 that you intend to do some articles on the KERMIT file transfer package; and having had some experience of the little green terror I thought that I would share of my learning — such as it is.

The Apricot version works fine ... on an Apricot Xi. Some of us, however, are lucky enough to use Xens. On a Xen it *does* work, except for the Send Break command that wakes up our PACX so that we can log on to the VAX cluster. And if that is 99 per cent of your use of Kermit, it's not good news at all.

Even worse, the MicroScreen doesn't turn on! And why is this, you ask; "sloppy programming" I reply (the same reason that some very splendid and worthwhile Public Domain utilities won't run under MS-DOS 3.xx). However, with some swift adjustments to the machine-dependent source code all is well.

The second little KERMIT hang-up is that it has a habit of terrorising our delicate PACX by wantonly transmitting all sorts of unpleasant characters, especially during the REMOTE TYPE instruction. Guaranteed to hang the PACX at 50 paces. That I can't help with except to recommend that you don't use REMOTE TYPE.

The worst flaw though is an inability to transmit SuperWriter format files on the grounds that they are too big for KERMIT's internal buffers. Conversion to System or External format solves this, but is not ideal.

One of my favourites was the bug in the Wang PC version of KERMIT that would only surface during the linking phase to inform you in no uncertain tones that a 'FIXUP ERROR' had occurred (whatever that might be). This did

not seem to be a problem until I issued the STATUS command, which caused the Wang to retaliate by hanging. Turning to the manuals I found that Wang had obviously seen the Apricot manuals and were determined to provide even less hard information about the LINK program than a small English firm would dare.

After substantive research my enquiries led me to investigate the Assembler directives and I found that the machine-dependent module was perfectly correct. The other ten were wrong! Removal of the offending phrase 'Code' found shortly after 'Code Segment Public' made the world once again safe for democracy.

Having got the little blighter to run I was ordered back into the front line to make it give some decent Help responses. This was done and without use of a safety net has been added to our system as a new module called MSSHELP.ASM. Suitably modified MSXAPR.ASM and MSSKER.ASM modules are included and will when compiled with the others will result in a new version of KERMIT.EXE.

Apart from these little difficulties, KERMIT is a great improvement on the VTRAN VT100 emulator we were stuck with before; and version 2.30 of KERMIT — when it is ready — could be even better, since it will allow VTRAN-style raw transmitting (or so the documentation says and I haven't had the time to get it to run). Dear old SuperWriter just can't handle those 70KB source files!

I am enclosing all this on disk to be added to your public domain collection (with a little tweaking that allows KERMIT to run CHKDSK.EXE under MS-DOS 3.10 correctly — unlike the CHKDSK.COM version that simply emits a smug incorrect DOS message and stops).

As for the code on the disk, usual PD rules apply — no sneaky making money or erasing my name from the credits. Copies are winging their way to the old gods at Lancaster and maybe to the Holy of Holies at Columbia!

Jerry Taylor
Epsom

Jerry's excellent KERMIT is now the version featured on the Apricot Comms Disk, DEALS FOR READERS number AP022.

Company news

MBS scores again

The dire results of 12 months ago produced a 'massive' restructuring at the computer distribution leader **MBS**, with the interim results looking very good — turnover up 25 per cent to £36.8 million. Profits of £750,000 compare with a year-ago loss, and more acquisitions are rumoured to fuel the expansion.

Compsoft for Octagon now?

Following the failure of a proposed reverse takeover deal with Sagesoft and the arrival of a new suitor in Multisoft (APRICOT FILE Release 2.12), it now looks as though Octagon Industries will acquire software developer **Compsoft**. **Octagon** was a late arrival in Apricot's bid for Wordplex (APRICOT FILE Release 2.10) but was itself thwarted by Norsk Data; the investment capital operation says it's been keeping an eye on Compsoft since the start of the year, and estimates the total cost of buying the company at some **£2.3 million** — and that's for a company capitalised at nearly £6 million. Compsoft's two principals have already sold 1.8 million shares to one of Octagon's major stockholders.

The flurry of activity started when Compsoft turned in a very disappointing £568,000 on sales of £2.17 million for the year to 31 March. Perceptive subscribers may recall that we suggested USM-quoted Compsoft as a penny-share flutter way back in APRICOT FILE Release 1.3, December 1985, when the shares were around 35p; well, a couple of weeks ago (ie when we were writing this) Compsoft was at 78p and rising. Now *that* would have paid for your resubscription ...

Quest share stimulus probed

The good results from **Quest** (APRICOT FILE Release 2.12) weren't solely responsible for the 28p leap in the company's share price on Friday 2 October which added **£4.5 million** to the group's value. It was caused by the announcement of a £40 million order from Russia, which appeared on the Stock Exchange's TOPIC viewdata news service at 10.20am; sadly it was bogus, and share dealings were suspended less than an hour later. The Stock Exchange has cancelled all dealings between those times and has launched a full-scale enquiry centering on the names of those who sold following the announcement. The scheme may have been orchestrated; it seems someone tried to plant news of the 'order' in the *Financial Times* a couple of days earlier. Quest shares resumed at around 190p.

MicroPro turnround

WordStar purveyor **MicroPro** turned a year-ago loss of $1.2 million into profits of **$2.9 million** for fiscal 1987. Revenues were up 8 per cent to **$41.3 million**, so the return on sales isn't fabulous. The trend looks good, though — revenues are up 18 per cent in its most recent quarter; and the long-overdue revamps of WordStar that appeared in the second half augur well for MicroPro's future, though some more strong-selling products do look essential.

Meanwhile, MicroPro expects to sell its three millionth copy of WordStar during the next few weeks. All UK purchasers of WordStar Professional and WordStar 2000 (including upgrades) will get the chance to enter a celebratory prize draw. Existing users can also qualify — ask MicroPro for a re-registration card. Sounds like a way for MicroPro to tidy up its mailing lists, but a luxury fortnight in the Far East could be a fair trade for the junk mail ... Betcha the winner doesn't have a three-year-old copy of WordStar 3.3 in Apricot PC format, though. MicroPro is on **01-879 1122.** ◆

DEBUG for Beginners:

Part Two: Advanced Editing

By Simon Potter

In this instalment we're going to look at some fancy styles of editing with DEBUG. Then we'll get into the meaty area of MS-DOS memory organisation, at which point you may wish to buckle your seat belt ...

The Compare command

Let's dispose of this one first. The Compare command will take two blocks of memory and compare them byte for byte. If the two areas are identical, nothing is displayed and you get back the DEBUG hyphen prompt. If they don't contain the same information, both are displayed with their respective data bytes — the first address followed by its contents, then the contents of the second location followed by the address.

This example should make things a bit clearer. It compares eight bytes starting at offset address 100 with the same range from 200:

```
-C 0100 L 8 0200
6897:0100 E9 65 6897:0200
6897:0101 6B 70 6897:0201
6897:0102 01 74 6897:0202
6897:0103 43 65 6897:0203
6897:0104 4C 6D 6897:0204
6897:0105 4F 62 6897:0205
6897:0106 43 65 6897:0206
6897:0107 4B 72 6897:0207
-
```

As you can see, address 0100 contains the hex value E9; the first address in the compare-with range, 0200, contains 65 hex.

None of the eight bytes compared by this command did in fact match, so we got output for each pair. If there had been a match on any of them DEBUG would have skipped that pair; if all of the locations matched, DEBUG would have simply responded with another prompt. No other message is displayed.

