

# FROBNICATE

*FUN FUN FUN 'TIL DADDY TOOK THE KEYBOARD AWAY!!!*



- VILLAGE LIFE IN INDIA
  - ASSEMBLER
  - THE ACORN CODE
- AND MORE!!!

WebLoad II upload

Uploading file

Time to finish and download speed

0:13:24 (1511bytes/sec)

Hissing Sparrow

Summer 1997 Issue 14 £0

123

## Index:

Page 2 . . . .	Index.
Page 3 . . . .	Editors Page.
Page 4 . . . .	Village Life In India.
Page 6 . . . .	Assembler programming.
Page 12 . . . .	Econet - a deeper look.
Page 13 . . . .	Diary of a demented hacker.
Page 14 . . . .	DIGIWIDGET.
Page 15 . . . .	Argonet (#2).
Page 16 . . . .	Update to Acorn machine list.
Page 19 . . . .	The Acorn Code

## Credits:

Editor . . . . . Richard Murray    [*froblicate@argonet.co.uk*]

Contributors . . Richard Murray.  
Village Life article by Ben Hartshorn.  
Machine List by Philip R. Banks.  
Acorn Code by Quintin Parker.

Graphics . . . . . Richard Murray.  
Village Life graphics by Ben Hartshorn.

You may print and/or distribute this document provided it is unaltered.

The contents of this magazine are © Richard Murray for legal reasons. All copyrights and/or trademarks used are acknowledged. Opinions stated are those of the article author and do not necessarily represent the opinions of Froblicate, BudgieSoft or Richard Murray. All reasonable care is taken in the production of this magazine, but we will not be legally liable for errors, or any loss arising from those errors. As this magazine is of a technical nature, don't do anything you are unsure of. Reliance is placed in the contents of this magazine at the readers' own risk.

Froblicate is managed by "Hissing Spinach",  
the publishing division of BudgieSoft UK.

Comments? Submissions? Questions?

**[froblicate@argonet.co.uk](mailto:froblicate@argonet.co.uk)**

Or visit our web site (as seen in Acorn User)...  
**<http://www.argonet.co.uk/users/rmurray/froblicate/>**

# EDITORS PAGE

This issue has seen a few changes. The first being the WWW site and dedicated email address; the second being DIGIWIDGET. Trust me, it isn't easy to make stuff like that with !Draw. :-)

Okay, well, the PC is on in the background. Reinstalling Windows95 (again!) and I'll probably kill of 90% of what I have and reinstall the stuff I'm likely to actually NEED. Hehe...

Thanks to Farnell I have data on the Econet interface ICs, you can see the beginnings of this in the Econet - A deeper look article.

Ben Hartshorn contacted me via email asking if I would like to say anything about their "Village Life in India" CD-ROM. So I asked him to write up about it. You can see this on the next page.

When I took the demented hacker out for a rest; I got flamed. It seems he has a kind of cult status. Hehe... So he is back and he has asked me to invite you to email him challenges and ideas. He asks for you to just send it along to the frobnicate@argonet.co.uk address - hacking it won't be a problem. :-)

I present something passably *useful* to do with assembler programming, and so finish up with Quintin's Acorn Code.

That's Frobnicate #14 for you.

\_\_\_\_\_  
/\_\_\_/ Personal: rmurray@argonet.co.uk  
/ \ichard. Editorial: frobnicate@argonet.co.uk  
http://www.argonet.co.uk/users/rmurray/

## WIN A COPY OF BUDGIEMGR (worth £15)

YOU could win a copy of BudgieSoft's Econet management software.

- Closing data is July 12th 1997 at 23:59:59 hours.
- All correct entries will be put into a virtual hat and computer-selected at random.
- In the event of NO correct entries, I will select the most amusing entries.
- You MUST give all your details. Entries with no snail-mail address will be ignored (who would I post the results to, eh?).
- There is no cash alternative.
- Richard Murray's decision is final.

Simply answer 12 Econet-related (sort of!) questions and YOU could win a copy of BudgieMgr!!!

To enter, either:

- Complete the form included with Frobnicate and email it to frobnicate@argonet.co.uk
- or:
- Go to <http://www.argonet.co.uk/users/rmurray/> and follow the link to BudgieMgr (or Econet) and then go to the competition page.

**GOOD LUCK**

# Village Life in India



Here, Ben Hartshorn the Project Manager of ACTION AID describes some of the trials and tribulations of the production of "Village Life in India" which is a multi-platform educational CD-ROM.

## Why "Village Life in India"?

Some time back Cambridge University Press (CUP) saw "Discover India", a CD-ROM title ACTION AID had developed for use by teachers of geography at Key Stage 3-4 (11-16 years) of the National Curriculum. They thought it could be developed for a younger audience, editing some sections, cutting others and introducing one or two new ones.

The result would be published in the Cambridge Primary Geography series, under the title "Village Life in India". It was to support and extend the package of a pupil's book, teacher's book and poster pack.

The Nilgiri Hills area of southern India is popular amongst primary schools teachers, based on the success of the "Chembakolli" photopack, which has now sold over 20 000 copies. The "Village Life in India" materials were intended to compliment this, and widen the area of study from the original village (as demanded by the National Curriculum).

## More than a Mac/ PC thing

The main snag was that the publisher required the title to operate not only on Windows PCs, and under MacOS, but also on CPUs running RISC OS. We needed to be able to re-author the existing material, previously put together in Macromedia Director on a Mac and ported to the PC, for a whole new platform.

Having looked into Genesis, and had offers of coding in Basic, the only realistic route was to jump into the beta-world of HyperStudio, re-coded for

RISC OS from the Mac and PC versions by TAG Developments.