So what? Well, obviously the **C** command is useful for comparing two blocks of data from a file; and in advanced applications it might be handy for comparing a program with the ROM BIOS — for instance, you might be trying to rewrite BIOS 2.7 and need to make sure that only certain parts had been changed. Otherwise, I've not found a great deal of use for Compare.

Advanced Enter

In Part One we covered the Enter command, used to place bytes of data in memory, in its 'display-then-modify' form. If you enter 'E' followed by an address, DEBUG will prompt you one byte at a time by displaying the present contents of the current location followed by a period. You can enter one or two hexadecimal characters to replace that value; or you can hit the space bar to go on to the next byte. (If you go too far, the hyphen will back up one byte each time it is pressed.)

But this can be slow and clumsy, especially where large amounts of data are concerned. And if you need ASCII data, you have to look up each character for its hex value before you make the change.

That's where the Enter command's 'replace' mode comes in handy. You can change several locations at a time; data can be entered in hex or ASCII format; and multiple bytes can be entered at one time without waiting for the prompt.

To use Enter in this form, follow the command with a start address: then give one or more replacement values preceded by a space or a comma. For instance, this would change whatever is in the four bytes starting at location 103 to the string 'TEST':

```
-E 103, 'TEST'
```

ASCII text — characters — have to be enclosed in single or double quotes. You could get the same result by using the hex codes:

```
-E 103, 54 45 53 54
```

And you can mix hex and ASCII:

```
-E 100,'TE' 53 54
```

If you now display that with a **D**-for-Dump, you'll see the word 'TEST' in the four locations starting at 0100 — because 53 is ASCII code in hex for 'S' and 54 is the code for 'T'.

The Fill command

Fill is another editing command, useful for entering a lot of same data. You specify a range of addresses and a list of one or more hex numbers or ASCII characters; DEBUG will replace the current contents of those locations with the new values given.

What's more, if there are fewer replacement values than there are bytes in the address range, the list will be repeated until the requested amount of memory is filled. (If the list is longer than the amount of memory to be filled, the extra items are ignored.)

As with the Enter command, Fill can take hexadecimal or character data. The address range can just be a start address, in which case as many locations are changed as there are replacement values. Or you can have a start and end address, separating them by a space or a comma. Or you can specify a number of bytes by following the start address with **L** and a hex value.

Unlike Enter, though, large amounts of data can be stored by Fill without specifying each and every character. As an example, to clear 32KB (8000 hex) of memory from address 100 onwards, you only need enter something like this:

```
-F 100 L 8000 0
```

That translates into 'Fill memory starting at offset address 100 in the current segment for 32K bytes (8000 hex is 32,768 in decimal) with the hex value 00'.

If the same command was entered with the data as '1234', the memory would be filled with the repeating string '123412341234'.

As a rule it's better to use Enter for changing smallish amounts of data — an error in the length parameter of the Fill command can destroy a lot of work. Enter will change only the number of bytes actually entered, minimising the effects of a parameter error.

Moving data

The Move command does just what it says — it moves data around inside the machine. In fact, it takes the contents of bytes from the address range nominated and moves them to the destination address.

It's particularly useful if you need to add an instruction into a program, for Move can be used to make room for the instruction. Be careful, though: data or labels referenced after the move will no longer be in the same place. Also, Move can be used to save a part of the program in free memory while you play with it; you can restore it at any time. It can also be used to copy the ROM BIOS into memory, where it can be written to a file or tinkered with to your heart's content — that way you can change things around in the BIOS without having to worry about reprogramming a ROM chip.

This will move 200 hex bytes of data from address 100 onwards to the address pointed at by ES:0100:

```
-M 100 L 200 ES:100
```

If you later want to return the data to its original position, you might use ...

```
-M ES:100 L 200 100
```

Unless the data has been changed while temporarily at the move-to location (ES:0100), this will restore the file to its original state.

How DOS handles memory

And now we get nearer to the nitty-gritty. MS-DOS keeps track of what's happening in memory generally (and to the currently-loaded program in particular) by reference to a special area of memory. It contains a series of locations called 'registers'; MS-DOS checks them all the time, and you can look at them too.

More to the point, you can use some of the registers to identify specific locations without knowing their actual addresses; and individual registers are used in assembler to implement instructions. In general, it's useful to know a little about registers — at least in broad terms. But it is possible to write assembly-language programs by rote, obeying the instruction syntax and utilising the registers required by the instruction, without being too aware of *why*.

The **R** command will display the contents of the registers. The command may also be used with a single register name; the current hex value of that register will be displayed, followed by a colon as a prompt. You can then enter a value to change the register.

But let's start simply: the display at the foot of the page came in response to an **R** command on its own. So what are we seeing here?

Everything on the first two lines refers to registers, and the equals signs indicate their current contents.

At the bottom of the display you get the contents of the first memory address of the currently-loaded program — its location, hex code, and assembler mnemonic (not the ASCII-code equivalents, this time). The **DS** entry is another register; it's displayed down there because it shows the contents of any memory referred to by the instruction.

IP is the Instruction Pointer, sometimes called the 'Program Counter' (and you can check its current contents by using R IP *or* R PC). It contains the address of the next instruction to be executed.

Well, that's not exactly true. It is 16 bits long, and the largest number which can be represented in binary form in sixteen bits is 65,535. But the Apricot *starts* with 256KB — 262,144 bytes, most of which could be used to hold an instruction. So how does MS-DOS know which instruction to execute next when it's located in byte 65,536? How does the IP register handle this address?

Well, it doesn't because it can't. Instead MS-DOS uses a clever trick. To cope with the maximum number of memory locations an Apricot can handle, you actually need 20-bit numbers — in 20 bits you can express values up to 1,048,575 decimal.

```
-R
AX=0000  BX=0000  CX=0000  DX=0000  SP=5ABE  BP=0000  SI=0000  DI=0000
DS=7A54  ES=7A54  SS=7A54  CS=7A54  IP=0100   NV UP DI PL NZ NA PO NC
7A54:0100 0000          ADD     [BX+SI],AL                  DS:0000=CD
```

MS-DOS handles this a bit like house addresses in a town. In theory, you could number each house in London separately; so my address could be 'S Potter Esq, No 134,523, London'. For anything but a small village that's a bit unwieldy, to say the least. It makes better sense to subdivide the address logically — by house number within street name, for instance. That way you can locate a particular address by finding the street first and then looking for the individual residence.

That's what MS-DOS does. It treats memory as being organised into areas called segments, which provide one part of the address: within each segment there are individual locations that provide the other part of the address. The 'street-name' segments are 16 bytes long.

To fit a 20-bit address into a 16-bit register, MS-DOS splits it into two parts. The 16 bits on the left of the binary number — which will fit into a register — become the segment address and go into a segment register. And you can see that this is why segments are 16 bytes long; if you add 1 to the segment address in the segment register, you're actually adding 16 to the full 20-bit address since the rightmost four bits have been temporarily dropped — you're adding 10000 in binary, which is 16 in decimal.

As for the four bits on the right of the address, they're the house number equivalent — they indicate the precise location within a segment; specifically, they count locations from the start of the segment.

To reference memory, then, the computer uses the segment address and the number of bytes above the boundary — the latter is called the offset. So when writing an address we use the format SEGMENT:OFFSET; and when DEBUG displays memory locations — with the D-for-Dump command, for instance — they come out in the same form.

The segment registers

Now we can be a bit more precise about where MS-DOS finds the next instruction in a program. The CS register is the *code segment*; the computer looks for the next instruction inside the segment referenced by the CS register, adding the number of bytes specified in the IP register to the start of that segment.