TAG were a source of help and floppy disks! HyperStudio certainly got the artwork to the Acorn. Stacks could be saved from the Mac (unless, like us, you had an AV machine, where an extra line of pixels would appear down the middle of the sprite). We went back to an old IISI for this part of the process. At least it worked.

Much of this was done without manuals to hinder the process. In fact, the Acorn manuals, when available, contained most of the functions implemented for Mac and PC users, which didn't show on the new operating system (or not for another six months).

## HyperLogo active

Meanwhile, our Lingo programmer got to grips with HyperLogo, a netherworld of oddities, without the functionality expected by users of Director. Then again, for £700 less, you wouldn't expect the same.

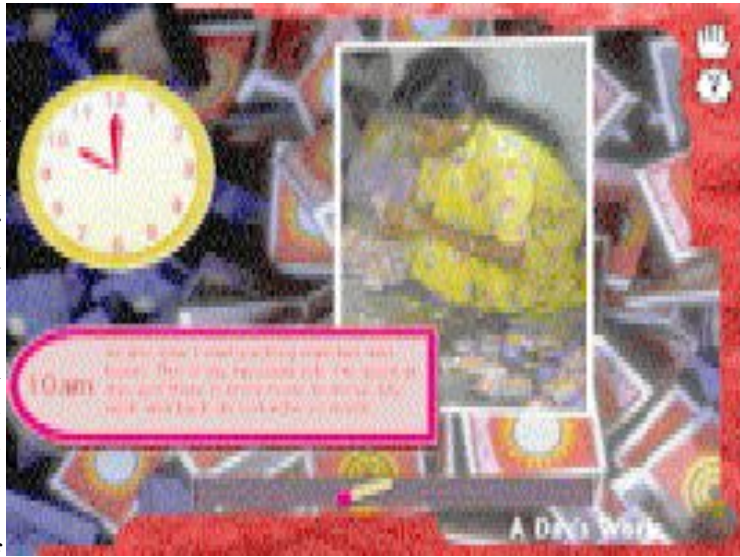
Twelve versions and four months later, we had a working, stable player. Gone were the glitches like, the HyperStudio stack could be positioned anywhere on the desktop, but Replay files could only play at x, y values that were divisible by 4. You've guessed it, the video jumped on playing! A work-around was to allow the stack only to move by four pixels in any direction. Our artwork was saved.

Of course, even TAG couldn't help us on a sunny day. Coming from just the right angle, the sun shone through the gaps in the plastic casing of the mouse, upsetting the light sensors. The mouse gave up. A blind was called for.

Whilst this happened, and routines became whole new lines of code, and TAG kept implementing features that should have been available but for some reason didn't quite ship, two big issues arose.

Firstly, QuickTime or Replay? Re-compress everything as Replay files? At £15/minute, perhaps not. Export pics and rebuild as movie files? After three seconds (36 frames) of digital video, it was apparent a better way forward must be found. Eventually, a codec became available that would recognise QuickTime files, and play them back under Replay. We licensed it straight away.

Secondly, nobody had put together a triple hybrid CD-ROM where media components were shared between the Mac and ISO (PC) parts of the disk. This was essential, as space was running out. In fact, without it we would have to move to Plan B, and there wasn't one of those. For the solution, we have to thank Eesox, who re-thought the problem, and managed to add our Acorn data to the ISO part of the disk. All users would see what they expected to see.



Taking a RISC

Whilst development progressed we had to license Replay from Acorn. No, they didn't have the logo in a format that a Mac (or PC) could read, and we didn't have the software to convert it. Our RiscPC was, in fact, on hire, and we were looking to invest as little as possible on something that was soon to be back in its box.

For the packaging, Acorn were also unsure about how they write their titles. RISCOS, or RISC OS? A small point, but one that any sizeable company sorts out and then doesn't worry about.

We were able to use the same excellent music as on the first title, and Apple had dropped their charge for licensing QuickTime (not that we paid it the first time, anyway). Macromedia had given up trying to keep records of who distributed their player. Now, that's sensible.

We began testing. RiscPC, no problems. Plenty of

RAM, fast(ish) processors, I got to quite like them. That is, apart from when HyperStudio didn't switch off a sound, which would play until you restarted your machine. But, A5000s. Hmm. With enough

RAM (and CDFast, again from Eesox) they could cope. At least they were better than the A7000, which played the sound at half the speed. Half the speed equals twice as long, which when it wouldn't stop was torture. We didn't expect anything else, though, given the installed speakers cost 16p for a pair. TAG found a way round it, again.

We ended-up with something close to the original, and beat our own path, given nobody had been there before. Our HyperLogo expert implemented some functions we, and TAG, thought were lost. Timed events, unimaginable at first, happened.

The only problem we couldn't get round, was school users trying to drive the disk on a 4MB RAM CPU. The system? 1.5MB. HyperStudio Player (if your stacks are worth looking at)? 1.5MB. Add in some for audio, screen and CDFast, and there's nothing left for Replay. Still, most primary schools are buying PCs for multimedia work. Now, we just have to explain how to load QuickTime.

"Village Life in India" won the Times Educational Supplement "Mixed Media Award" at this years show. The publishers were and are happy.

"Village Life in India" CD-ROM costs £56.50 (including postage and packing) and is available from:

Development Education,  
ACTIONAID,  
Chard,  
Somerset TA20 1FA.

Email : [deved@actionaid.org.uk](mailto:deved@actionaid.org.uk)  
Website : <http://www.actionaid.org>

# Assembler programming

Our past articles have given a brief introduction to the ARM processor, registers and a basis of how ARMcode fits together.

Anybody who has programmed 6502 assembler or PIC assembler should find learning ARM assembler is fairly simple. Indeed, there are only a handful of opcodes to learn – the conditionals can be applied to virtually any instruction. The Branch with Link instruction also allows for primitive subroutines to be created, and array accesses are all possible.

What we shall do this month is jump straight in with an example program. Something that you will find useful (I mean – what *is* the point of writing “Hello World” a few thousand times? When you learn to code it is much better to learn on *useful* projects).

This issue’s project is a 16-colour screen mode compressor. It can be adapted to other modes, but I’ll leave that as an exercise for the advanced. After all – ChangeFSI does a marvellous job of reducing the number of colours in an image, and for demos with ‘real life screenshots’, 16 colours is usually adequate (unless you are plugging the latest 16 million colour game you wrote – in which case you won’t need this!).

Average savings depend on the image. It does badly with dithered images, it does well with images that don’t have loads of fiddly things (a screenshot of a program running in the DeskTop for example). It has been designed to work with MODE 12 and MODE 35. For others, it is a case of suck it and see.

Right. Well... In the old days when it was safe to leave your car unlocked and your daughter unchained, screen memory started at &2000000 and worked upwards. You needed to *know* a few modes such as 12 and 13 and 15... Like, who do you know that uses MODE 8?

Modern life, however, is not so simple. We would be foolish to assume the screen start position and/or size. We would also be foolish to assume the ‘default’ palette. So all of this information has to be examined. 16 colour modes use 4bpp, or 4 bits per pixel. One byte equals two pixels.

The compression system is appropriately crude. We will work on a “better” scheme later. However for now we shall be using a kind of Run-Length-Encoding. We treat the screen memory as a sequence of bytes, and we encode it into a set of byte counts and byte values. For example, assume screen memory is:

```
11111111114235111112111111122222222222222111111111111
```

This would be a bunch of 1’s followed by a 4, a 2, a 3, a 5, more ones... Which becomes:

```
A114121315511271F2C1
```

This has reduced the stream to approximately a third of original size. The “A1” means 10 x 1 (A is 10 in base 16). Then 1 x 4, 1 x 2, 1 x 3, 1 x 5, 5 x 1, 1 x 2, 7 x 1, 15 x 2 and finally 12 x 1.

If you don’t get it, study the above for a few moments. It is quite simple once you understand. By way of example, 11112246885788 becomes 4122141628151728; and you can see how it becomes inefficient with oft changing bytes (such as dithered images), the output is two bytes larger than the input.

On the next page, we begin.

Our format is to be defined as:

Ident: "BudgieScrn (compressed 1)" <newline> which is 26 bytes.  
 Mode: 1 byte giving mode number.  
 Palette: 16 entries, six bytes each. Giving red, green and blue values for first flash colour and second flash colour or each entry.  
 Image: Data as defined above.

We will leave the screen size validation up to the loader instead of encoding it into the image.

So... Let us begin:

```
REM >BudgieSoft screen saver © 1997 Richard Murray
REM V1.01 [Froblicate #14]
:
ON ERROR PRINT REPORT$+" at "+STR$(ERL/10) : END
:
DIM code% &400
FOR pass% = 4 TO 6 STEP 2
P%=0
O%=code%
[ OPT pass%
```

This is a fairly standard beginning. The thing to note is that we are using offset assembly. In this way, our code is compiled as a transient utility instead of an "absolute" application. It should also be loadable. The divide-by-ten in the error message is so that 'real lines' match up to !Edit lines, so if you get an error just press F5 and enter that number.

```
MOV R7, R14
```

We now 'save' the return address in register 7. This is so that it does not get overwritten by BL (Branch with Loop) stuff. Note, if you want to fiddle with the code – remember register 7 is *special*.

```
MOV R0, #&8C
SWI "XOS_Find"
BVS exit
MOV R6, R0
```

Our program is designed to be called as "SSaver <filename>". Conveniently the OS\_Find call (opens/closes files) has a flag to look for the filename in the entry parameters. So this stuff attempts to open our file with write access. If it fails the V flag (oVerflow) is set. The BVS command (Branch if V Set) will carry us out to a safe exit. Otherwise we can assume all went well, and move our file handle into register 6 for safe keeping.

By now we have our file open and are ready to start building up our image data.

```

MOV    R0, #2
MOV    R1, R6
ADR    R2, header_text
MOV    R3, #26
SWI    "XOS_GBPB"
BVS    error

```

This sets up a call to OS\_GBPB to write the header; which is defined at the end as “.header\_text”. The BVS takes us to an error handler in case we don’t have write-access after all!

```

MOV    R0, #&87
SWI    "XOS_Byte"
MOV    R0, R2
MOV    R1, R6
BL     write_byte

```

If you were to look up OS\_Byte &87 in a manual, you might wonder about my sanity. After all – what relevance could “get character at cursor position” have?

Frankly – none. :-). However in register 2 it returns the current MODE number, and THAT is what we are looking for. We then set up a call to a routine that outputs a byte for us, defined later as “.write\_byte”.

```

MOV    R5, #0
.col_loop
MOV    R0, R5
MOV    R1, #16
SWI    "OS_ReadPalette"
MOV    R1, R6
MOV    R0, R2, LSR #8      ; red 1st
BL     write_byte
MOV    R0, R2, LSR #16    ; green 1st
BL     write_byte
MOV    R0, R2, LSR #24    ; blue 1st
BL     write_byte
MOV    R0, R3, LSR #8     ; red 2nd
BL     write_byte
MOV    R0, R3, LSR #16    ; green 2nd
BL     write_byte
MOV    R0, R3, LSR #24    ; blue 2nd
BL     write_byte
ADD    R5, R5, #1
CMP    R5, #16
BLT   col_loop

```

This looks complicated and I’m not going to explain it to verbosely. However what we are trying to do here is to read the palette (care of OS\_ReadPalette). It returns two registers for the six colour values. As we are using 32 bit words, we can cram several 8 bit values into one word. Hence the LSR, which is a Logical Shift Right. LSR takes a value (say %110110100) and shifts it (say two places) to produce an output (such as



%001101101). We loop through the 16 colours and extract the six values for each, writing them to the file as we are going along.