Because the CS register references the segment for program code, it can conveniently be used to identify addresses — for instance, CS:0200 is a sensible way of saying 'offset 200 hex in whatever you currently have as the code segment'. In fact, the register names are frequently used like this: you could say the next instruction to be executed is that given by CS:IP. This shorthand can be used with any of the register names in assembly-language instructions as well as in text descriptions of what's where in memory.

CS is one of a group of registers used for holding segment references. The DS register gives the *data segment*: when you use an instruction that includes a memory address for data, the computer looks that number of bytes above DS.

The SS and SP registers identify the *stack*, a convenient chunk of memory for short-term storage of data. We'll see how the stack is used when we come to look at some individual assembler commands; but the SS register gives the segment, SP provides the offset into the stack segment.

The ES register defines what's called the 'extra segment'. Some instructions use this, most do not. Normally it can be used as you see fit.

A bunch of registers are also utilised for addressing memory, typically with a segment reference: the BP (base pointer), SI (source index), and DI (destination index) are pointer and index registers used in addressing and often give an offset within a segment.

Data registers

The AX, BX, CX, and DX registers get the most use; they're called the data registers, and are basically general-purpose registers.

All registers are two bytes long. But these four can be addressed as either one sixteen-bit register or two eight-bit registers; the AX register, for example, is subdivided into the eight-bit registers called AL and AH. The 'L' denotes 'low byte', the 'H' is for 'high byte'.

That terminology stems from the idea of loading a 16-bit word into the register and then addressing the 'high' byte (bits 8 to 15) or the 'low' byte (bits 0 to 7) separately. Why? Because frequently you won't need to work with 16 bits for these registers — ASCII codes, for example, are only a byte long. To be forced to use a full two-byte word would be a waste of time and memory.

The data registers don't have predefined functions, and in some assembler instructions any of them can be used when a working area is required. But it's fairly common for programmers to use the AX register (or AL) for arithmetic operations; for some reason it's faster than the others.

When a file is loaded MS-DOS uses CX to give the length of the file; if it were larger than 64K bytes, BX would also be used for the high-order part of the number indicating the file's size (in the same way as the segment:offset method is used to give any large address). And when a file is saved, MS-DOS will take the number of bytes to be saved from the current settings of BX:CX if no parameters are used with the Write command — if BX=0002 and CX=0200 then 20200 hex bytes will be written. So while you might utilise these two registers as working areas, it's safest to make sure when you're done that they do contain the file length.

Because of that, most programmers avoid using BX:CX for *anything* but file length. If it *is* used, CX is typically a counter for loops.

The DX register generally holds the I/O port addresses for input and output instructions.

```
-R
AX=0000  BX=0000  CX=0000  DX=0000  SP=5ABE  BP=0000  SI=0000  DI=0000
DS=7A54  ES=7A54  SS=7A54  CS=7A54  IP=0100   NV UP DI PL NZ NA PO NC
7A54:0100 0000           ADD      [BX+SI],AL                  DS:0000=CD
```

Flags

The F register doesn't get an 'F=' in the display from an R-for-Registers command. Instead its current settings are indicated by that string of two-digit codes we've highlighted above.

The eight flags are effectively markers for the current state of the program being executed, and they'll change as different instructions are executed. These indicators are either 'on' or 'off', the display coming up in sequence as follows:

Flag	ON		OFF	
Overflow	yes	OV	no	NV
Direction	decrement	DN	increment	UP
Interrupt	enabled	EI	disabled	DI
Sign	negative	NG	positive	PL
Zero	yes	ZR	non-zero	NZ
Auxiliary Carry	yes	AC	no	NA
Parity	even	PE	odd	PO
Carry	yes	CY	no	NC

The purpose of the flags is straightforward enough. In general, think of the program as someone in a downhill ski race; his/her course is marked by flags — and the computer's flags are program markers in much the same way.

Using R-for-Registers

Back to DEBUG. Let's take a look at the registers for a specific file. So load the sample file MENU.TXT and Dump the first few locations. Now let's take a look at the state of the registers: the screen display should look something like the panel at the foot of the page.

When you type in an 'R' with no parameters like this, the display shows the hex contents of the registers and the flag settings for the address 705A:0100.

- The four segment registers — DS, ES, SS, and CS — are all set at 705A, the same number you've been seeing on the left side of your Dump display while strolling through DEBUG.
- IP, the instruction pointer, is set at 0100 hex — the first byte of the file.
- Nothing's happening in the AX and DX data registers
- The CX register tells you the length of the file currently loaded; if it were larger than 64K bytes, BX would contain the high-order part of the number indicating its size in the same way as the segment:offset method is used to give any large address.
- The first three hex bits of our file are shown in assembly language form.

To change the contents of one of the registers, enter R with the register's name. Take a look at the current value of the RX register — it's set to zero. Let's place 1234 hex in that register:

```
-R AX
AX 0000
:1234
-
```

DEBUG responds with the register's name and the current contents, followed on the next line by a colon prompt. Typing the new value pressing Return will change the register and give you the normal hyphen prompt again.

You could check the change with another R command, and the new value should be there — note that *nothing* apart from the AX register's value has changed.

The R command can only be used for 16-bit registers — it cannot change the eight-bit registers. For instance, to alter just the contents of AH you'll have to enter the data in the AX register with your new AH value and the old AL value.

```
-N MENU.TXT
-L
-D
705A:0100  20 0D 0A 0D 0A 0D 0A 0D-0A 0D 0A 0D 0A 0D 0A 0D    ................
705A:0110  0A 0D 0A 20 20 20 20 20-20 20 20 20 20 20 20 20   ...
705A:0120  20 20 20 30 2E 20 20 20-20 20 20 54 49 44 59 2D      0.       TIDY-
705A:0130  41 4C 4C 0D 0A 20 20 20-20 20 20 20 20 20 20 20   ALL..
705A:0140  20 20 20 20 20 31 2E 20-20 20 20 20 20 43 48 4B        1.       CHK
705A:0150  44 53 4B 20 43 3A 0D 0A-20 20 20 20 20 20 20 20   DSK C:..
705A:0160  20 20 20 20 20 20 20 20-32 2E 20 20 20 20 20 20            2.
705A:0170  43 48 4B 44 53 4B 20 41-3A 0D 0A 20 0D 0A 20 20   CHKDSK A:.. ..

-R
AX=0000  BX=0000  CX=00EF  DX=0000  SP=CF2E  BP=0000  SI=0000  DI=0000
DS=705A  ES=705A  SS=705A  CS=705A  IP=0100   NV UP DI PL NZ NA PO NC
705A:0100 200D           AND      [DI],CL                     DS:0000=CD
```

```
-R
AX=1234  BX=0000  CX=00EF  DX=0000  SP=FFEE  BP=0000  SI=0000  DI=0000
DS=2386  ES=2386  SS=2386  CS=2386  IP=0100   NV UP DI PL NZ NA PO NC
2386:0100 200D          AND     [DI],CL                    DS:0000=CD
```

Stepping out

With the Trace command you can see how the registers change according to what's going on in a particular location of the file. We're playing with a text file, of course, but usually the 'file' would be a program; Trace assumes that what's in a particular address is part of an assembler instruction, and tries to execute it.

Enter 'T' at the hyphen prompt. MS-DOS will 'execute' what it takes to be an instruction at 100 hex and advance to the next 'instruction' (which will be at 103 hex, since assembler instructions always take two bytes — more about this in Part Three). Your screen will show something like the display at the top of the page.