The header is now complete.

```

ADR    R0, vdu_block
ADD    R1, R0, #12
SWI    "OS_ReadVduVariables"
LDR    R2, [R1]
LDR    R3, [R1, #4]
ADD    R3, R3, R2
SWI    "XOS_RemoveCursors"
    
```

Even though we are not going to write the screen dimensions to the file; we do need to know where the screen starts and ends. This calls "OS\_ReadVduVariables" asking for the screen base address and the size of the screen. Adding the two gives us our result, with the data held in the memory area defined later as ".vdu\_block". While we are at it, we switch the cursor off.

```

MOV    R1, R6
.main_loop
LDRB   R0, [R2]
BL     write_byte
MOV    R4, #1
.loop2
LDRB   R5, [R2, #1]!
ADD    R4, R4, #1
CMP    R5, R0
BNE    skip
CMP    R4, #&FF
BLT    loop2
.skip
SUB    R0, R4, #1
BL     write_byte
CMP    R2, R3
BLT    main_loop
    
```

Here is the main loop. What it does is it outputs the value of the current byte (ie: the colour) and then it reads through the memory looking to see how long it continues for. As a byte can only hold 256 different values, it loops up to a maximum of 255. Once a value has been established, it will write it out to the file and then branch back to the main loop; repeating until the end of screen memory is reached.

```

MOV    R0, #&20
BL     write_byte
MOV    R0, #&20
BL     write_byte
    
```

A really early version of this software messed up by trying to read a little beyond the end of the screen (and

hence the end of the file), so it became practice to add two spaces to the end of the file – just in case – even though the <cough>bug</cough> was fixed.

```
MOV    R0, #0
MOV    R1, R6
SWI    "XOS_Find"
SWI    "XOS_RestoreCursors"
```

This wraps up, closing our file and restoring the cursor. It falls through to “.exit” which follows next.

```
.exit
MOV    PC, R7
```

Leave the program – places stored return address (in register 7) into the Program Counter (register 15).

```
.write_byte
SWI    "XOS_BPut"
MOVVC  PC, R14
```

Simple routine to write a byte of data. Not terribly efficient, however you could change it to move the file handle here instead of beforehand (using MOV R1, R6). It returns by way of function returning (move return address into Program Counter); but only if the V flag is clear. If something went wrong, however, then we fall through to “.error” below.

```
.error
MOV    R2, R0
MOV    R1, R6
MOV    R0, #0
SWI    "XOS_Find"
SWI    "XOS_RestoreCursors"
MOV    R0, R2
ORR    PC, R7, #1<<28
```

This closes the file, restores the cursor and exits with the V flag set (error condition). It is the ‘standard’ error handler for this program.

```
.vdu_block
EQUUD  149
EQUUD  7
EQUUD  -1
EQUUD  0
EQUUD  0
```

This is the block for the OS\_ReadVduVariables command. The 149 and 7 are the codes for the values we wish to read. The -1 terminates the list. The two 0’s are where the information is placed by the OS.

```
.header_text
EQU$ "BudgieScrn (compressed 1)"
EQU$ 10
EQU$ 0
ALIGN
```

This is the header.

```
]
NEXT
:
OSCLI("Save <Obey$Dir>.Saver "+STR$(code%)+ " "+STR$(0%))
OSCLI("SetType <Obey$Dir>.Saver &FFC")
:
END
```

And finally the denouement. Close the loop, save the file and exit.

There you have it!

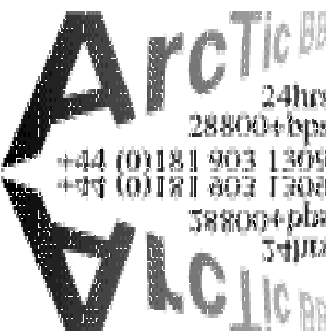
To use it, try an Obey file similar to:

```
| Obey file to run SSaver
|
ScreenLoad <Obey$Dir>.mypiccy
|
<Obey$Dir>.SSaver <Obey$Dir>.comp_piccy
```

In the next issue, we shall create a loader. Between now and then, why not see if you can fiddle with the above to make it work in reverse?



**NetMail**  
 A fully Fidonet compatible network based mailing system.  
 Exclusively available on the Acorn platform.  
 Coming soon from Squirrel Solutions  
 For further info E-Mail richards@arcticbb.demon.co.uk



**Arctic BB**  
 24hrs  
 28800+bps  
 +44 (0)181 903 1305  
 +44 (0)181 803 1305  
 58800+bps  
 5400

F r o b n i c a t e  
**THE MAGAZINE**  
**FOR ACORN TECHIES**

**YOU** could advertise here.  
 Contact [frobnicate@argonet.co.uk](mailto:frobnicate@argonet.co.uk) today.

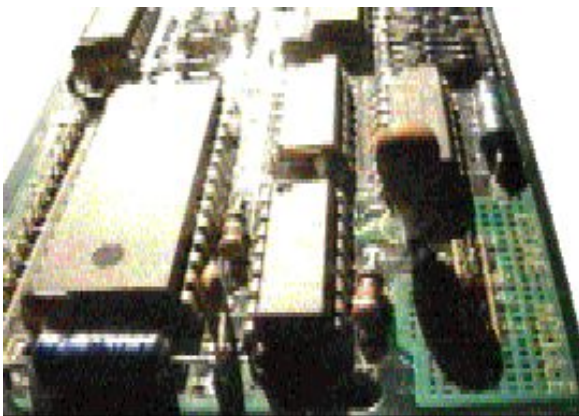
# Econet - a deeper look

ECONET, Acorn's Economy Network system (designed originally for BBC micro type computers) is based around multiple EF68B54 ADLCs connected to a wire. This wire carries the serial data signals as a balanced pair, and includes a balanced pair carrying clock signals to sync everything.