The only change after executing the instruction at 100 hex is that IP is now pointing to 0103, the next address to be disassembled and executed. But by Tracing through a program in this way, one step at a time, you'll start seeing changes in other register values — in short, you can clearly see the actions your computer is taking while executing the program.

If something is going on which you'd like to change, you can use DEBUG to make the alteration: test the amendment with the program-execute command Go: and, if it works, save it to disk with the Write command.

You don't have to Trace one step at a time. You can use T followed by a hex number to execute and trace through that many instructions.

And you don't have to trace from the beginning of a file: follow the T with an equals sign and a hex address offset to commence the trace from that location.

You can combine those two — use T followed by equals and a start address, then a comma and a hex value for the number of instructions to be executed. So T=200, 16 would execute and Trace through 22 instructions starting at offset address 0200 (16 hex is 22 decimal, remember). CONTROL-S can of course be used to pause the scrolling trace.

Change flag settings

There's also an RF command, which displays the settings of the flags for the current instruction:

```
-RF
NV UP DI PL NZ NA PO NC -
```

The display of current flags ends with a hyphen prompt. Pressing Return will leave the flags as they are; or you could change the contents of one or all of them by entering the opposite value from the table above. The changes need not be entered in any particular order, and you don't have to use capital letters; but you must use enter only valid combinations as listed in the table (an invalid entry will have DEBUG reporting 'BF' for Bad Flag and you'll be returned to the hyphen prompt with no changes made).

Last words

End of Part Two. You should now have some appreciation of how MS-DOS handles memory and how it keeps track of programs as they execute. Next time — using DEBUG for assembler programming. ◆

DEBUG Commands in summary

Note: Optional parameters are indicated by square brackets.

command	syntax	function
ASSEMBLE	A [start address]	enter assembly-language instructions from the address given; assembler instructions are entered subsequently. If the address is omitted, offset 0100 is assumed
COMPARE	C range, address	compares the contents of locations in the given range with the same number of locations starting from the second parameter
DUMP	D [range or address]	displays contents of locations in the range specified, or 128 bytes from the start address given. If the parameter is omitted, 128 locations from offset 0100 are displayed

ENTER	E address [, list]	replaces contents of locations starting from the given address with values in the list. If the list is omitted, individual bytes from the start address are displayed for manual alteration or skipping
FILL	F range, value[s]	current contents of the address range are overwritten by value(s) in the list
GO	G [= address] [, address]	executes program in memory. Addresses are optional; first one indicates where execution is to start, second is a breakpoint at which execution should stop with flags and register contents displayed. If parameters are omitted, execution starts from the address currently indicated by the CS and IP registers
HEXARITHMETIC	H value, value	calculates and displays the results of adding and subtracting the two hex values
INPUT	I value	inputs and displays one byte from the port address specified by the hex value
LOAD	L [address [, drive, blockid, blockno]]	loads from disk into memory. If parameters are used, DEBUG loads from the nominated drive the specified number of blocks (blockno) starting at the block given by blockno; blocks are loaded into memory starting at the address. If only the address parameter is given, that specifies the start point for loading the program referenced by the last Name command or specified when DEBUG was invoked; if L is used without parameters, that program is loaded at CS:0100
MOVE	M range, address	moves the contents of the address range into bytes starting at the second parameter
NAME	N filespec [,filespec[s]]	nominates a file to be Loaded, Written, or used as a parameter for a currently-loaded program
OUTPUT	O value, byte	outputs the second parameter (a hex number) to the port specified by value
QUIT	Q	terminates DEBUG and returns you to MS-DOS
REGISTER	R [register name]	displays current settings of specified register — or of all registers, if no parameter is used. In the latter case, individual registers can be examined and changed subsequently by entering their two-digit identifier code
REGISTER FLAGS	RF	displays current settings of flags. Current settings can subsequently be altered
SEARCH	S range, list	searches the nominated address range for ASCII and/or hex values given in the list. DEBUG then reports any locations at which the searched-for items were found
TRACE	T [=address] [, value]	executes one instruction (if the final value is omitted) or more (as many as are indicated by value) and displays the register settings after each execution. Execution begins at the location currently given by the CS and IP registers unless the start address is specified
UNASSEMBLE	U [range or address]	shows contents of memory as assembly-language equivalents. This begins at the location currently given by the CS and IP registers unless a start address is specified; 32 bytes are disassembled from the start point unless a range is given
WRITE	W [address [, drive, blockid, blockno]]	writes the file being debugged to disk. When no parameters are used, the save begins at the location currently given by the CS and IP registers; and the number of bytes saved is specified by the current values in the BX and CX registers (which will have changed if Go or Trace have been used, or if the file's length has been altered by additions or deletion). The fullest version of the Write command saves the file starting at particular address on to a nominated disk drive, where it is placed in the number of blocks specified (blockno) starting at a given block (blockid)

Free Small Ads

Apricot Sales

■■■ APRICOT F2 with 768K RAM, colour monitor, Apricot internal modem, two infra-red mice, MicroCache, original copies (with manuals) of Microsoft Word, SuperCalc 3, Communiqué, Brainstorm, GEM, GW-Basic, MS-Basic, and IANKEY. Also copies of WordStar Professional, Friday, Cardbox, Open Access, ChitChat, and a dozen disks of utilities and games. Also a dozen Apricot-specific textbooks plus all back issues of APRICOT FILE and Apricot User. First offer over £650 secures. Phone Chris on Brighton (0273) 690422.

■■■ APRICOT F2 with 512K RAM, twin 720K drives, infra-red keyboard and mouse plus GEM Write and GEM Paint software. Complete with all manuals and spare disks. £375 ono. Contact R Brown, 0276 36087, after 6pm.

■■■ APRICOT POINT 32 NETWORK FILE SERVER — 10MB, 512KB, one internal network card, all software, cables, manuals (or attach an F series monitor and keyboard and it becomes an F10). Unused: £495. Additional network card £100 (list price £199). Call M Lund on 0603 619711 office hours.

■■■ APRICOT PC with 24MB external hard disk, 768K memory, 12in and 9in monitors, mouse; plus enough software to run an empire (includes GEM, WordPerfect 4.2, 1-2-3, Norton Utilities, Fancy Font, Mix-C compiler); parallel printer cable, ASYNC for IBM PC, 30+ floppies in Flip 'n' File box. A steal at $2,500. J Mark Bailey, 2128 Southwood Road, Jackson, MI 39211, USA — tel (601) 982-0005.

■■■ APRICOT Xi: 10MB partitioned hard disk, 1MB RAM, MS-DOS 3.20, BIOS 3.12, Concurrent CP/M-86, two monitors, carrying case; plus CANON PW1080A matrix printer; plus three word processors, three databases, three comms packages, an integrated package, spreadsheet, GW-BASIC, Turbo Pascal, Mix-C, Eco-C, TRAPDOOR and BYTES diagnostics, both Apricot Technical Reference Manuals, Black Hole disks, both Apricot DOS shells, dozens of utilities and games galore — and a box of spare disks. £1,400 plus VAT ono (no split sale). Contact John Lyth, 01-583 5866 (day) or 0689 55667 (evenings).