The Advanced Data-Link Controller is clocked at 2MHz. It is a 6800 series IC.



Econet operates by linking Econet almost as a fifth podule. The data is linked to the DB0-DB7 on the data bus. It uses LA2 and LA3 of the address bus (similar to the serial controller and floppy controller). The remaining parts (read/write, Econet FIQ etc) link into the IOC.



The diagram below shows the relationship between 6854 pins and the Econet interface.

68B54	Connects to
0v	<input type="checkbox"/> 0v
RTS	<input type="checkbox"/> CTS, and helps Econet flow control
RxD	<input type="checkbox"/> Econet data+ and data-
RxC	<input type="checkbox"/> TxC, and helps Econet flow control
TxC	<input type="checkbox"/> RxC, as above.
TxD	<input type="checkbox"/> Econet data+ and data-
IRQ	<input type="checkbox"/> Net FIQ on IOC
RESET	<input type="checkbox"/> RST (system reset)
CS	<input type="checkbox"/> S2 (IOC ADLC select)
RS0	<input type="checkbox"/> LA2 (address bus #2)
RS1	<input type="checkbox"/> LA3 (address bus #3)
R/W	<input type="checkbox"/> WBE? (IOC?)
E	<input type="checkbox"/> CLK2
Vcc	<input type="checkbox"/> +5v
D7	<input type="checkbox"/> BD7
D6	<input type="checkbox"/> BD6
D5	<input type="checkbox"/> BD5
D4	<input type="checkbox"/> BD4
D3	<input type="checkbox"/> BD3
D2	<input type="checkbox"/> BD2
D1	<input type="checkbox"/> BD1
D0	<input type="checkbox"/> BD0
RDSR	<input type="checkbox"/> Not connected
TDSR	<input type="checkbox"/> Not connected
FLAGDET	<input type="checkbox"/> Not connected
LOC/DTR	<input type="checkbox"/> Not connected
DCD	<input type="checkbox"/> Connected to RxC/TxC via IC
CTS	<input type="checkbox"/> Connected to RTS, above

In following articles, we shall be looking at the 6854 structure, ADLC frames, Econet frames, the MC14411 clock generator and more...

# Diary of a demented hacker

It's like this really amazing thing happened today. I ran out of money. The people running my borrowed credit card sent me this really nice letter instructing me to cut my card into several pieces; and a few days later I got a not-so-nice letter instructing me to cut myself into several pieces. I fiddled around with my modem and paid them back. Oh, I also terminated the contract of the fool stupid enough to sign their name on the letters.

It is a quandary. I know the principal is watching over me to look for one mistake. I could, if I wanted to impress, turn up to college in a Porsche. There's a heck of a lot of unnamed money sitting in Switzerland. But then the principal would come down heavy on my case and I can't do anything to him. He has set up verbal and written instruction with loads of people such as the college board, the bank, the government... He tells them to use paper records and *not* to trust the computer. They probably think he is a nut, but it effectively stops me from messing up his life. I tried it a few days ago. I set his water rates to £250 below the zero mark. They looked it up in their hardcopy, told him not to worry and then took their computer apart for a few days.

Anyway... It looks like I have to get myself a part-time job that I can do after college. The black market looked inviting, but it turned out to be a bit boring. Money is left in one suitcase, weapons in the other. They are about two miles apart. It always goes down at night. Boring. I like interaction. It gives me a chance to size up the "customer" and maybe make them squirm a little. After all, if the customer is always right – why the hell are they asking you for advice?