■■■ APRICOT POINT 32 NETWORK — or will split. Includes 11 Apricot F1Es with 512K (together at £3,850), 4 F1Es with 256K and network cards (£1,200), 16 stand-alone F1Es with 256K (£4,800), one Apricot PC (£400), one 10MB add-on disk drive (£1,000). Plus SuperWriter, SuperCalc, SuperPlanner, WordStar Professional, GW-BASIC, Pegasus, Delta IV, Logo. Total price £11,250 or will split. Contact Ms A Goodrum, 01-953 6024.

Apricot Add-on Sales

■■■ 512K EXPANSION CARD — SuperRAM by Dataflex with Magic Memory and manual, used for nine months in PC. Price £120. Contact 0732 810250 evenings.

■■■ APRICOT SOFTWARE: Delta 4 v4.1c £125, Delta 2 v2.1c £60, SuperCalc 2 v1.00 £95, Cardbox Plus v2.1 £125, Lotus 1-2-3 v1.00 £150. All brand new with registration cards etc; prices are a guide only — I will haggle! Also three Point 32 LAN packs and a Bank 100 tape, again brand new: offers? Can deliver in London/Kent. Phone 0679 63258 evenings and weekends.

■■■ DATA WARDEN 20MB Tape Cartridge Backup with PC/Xi controller card (additional cards from Simon Computers allow several micros of different makes to share the backup). Includes manual and software. Brand new with Apricot warranty. £590 (list price £995). Call M Lund on 0603 619711 office hours.

■■■ SANYO CM3117 COLOUR MONITOR for F2/F10; list £399, asking price £195. Also brand new software — LOTUS 1-2-3 £190, OPEN ACCESS £199. Call M Lund on 0603 619711 office hours.

■■■ APRICOT 9in MONITOR for PC or Xi. Cream. Little used, £40 plus VAT. Telephone 0705 730566.

■■■ SUPERWRITER BUGS CURED — twelve files identifying and curing errors in SuperWriter version 1.03 program and manual, all for a copying fee of only £5.75 (includes VAT). Call Tudor Wynn-Williams on 097 081 375.

■■■ APRICOT PC 9in MONITOR. Brand new and unused (I bought a 12in screen instead): £100. Also APRICOT PC MOUSE, virtually unused: £75. Phone Robin Leaver on 09662 4104.

■■■ APRICOT PC SOFTWARE — OPEN ACCESS integrated system (database, spreadsheet, 3-D graphics, word processing, comms, appointments diary); virtually unused, a real bargain at £250. Sapphire's REPORT MANAGER (3-D spreadsheet, over 60 functions); £100. THE ULTIMATE DIARY (up to 5,000 entries): £15. MICROFILE (superbly simple electronic card index for up to 32,000 records): £20. NUTCRACKER SUITE (simplified text editing with typewriter-like functions): £15. Contact A Leigh on 01-542 4872 evenings.

Other Sales

■■■ HEWLETT-PACKARD HP28C scientific calculator with HP82240A infra-red printer. Both used for one day only; cost over £330. First offer over £230 secures. Phone Chris on Brighton (0273) 690422.

■■■ BROTHER TWINWRITER. Hardly used. RRP £1,295 — price £650, including spare ribbons. Contact G Hacker, Infospeed Ltd, 0202 734363.

■■■ DATA GENERAL 16-bit minicomputer with removable hard disks and multiple comms port. Original cost £25,000-plus; project failure now means all offers considered (or exchange for Apricot). All documentation, plus Assembler, Basic, Fortran. Phone 01-948 1603 any time.

■■■ COMMODORE MPP1361 printer; 136 columns, 150cps. As new (hardly used). Offers around £350. Phone Robin Leaver on 09662 4104.

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

■■■ 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 9in MONITOR (cream) wanted for PC. CARRYING CASE also needed. Phone 01-908 4631.

New from Deals For Readers

Modem

Stradcom II

Order no:	AH008
Price:	£200

For the Xen-i — we think this is just about the best internal modem for IBM-compatibles. Offers 300/300 (V21) and 1200/1200 (V22) speeds, choice of pulse or tone dialling, phone socket, auto-dial/answer, monitoring speaker. Latest version has improved support for PSS links, and Dataflex says new chips and a degree of re-engineering make Stradcom 2 "the most reliable PC modem on the market, particularly for use in areas still served by the old Strowger-based exchanges" (where the amount of noise can befuddle less resilient modems). Hayes-compatible, naturally. Comes complete with special customised version of Procomm called Baudwalk.

Cables and connections

Problems with RS-232 connections? These ready-to-go cables and simple-to-use tools should help ... We can also obtain just about any parallel or serial cable for connecting printers, plotters, modems or what have you to virtually any make of computer — call the DEALS FOR READERS service on 01-833 3501 with your queries.

File transfer cables

Apricot to IBM PC:

Order no:	CC146
Price:	£19.00

Apricot to IBM PC/AT:

Order no:	CC515
Price:	£19.00

Apricot to Sirius:

Order no:	CC513
Price:	£19.00

Null modem cables for file transfer; two metres with male connector at the Apricot end and male or female as appropriate at the other. These and the following items are all fully-moulded RS-232 cables with high-impact connectors, moulded rubber thumbscrews, and a five year guarantee. Not the cheapest we've seen — but these are some of the best quality we've found, and we don't think you should take chances with inferior connectors for the sake of a few pence.

Null modem cable

Order no:	CC553
Price:	£18.00

General-purpose two-metre length of male-to-male null modem cable for file transfers.

Modem cable

Order no:	CC552
Price:	£19.00

General-purpose RS-232 cable with male connectors on both ends — should be suitable for any modem except Pace Nightingale; for that, order separately as CC511.

General-purpose cable 25 lines connected

Order no:	m/m	CC260
Price:		£18.50
Order no:	m/f	CC261
Price:		£18.50
Order no:	f/f	CC262
Price:		£18.50

Two metres of RS-232 cable for modems, printers etc with all 25 pins connected — note that all three possible combinations of male and female connectors are covered.

General-purpose cable 9 lines connected

Order no:	m/m	CC255
Price:		£12.50
Order no:	m/f	CC256
Price:		£12.50
Order no:	f/f	CC257
Price:		£12.50

Two metres of RS-232 cable with pins 1-8 and pin 20 connected — again, note that the possible combinations of male and female connectors are all covered.

Parallel printer cable

Order no:	CC190
Price:	£12.00

Two metres of purpose-designed Centronics-type flat ribbon cable with appropriate connectors.

Serial printer cables

Apricot to Brother:

Order no:	CC117
Price:	£17.00

Apricot to LQ1000:

Order no:	CC510
Price:	£17.00

Apricot to other Epsons:

Order no:	CC512
Price:	£17.00

Apricot to Oki:

Order no:	CC516
Price:	£17.00

Apricot to HP LaserJet:

Order no:	CC555
Price:	£11.50

Two metres of purpose-designed RS-232 cable with appropriate connectors. Use CC255 or CC256 above for other printers.

How to order

Simple — send us a note or a purchase order with the item's name and the reference number as quoted in the list.

We can accept purchase orders from companies and other organisations *only*, and *only* for goods totalling more than £50. Sorry, but we can accept telephone orders only for credit card sales.

All our prices include VAT and despatch (except for overseas mailing of books: see the relevant entries). Your order will arrive with a VAT invoice/receipt. There are three ways to pay:

- **Girobank transfer** : Our National Girobank account number is **59 028 1909**
- **Cash with order** : A cheque drawn on a UK bank, a Sterling money or postal order, or a Eurocheque can accompany your order (make cheques payable to 'TP GROUP' and cross them)
- **Credit card** : We can accept **Diners Club, Access** (*aka* **Mastercard**), and **American Express** (*but not Connect or Barclaycard/Visa*). For a payment we need a signed letter that includes your name as it appears on the card, your card number, and its expiry date.

WordPerfect's MicroScreen

Fixing the LEDs and labels

By Andrew Glennie

As currently supplied, WordPerfect uses the Apricot MicroScreen; but it doesn't utilise the LEDs to indicate that the keys are operative. This is because the relevant Escape codes are absent from the WP.COM file.

The relevant code is ESC/? (and that should be added to the APRICOT FILE Release 2.11 exposé of Escape Codes). This code turns on all the LEDs; ESC/@ turns all the LEDs off. So to make the LEDs operate you just have to insert these codes into the various COM files used by WordPerfect. Of course, we wouldn't do this on our *Master* disks, would we? But once you have the program operating as you wish and there are no bugs, you may wish to update your Masters.

You can use DEBUG to modify the files. Start with a copy of WP.COM (it's the easiest to alter) and type in D200 to Dump location 200.

The principle change is at locations 207 to 209, where you'll be inserting 1B 2F 3F (the hex equivalent of ESC/?). First you'll have to create some space for it, the easiest method being the Move command:

```
-M 207 258 210
```

This moves the block of locations from 207 to 258 along three places to start at 210, creating a gap for the three codes required. Those can now be entered with ...

```
-E 207
  213E:0207  4D:1B 6F:2F 76:3F [CONTROL-C]
-Q
```

(Your input is shaded.) Having turned on the LEDs with this, they should be turned off when leaving WordPerfect. That is achieved by modifying locations 308 to 310 to contain 1B 2F 40 — the hex equivalent of the Escape code sequence ESC/@.

You could of course enter all the alterations by using Edit to overwrite the existing contents; but getting the spacing right for the Microscreen labels is a tedious task — I know because that's what I did the first time!

There is still one problem. If you want to use WordPerfect's Shell option to perform other tasks, such as updating the Dictionary or resetting printer commands, you'll note that there are also COM files associated with the relevant EXE files; and these have helpful little messages which use the MicroScreen. Say goodbye to your new WordPerfect labels, because when you return to the word processor all you see are the messages left over from the Shell access — or, if you're lucky, a completely cleared MicroScreen.

To fix this means a little more work with DEBUG, which unfortunately means some typing unless you've been crafty. The patch involves replicating the information seen above at location 200 to 260 in the new WP.COM file to location 300 in each of the offending COM files — which means at least CONVERT.COM, SPELL.COM, PRINTER.COM and PRHELP.COM (for WordPerfect version 4.2). The result will be that your Microscreen will reset itself to the required information when you exit one of the utility programs.

You could type in the information by hand, which is what I did (oh, the tedium). Or you can use the Load command from DEBUG — which I have not tried, so on your own head be it. The principle is that you Write the specific locations containing the revisions to a named file on disk, nominate the file to be patched by using Name, then Load the revisions to insert the new data.

At the bottom of the page is the relevant part of the final version of CONVERT.COM, with the major changes showing at location 300. The more observant may spot some conflict between the version of the utility programs noted on the screen and what appears on the MicroScreen. These can be altered by making the relevant changes to the text situated at location 200 in the relevant COM files.

Best of luck — and I hope you enjoy the new improved WordPerfect MicroScreen. ◆

```
213E:0200  0A 1B 55 1B 57 1B 67 1B-45 1B 63 1B 2F 40 57 6F   ..U.W.g.E.c./@Wo
213E:0210  72 64 50 65 72 66 65 63-74 20 44 6F 63 75 6D 65   rdPerfect Docume
213E:0220  6E 74 20 43 6F 6E 76 65-72 73 69 6F 6E 20 55 74   nt Conversion Ut
213E:0230  69 6C 69 74 79 20 56 65-72 73 69 6F 6E 20 32 2E   ility Version 2.
213E:0240  32 20 1B 56 24 00 00 00-00 00 00 00 00 00 00 00   2 .V$...........

213E:0300  1B 55 1B 57 1B 45 1B 2F-3F 4D 6F 76 65 20 20 53   .U.W.E./?Move  S
213E:0310  75 70 2F 53 75 62 20 53-68 65 6C 6C 20 44 61 74   up/Sub Shell Dat
213E:0320  65 20 20 4D 72 67 2F 53-72 74 20 4D 61 72 6B 20   e  Mrg/Srt Mark
213E:0330  20 42 6C 6F 63 6B 20 4C-52 5F 49 6E 64 20 20 54    Block LR_Ind  T
213E:0340  68 65 73 20 20 52 70 6C-63 65 20 4D 72 67 5F 43   hes  Rplce Mrg_C
213E:0350  20 20 20 46 74 4E 6F 74-1B 56 24 00 00 00 00 00      FtNot.V$.....
```

The Workshop

Batch File Recipe:

Disk tidy

By David Gaunt

TIDY.BAT is a batch file for deleting unwanted files from a disk — not unlike the DOTOALL utility when used as DOTOALL DEL *.BAK. The syntax in this case is ...

```
TIDY <drive:> <directory(s)>
```

If no directories are specified, the batch file will chain to CATDIR.BAT (like my CAT.BAT disk cataloguer — see the *Batch File Recipes* from APRICOT FILE Release 2.12) and work through all directories on a disk.

The loop in lines 8 to 18 first ensures the files are write-enabled using the public domain utility CHMOD before prompting you to verify deletion of *.BAK files. (The loop can easily be modified to delete any other temporary files you have created.)

This batch file changes the default directory, and so with a little ingenuity it can be adapted for use with other utilities which do not fully support directory paths — CHKDSK, for example, could be used to look for non-contiguous blocks in all directories.

TIDY.BAT

Line numbers are included for reference — don't enter them when keying in this batch file.

```
1   ECHO OFF
2   ECHO ^[z^[(
3   IF %1@==@ GOTO LABEL1
4   IF NOT EXIST %1CATDIR.BAT GOTO LABEL2
5   SET DRIVE=%1
6   IF %2@==@ %1CATDIR %0 %1
7   SHIFT
8   :LOOP
9   ECHO ^[z^[(
10  IF %1@==@ FOR %%@ IN (CD CRLF) DO %%@
    %DRIVE%\
11  ECHO ^[)^[p Verify deletion on the following
    files in directory %DRIVE%%1 ^[q^[(
12  CD %DRIVE%%1
13  IF EXIST %DRIVE%*.BAK CHMOD %DRIVE%*.BAK/W
    >NUL
14  IF EXIST %DRIVE%TEMP*.* CHMOD
    %DRIVE%TEMP*.*/W >NUL
15  IF EXIST %DRIVE%*.BAK VDEL %DRIVE%*.BAK
16  IF EXIST %DRIVE%TEMP*.* VDEL %DRIVE%TEMP*.*
17  IF EXIST %DRIVE%*.$$$ VDEL *.$$$
18  FOR %%@ IN (SHIFT IF) DO %%@ NOT %1@==@ GOTO
    LOOP
19  CRLF
20  :LABEL1
21  ECHO ^G^[p The syntax is TIDY [drive:]
    [directory(s)] ^[q
22  ECHO ^[p directories are optional ^[q
23  CRLF
24  :LABEL2
25  ECHO ^G^[p No file of directories
    (catdir.bat) found on drive %1 ^[q
```

Contribute!

Why not share your own useful utilities? Let us have your own home-brewed batch files, and we'll write them up for this *Recipe* series. Just us a line at APRICOT FILE, PO Box 509, London N1 1YL.