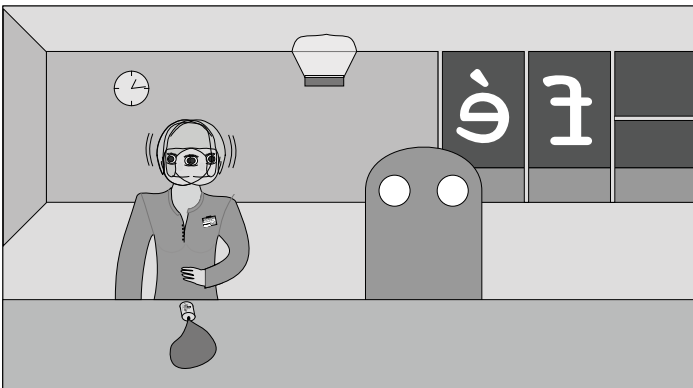
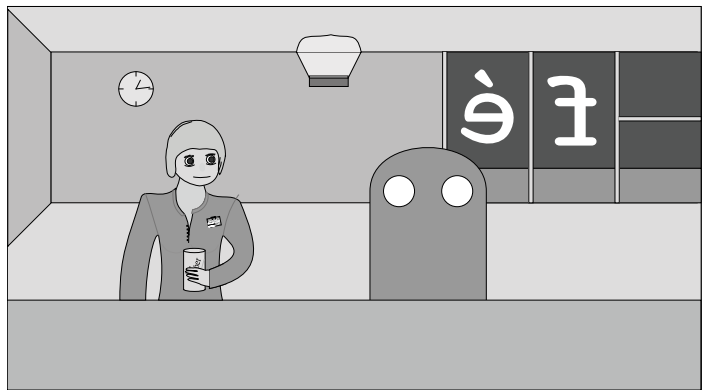
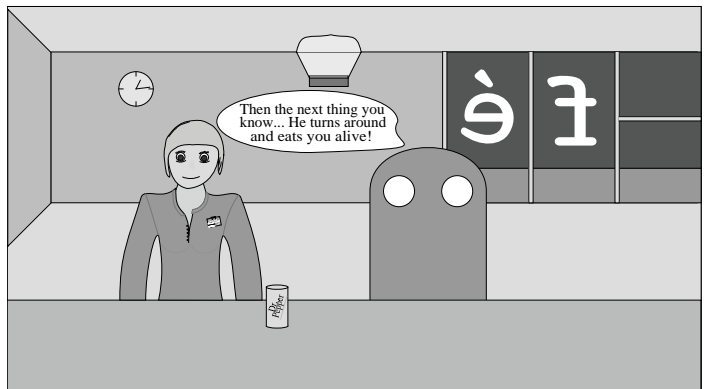
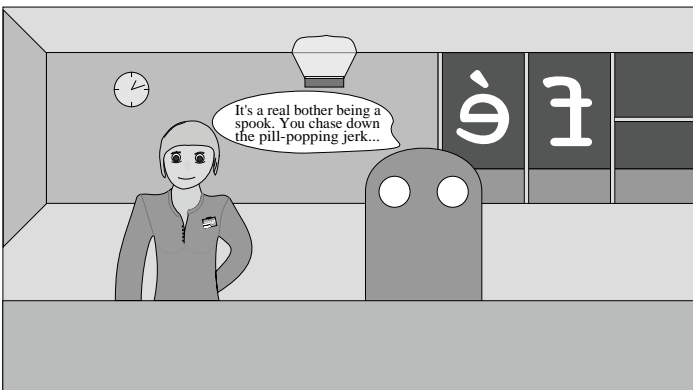
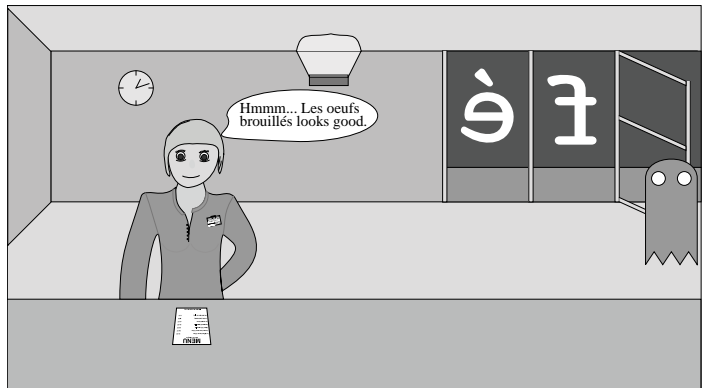
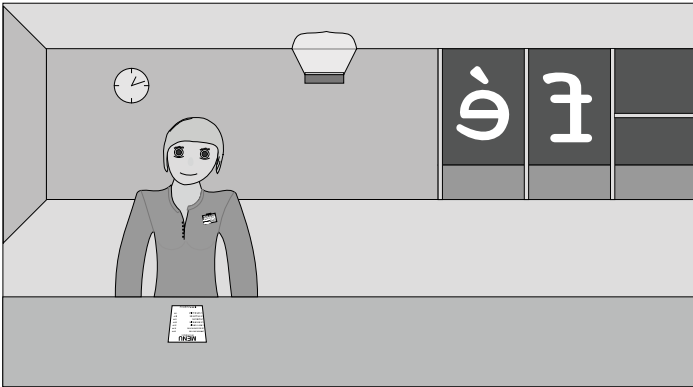
I walk down the high street. The bank trains internally. The estate agent don't use computers. The repro shop uses horrible horrible Macs from an obscure era when computers and LASER printers had a processor in common. The chemist are looking for somebody. Nosing around, you learn all kinds of cool things like how to apply cream to boils. You don't, you just have to tell the customer (the customer who is always right, remember?). Time to blow chunks and look elsewhere.

On my way home I pass the council building. A great shining edifice to taxation. This wonderful fully air-conditioned multi-storey hole built with our tax money. It'd be a piece of cake to sign up for unemployment benefit, housing benefit, family benefit and a hundred and one other benefits. Could easily rake in £250 a week. But I'm highly paranoid about the government. They have resources I can only dream of and they have the official secrets act. Like all the bad movies say, "People have accidents". And who would miss me?

I pop into the newsagent to pick up the latest Acorn User. There's a nice bit about filling blank spaces on the Internet, not to mention more 3D stuff. The cash till isn't taking SWITCH cards so the girl taps out the number by hand. Six, seven, nine, five, six, zero, one.... I am busy scribbling the number inside the Acorn User. Expires November 1998. Issue three. I close Acorn User and impatiently place my magazines on the counter. Discreetly I look over. It's a NatWest card. Good grief, what kind of name is that? Ah... He's a Doctor. I size the guy up. He is wearing a grey woollen suit and expensive highly polished shoes. Tucked into his jacket pocket is a contactless ID card with a barcode on it. He has bought twelve copies of New Scientist. This guy looks good for a few hundred at least.

Later that night I hang up the phone and rip a mouth-sized chunk out of my pizza, having just bought a shiny new X2 modem and a 32Mb SIMM – payment by SWITCH.

# DIGIWIDGET



# ARGONET (#2)

Following on from the last review of the Argonet package, I have been sent a user guide (including oddly enough an Acorn wired serial cable) and the WWW uploader (& soTIMM) have been made public.

The user guide is written for people new to the software. It looks professional and covers the package without getting overly technical. It still has a few things missing – for example it does not cover the telnet when operating in “finger” mode, nor does it explain the depths of WebVoyage configuration. However for introducing the Internet to non-techies who don’t want to wade through a lot of rubbish; it does a good job.

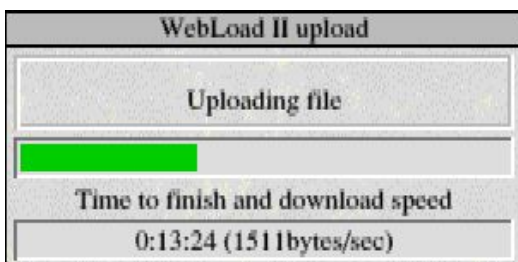
I, personally, think the single most lacking feature in the guide is a big notice in the front saying:

TURN OFF THE SEND EMAIL ADDRESS TO HTTP SERVER.  
IF YOU SUBSCRIBE TO ALT NEWSGROUPS, FAKE YOUR HEADERS  
FAILURE TO COMPLY WILL RESULT IN HUNDREDS OF  
UNSOLICITED AND JUNK EMAILS ARRIVING TO YOUR MAILBOX.

Still, maybe next release. :-)

Also included was the Rough Guide to the Internet. A nice touch.

WebLoad 2 mating with soTIMM (son of TIMM) is a gizmo designed to allow you to update your WWW site by point & click. It supports



updates a doddle. No fiddling with ftp. It isn’t perfect, but it is like 100000% better than the previous method.

subdirector -ies and partial site updates. It is so very easy and it really does make site

Added to this, Argonettors will soon be getting a copy of Acorn’s new WWW browser. Currently, 30 students have been taken on to test it – and the new WindowManager (supporting windows within windows) to destruction.

When will it be released?

The virtual Smarties jar is about the size of a can of Dr. Pepper. It is filled roughly two-thirds. The time until Acorn release the new browser is the number of Smarties in the virtual jar divided by the air temperature in the jar (in Kelvin) multiplied by the number of kilohertz at the upper end of the Long Wave radio band.

Jonathan Duddington (of Speak fame) has written a new integrated news/mail viewer/editor (currently named !Pluto (*wrong planet!* <g>)) which I have been privileged enough to betatest. I hope soon to bring you further details on this software, but first glance it is looking good.

Other third party add-ons, briefly. I have written a multiple-function add-on called Foxy, and a mail processor called FetchMail. Details on my site.

Christina West (<http://www.argonet.co.uk/users/xina/>) has the ultimate in call costers. It provides on-line updates of money burned and an optional total at the end. It keeps track down to several decimal places. Also on Xina’s site is information on getting into IRC – International Relay Chat (called CB simulator by some). It is like the old multiuser chats on a BBS; except with many people across the globe.

If you have any useful add-on software you would like detailed, contact me. Also if you are using a little something you find invaluable.

By the way, for the Smarties jar – I make it approximately 55.

# Update to Acorn machine list

Maintained by Philip R Banks.

<http://alpha.nedernet.nl/~bigben/MachineLst.html>

Previous machines covered in an earlier issue.

## Acorn RiscPC series

This is the next generation and architecture of machines – superceeding, but compatible with, the Archimedes range of machines. All of them feature a highly configurable and modular system that makes a bewildering variety of options available. All are founded on the ‘second generation’ chipset featuring VIDC20, IOMD and the newer ARM6, and better, cell processors. This new range was launched on the 15th of April 1994 with the RiscPC 600 series of machines.

The new machines feature the processor card option, the concept of which was first shown in the A540 as well as a unique second processor slot allowing the machines to have two processors in the system, at once, of radically different types. Simply by slipping a 486 chip in, on an appropriate board of course, Intel based software can be run on the machine adjacent to native ARM programs. Both processors share the system resources and can be allocated memory and the like to use. Memory management has been improved with memory paging always being done in 4K pages.

Further more the podule interface has been extended with DMA to and from podules, extended addressing, 32bit data pathways from the IO system as well as a vastly expanded memory map for each podule. Realtime video from the IO system becomes a reality with high speed and data transfer applications being boosted considerably. These machines are to be the ones to carry Acorn for the next three to six years and look set to do so nicely.

## Risc PC 600 series

The initial set of machines that launched the RiscPC range. Initially they all featured 30 MHz processors, but as of the Launch of the Risc PC 700 the 600 series machines all feature a 33 MHz ARM 610 instead.

As of the start of 1996 the base model of the RiscPC 600 was a 4Mb DRAM machine with a 425 Mb hard drive.

### *RiscPC 600 2M HD210*

2048Kb Memory. (No VRAM)

Advanced Disc Filing System by default with a High Density drive.

IDE interface by default. (210 Mb.)

ARM 610 processor (33 MHz)

16 MHz RAM.

2048k OS in ROM.

~26 MIPS. (Arbitrary estimate.)

### *RiscPC 600 5M HD210*

4096Kb Memory. (1Mb of VRAM)

Advanced Disc Filing System by default with a High Density drive.

IDE interface by default. (210 Mb.)

ARM 610 processor (33 MHz)

16 MHz RAM.

2048k OS in ROM.

~26 MIPS. (Arbitrary estimate.)

### *RiscPC 600 9M HD420*

8192Kb Memory. (1Mb of VRAM)

Advanced Disc Filing System by default with a High Density drive.

IDE interface by default. (420 Mb.)

ARM 610 processor (33 MHz)

16 MHz RAM.

2048k OS in ROM.

~26 MIPS. (Arbitrary estimate.)

**A7000** - a successor to the A4000 machine, this is a ‘cut down’ Risc PC intended for school/home entry as a cheaper alternative to the RiscPC. Both this and the RiscPC 700 are the first machines to feature



RISC OS 3.6. Unlike the RiscPC the A7000 has no second processor slot, only one DRAM socket and no VRAM capabilities. This limits the memory expansion of the machine to 128Mb + whatever memory is soldered on the motherboard.

Launched 12th of July 1995

A7000 Net  
 2048Kb Memory.  
 Advanced Disc Filing System by default with a High Density drive.  
 Ethernet interface in place of hard drive.  
 ARM7500 processor. (32 MHz)  
 16 MHz RAM.  
 4096k OS in ROM.  
 ~25 MIPS. (Arbitrary estimate.)