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.

Postal address

TP GROUP
PO Box 509
London N1 1YL, England

Personal calls and deliveries

TP GROUP
Unit 4, 47-51 Wharfdale Road
London N1 9SE, England

Phone, E-mail, Telex

Telephone	01-833 3501
Telecom Gold	81:JET767
	[from overseas 10081:JET767]
Easylink	19011170
The Source	BDE566
Telex	265871 MONREF G
	quote reference 81:JET767

The People

Editor	Dennis Jarrett
Production	Fiona Carey
Write	Garret Keogh
Subscriptions	Terrance Duggan
DFR/Technical	Roger Gann

The Guarantee

The annual subscription to APRICOT FILE buys 12 monthly issues as follows:

Britain	£50.00
Rest of Europe	£60.00
Elsewhere (air mail)	£65.00

If having subscribed you come to feel that APRICOT FILE does not meet your requirements or that it does not match our claims, we will gladly cancel your commitment to the magazine and refund the unused portion of your subscription.

■■■ ISSN 0268-6066 ■■■

Doodles by David

Tips and Tricks

The occasionally essential column

By David St.John Wallis

Did you know you pay Telecom Gold for files you receive as well as those you send? And did you know that when Gold flash up helpful signs like "^XX?XX not knownnn please res[pecify" (*sic*) you're still paying?

So here's a heuristically useful spreadsheet:

```
A1:   'TELECOM GOLD COSTER
A3:   'ENTER NUMBER OF CHARACTERS SENT/RECEIVED:
A4:   'ENTER TG CONNECT MINUTES:
B4:   'CONNECT MINUTES:
B7:   'The character cost was:
F7:   +F3/512*0.04
B8:   'The connection cost was:
F8:   +F4*0.065
B10:  'The total net cost will be:
F10:  SUM(G7: G8)
B11:  'And VAT will be:
F11:  +F10*15/100
B13:  'THE TOTAL GROSS COST CHARGEABLE SHOULD BE:
F13:  +F10+F11
```

Some might multiply line H13 by two to pay for the call itself. Tsst. Tsst.

Here's another one, a letter to Telecom Gold written in SuperCalc that's designed to transmit the way it will get received anyway.

```
A1  =  "Dead
B1  =  "TelecOm
C1  =  "GOld
D1  =  "Systxxx
E1  =  "PersOn:
A2  =  "YOur
B2  =  "cO%ts
B3  =  "ab%o[]ute[]y
C3  =  "outdageou%
D3  =  "and
E3  =  "I
A4  =  "pdopo%e
C4  =  "te]]
D4  =  "a]]
E4  =  "my
A5  =  "friend
B5  =  "to
C5  =  "go
E5  =  "For{num%
A6  =  "in
B6  =  "fu{ure.
```

In case *you* want to write a letter like this, use the Contents option after the /Output command. With Lotus 1-2-3, go for the Options Other Cell-Formulas doodah after Print File.

User friendly

If you are having difficulty with Prestel and version 2.20 of Communiqué, go into Parameters and change the sign-on file to:

```
[SNGLE]
identity
[I]
password
[P]
```

The secret? You guessed. A conspiracy of Telecom Gold and Apricot Software decided to spell 'single' without an 'i'. And identity and password must be in lower-case. Occasionally essential? Ha!

Disgusted?

I thought last month's APRICOT FILE was dreadful. Didn't you? Boring old spreadsheets, long underdue crits of sluggish WordPerfect, mindless escape codes. Not even a good typo to liven things up. What it missed was the sardonic repartee, the trivial, the drivel for which this column has become world fatuous and read throughout the known universe and Neasden. Complaints, please, on the back of a bottle to D Jarrett Esquire of Kings Cross.

Decisions, decisions

A chap of my acquaintance has offered to buy me half an 80386 machine. The one with the big disk and bigger reviews, DTPs, Mandrax prescriptions and suchlike — gets you out of jury service, etc. But then I see that Carrington is offering the Apricot upgrade to IBM AT-compatibility from £699 (*this is list price for the cheapest version of the upgrade; see full details and our review in Release 2.12 — Ed*).

"Don't get caught in the rush", expostulates Carrington. Then one notes in the computer press that heaps of clone-makers are planning 80386-based PS/2-type machines to run at 20MHz — that's 25 per cent quicker than the best Apricot.

What to do? What to *do*? Even if lovable old Nige Lawson stumped up for an EGA vertical monitor with Parental Guidance button, next week there will be an 80486 and after that bubble memory will come back with superconducted nodules in the stack register fangulopter. On the whole, wouldn't it be cheaper to take that second week in Torremolinos or to go shouting at Rev Paisley?

Well, in case you're holding your breath I'm going to see Dan Nelson and Ian Craven again. Singapore chaps. They've got good ideas. And they make decisions. Perhaps I should go into a partnership with them reviving the island's nitelife instead. Hey ho ... ◆

NCC software evaluation

The **National Computing Centre** has awarded its first five certificates under the much-argued **evaluation scheme for microcomputer business accounting software.** The recipients are **BOS Software, CMG, Plusmark, Systemics** and **Systems Axis.**

The NCC says "the service probably gives the UK a world lead in voluntary evaluation of accounting software", but it still doesn't compare with something like *Software Digest Ratings Newsletter* — the US *Which*-style end-user publication that tests packages. The original NCC scheme, drawn up with the Institute of Chartered Accountants in England and Wales plus some input from HM Customs & Excise, attracted a lot of flak from the fledgling Committee of Accounting Software Houses on the grounds of cost and time; the current version is much watered down, and for instance the testing can actually be done by the applicant — the NCC has produced a procedures manual for the purpose.

So what remains is a non-mandatory 'Seal of Approval' system with four main objectives. Probably the most important is also the least rigorous aspect — users are polled for their views on the practical usage of the package. But then the users would be reference installations nominated by the supplier; and obviously a package would have to *have* some users before the vendor could apply, so brand-new products don't get a look in.

The other aims are all a bit obvious. The package has to conform with the Excise's VAT requirements, and it has to provide all the accounting functions claimed for it; but then the buyer has some legal redress anyhow if either of these aspects is deficient. And the scheme finally checks that the software "provides facilities to ensure the completeness, accuracy and integrity" of its accounting functions, which could be the meatiest part of the report produced by the NCC on validated products.

In short, the scheme isn't really A Bad Thing — although there is some law relating to sale and contract, there are many loopholes; it's helpful to have a definitive statement of just what a package can do, particularly if suppliers can be persuaded to incorporate that into their explicit contractual terms. But it doesn't provide anything more than confirmation of the content of the software; there's no indication of quality or performance. And while the NCC expects that the scheme will be well-supported by the software developers — "the UK's software accounting industry will benefit from increased credibility in marketing its products around the world" — that still sounds to us like ambitious optimism.

The NCC has another such scheme in the pipeline, this one for payroll packages. Again, it was developed with various interested parties — Inland Revenue, DHSS, CASH and the BCS Payroll Group — and this one has also drawn some criticism. It has its share of industry supporters, notably the big boys; but several package developers share the views of Edward Landsler of Landsoft, who has circulated a critical letter to other payroll software vendors — "if this scheme goes ahead, you will be required to pay a private commercial organisation, the NCC, £400 per year for the privilege of staying in your own business ... I would appeal to every software house to boycott the scheme".

Objections centre on the NCC's role as a non-accountable profit-making commercial organisation and on the annual subscription rate (which is expected to increase anyhow). But it looks as though the DHSS's approval and the participation of the major players in this market will ensure the scheme's acceptance — probably to the detriment of the small software house.

Printers in Europe

According to market researchers **IDC**, sales of **printers** in Western Europe during 1986 increased by 37 per cent over the previous year. The **UK** represented the largest single market, with **393,000** shipments (but only a five per cent increase forecasted between 1986 and 1992). West Germany came second (390,600 units) followed by France (308,000). The countries for which IDC expects the fastest growth are Spain with 15.8 per cent and Sweden at 13.5 per cent; in both cases the market is still pretty young, of course.

... More matrix

The action in 80-column printers happens around the Epson **LX800** right now, which means print speed around 150cps and a price competitive with the LX800's **£275.** The the latest from **Citizen,** the **LSP100,** looks to have a good spec in this class: 150cps in draft mode, slightly faster NLQ (30cps against 25), a bit more buffer (4K rather than 3K), and a better price tag of £249. Interfaces are plug-in cartridges rather than Epson's built-in connectors. The other contender in specification terms is probably the No.2 seller in the class, the **Star NL10** at **£248** — 5K buffer, 30cps NLQ, but only 120cps in draft mode. Citizen is on **0895 72621.**

... Laser update

And the latest crop of **laser printers** includes the first from **Brother,** the **HL-8.** Brother claims a third of the daisywheel market, so it ought to know what it's doing in presentation-quality printers. Somewhat disappointingly, the HL-8 is a very conventional example — 512K memory expandable to 2MB, five resident fonts, 8 pages per minute, RRP £2,195. The only relatively unusual feature is a second cartridge slot for extra fonts. Brother is on **061-330 6531.**

Incidentally, Brother has joined the move to UK manufacturing — its new £4 million **Wrexham** plant opened on 1 October, starting with 3,000 printers a month and moving to 10,000 in 1988.

Olivetti has set up a new peripherals subsidiary called **Diaspron** and signed a 12-month contract for **Overture 110** printers that could be worth £1.5 million. The Overture is a decent little package and probably deserves more exposure.

Mekom has dropped the price on its **Kyocera F-1010** laser printer and announced a new top-end model. The F-1010 loses £300 for a new price tag of **£2,495** — is still a grand above the cheapest laser printers, but the F-1010 has 62 resident fonts and a megabyte of memory. The new **F-2200** is a dual-bin 10ppm printer with 1.5MB on board (can be upgraded to 3.5MB for really heavy graphics). At **£4,760** it seems a bit expensive for a non-Postscript printer. Mekom is on **021-454 2288.**

All at sea again

Readers may recall the unfortunate end of Apricot's last yacht-racing sponsorship, when skipper **Tony Bullimore** ran into rocks towards the end of a very successful season (APRICOT FILE Release 2.4). Bullimore is now back with an even cleverer trimaran, and so is Apricot's chequebook — the *Spirit of Apricot* will sponsored by "a six-figure sum" for the new season. Quoth Apricot marketing person **James Blackledge**: "*Spirit of Apricot* will reflect many aspects of Apricot Computers' philosophy — innovation, advanced technology, the will to win and be Number One" ...

User group for TAS

Yorkshire-based **Data Management Ltd** has established a good-looking user group for the under-sung TAS database system. **£36 per year** buys a quarterly bulletin, access to an on-line bulletin board (with public domain software available), an evenings-only 'hot-line' telephone support service, and a rolling 'guide to TAS for the inexpert user' which Data Management is writing. Details from **0904 760847**.

Tandata switches to Multicom

The MULTICOM comms package from **Richard Cumming Associates** has been around for a while — it's British, is very versatile, has always featured good support for viewdata as well as ASCII files, and has never achieved the recognition it deserved. This might change now that **Tandata** has elected to junk its own TAN-LINK comms package in favour of a version of MULTICOM with a Tandata-specific front end; TANDATA MULTICOM also appears to include the full facilities but doesn't mention all of them in its manual (no reference to the command file and editing facilities). In IBM format, TANDATA MULTICOM will be supplied free with the Tm602, priced at **£49** when purchased with another Tandata modem, or costs **£99** if you buy it alone. The full manual is available for another **£25**, so it still represents a discount on the RRP of **£150**. But Apricot and Sirius users — for whom the package was originally written, as it happens — will need a specially-configured version to be supplied with the full manual at **£124**. Tandata is on **0684 892421**.

Xen-i modem

We forecast that Apricot would be rebadging the rather fine Stradcom 2 internal modem from Dataflex; instead it's gone for a mild reworking of the **DaCom Unity Gold**, which will be two hundred quid more expensive at **£399**. The extra buys you 1200/1200 bps V22 operation (the Stradcom is limited to 1200/75 V23) and an on-board loudspeaker, plus some mods to ensure that the thing will work at the higher bus speeds of the Xen-386. It fits the half-length slot of all Xen-i models, comes bundled with Communiqué and a Telecom Gold registration, and has a modem driver which enables any software written for Apricot's previous in-house design of internal modem to work with the new Triple X (if there is any such software around).

WordPerfect promises

Via a carefully orchestrated series of press leaks, **WordPerfect Corp** has indicated that the next release of WordPerfect — **version 5.0** — will be something rather special. Key points are a complete code rewrite, support for PostScript, and true WYSIWYG; shipment should be around January 1988, and it's not impossible that Apricot versions will no longer be available ...

Efficiency at BT?

If **British Telecom** is to match current European levels of efficiency in management, organisation and quality of service, it will have to shed a third of its workforce — about **70,000** people — by 1990. And that ought to start as soon as BT gets over the major backlog of digitisation. So says *Uncrossing the Lines*, a report by market analysts **Alexanders Laing & Cruickshank**; their prognosis speaks of "indifferent" service quality for the rest of the decade, with brighter prospects for the 1990s following some reorganisation. Possible flies in the ointment include undue interference from OFTEL (demanding too much too soon) and competition from Mercury (BT is most vulnerable at the top end of the market, where Mercury is most active). The report also slams the "feeble" export performance of British telecomms manufacturers, comparing them unfavourably with the likes of LM Ericsson — "the UK companies have not only stood still: they have been left behind".

New Apricot mag

Apricot is publishing a new quarterly promotional magazine, "available to all users of Apricot microcomputer equipment". *Viewpoint* is very smart — though it doesn't look as though PageMaker was used — and runs to 16 tabloid-size pages, all of course promoting Apricot's current interests (LANs, DTP, IBM-compatibility). Interestingly, we heard that Apricot had to pay the big computer-industry publisher VNU for one-time rental of a mailing list from which to junk-mail Apricot users; seems Apricot itself doesn't have such a list, and VNU built one for promotion of *Apricot User* and *Apricot Systems* on the basis of some outdated customer names which Apricot actually supplied ... Contact Apricot direct (**021-456 1234**) if you'd like to register to receive *Viewpoint*. ◆

Date Prompt revisited

Steve Rainey dropped in our PCW Show stand to mention a misprint in the SuperWriter date prompt tip from him that we published back in APRICOT FILE Release 2.6. This is the correct prompt to place at the start of a batch file which loads SuperWriter:

```
PROMPT  $ej$ex1$eY8 $ep
        Today's date $d
        $eq$ey1$ek
```

Of course, we've printed the PROMPT like this for clarity — don't enter it on separate lines; use only Return once only, at the end. The original misprint was a lower-case y in the escape code to select the 25th line; it should be `$eY8`.