2048Kb Memory.  
 Advanced Disc Filing System by default with a High Density drive.  
 IDE interface by default. (425 Mb.)  
 ARM7500 processor. (32 MHz)  
 16 MHz RAM.  
 4096k OS in ROM.  
 ~25 MIPS. (Arbitrary estimate.)

4096Kb Memory.  
 Advanced Disc Filing System by default with a High Density drive.  
 IDE interface by default. (425 Mb.)  
 ARM7500 processor. (32 MHz)  
 16 MHz RAM.  
 4096k OS in ROM.  
 ~25 MIPS. (Arbitrary estimate.)

**Risc PC 700** series is the second generation of Risc PCs to be shipped. They feature the newer ARM710 processor, 16 bit sound as default with audio mixer on the motherboard and RISC OS 3.6. Not so much a major advancement as an incremental improvement of the RiscPC to keep pace with the computing world.

Launched 12th of July 1995.

*RiscPC 700 4M HD425*  
 4096Kb  
 Advanced Disc Filing System by default with a High Density drive.  
 IDE interface by default. (425 Mb.)  
 ARM710 processor. (40 MHz)

16 MHz RAM.  
 4096k OS in ROM.  
 ~32 MIPS. (Arbitrary estimate.)

*RiscPC 700 5M HD425*  
 4096Kb (1 Mb of VRAM)  
 Advanced Disc Filing System by default with a High Density drive.  
 IDE interface by default. (425 Mb.)  
 ARM710 processor. (40 MHz)  
 16 MHz RAM.  
 4096k OS in ROM.  
 ~32 MIPS. (Arbitrary estimate.)

*RiscPC 700 10M HD850*  
 8192Kb (2 Mb of VRAM)  
 Advanced Disc Filing System by default with a High Density drive.  
 IDE interface by default. (850 Mb.)  
 ARM710 processor. (40 MHz)  
 16 MHz RAM.  
 4096k OS in ROM.  
 ~32 MIPS. (Arbitrary estimate.)

### Acorn Diversification

Acorn, around October 1995, restructured itself into separate companies. Consequently the originating company will be noted by the machine model name, in brackets, from here on down.

**Network Computer** (Acorn Network Computing) is the production implementation of the Reference Standard that Oracle commissioned Acorn to specify for Network Computers. In common with the philosophy the default NC does not have any storage devices beyond a smart card, but can be upgraded as required by the customer.

Launched August 1996.

4096k Memory (expandable to 16384kb)  
 IP Stack by default, capable of using extra drives as required.  
 Smartcard interface by default.  
 ARM 7500FE processor. (40 MHz)  
 12 MHz RAM.  
 4096k OS in ROM.  
 ~35.9 MIPS.

**Set Top Box 2** (Online Media), an advanced networked computer designed to provide an interactive TV station in the home. These machines require high capacity networks to operate and as such are, as of early 1997, limited mostly to trials in selected areas. These machines feature hardware MPEG decoders as well as an infrared remote for controlling it. Strictly speaking these machine are no longer quite computers but are more converging into versatile home appliances.  
 Launched July 1996.

- 2048k Memory (expandable to 32768kb)
- CDFS by default.
- Optional CDROM drive.
- ARM 7500FE processor. (32 MHz)
- 12 MHz RAM.
- 4096k OS in ROM.
- ~28.7 MIPS.

**Stork** (Acorn Risc Technologies) the successor to the A4 this is effectively a RiscPC in portable form. There are a few improvements over the capabilities of the RiscPC range including a PCMCIA interface. It is unique in being the first Acorn portable to offer a docking station arrangement. Not launched initially as a product it has been more a licensable design available for volume customers to order as required. A by product of this is that design is somewhat mutable featuring at least two types of LCD screen chosen as required by the customer.  
 The design has been available since mid 1996.

- 8192Kb Memory (expandable to 256Mb)
- ADFS by default.
- IDE interface. (425 Mb)
- ARM 7500 processor. (32 MHz)
- 12 MHz RAM.
- 4096k OS in ROM.
- ~29 MIPS.

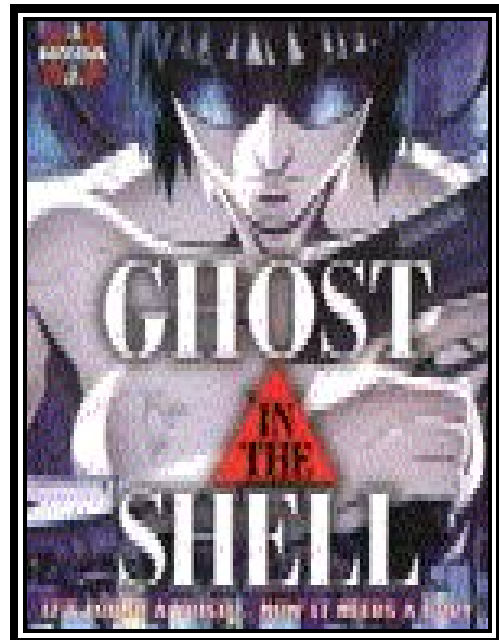
**NewsPAD** (Acorn Risc Technologies), perhaps the most unusual of the portable offerings from ART this is a graphics tablet arrangement of portable with no integral keyboard offered. Rather input is through an inbuilt touchscreen overlaid on the colour LCD display. Two PCMCIA slots are available along

with 496 bytes of non-volatile RAM, compared with the more usual 240 bytes offered. A docking station provides parallel, external monitor, serial, keyboard and mouse ports along with a floppy disc drive.  
 Available since Mid 1996.

- 8192Kb Memory (expandable to 256Mb)
- ADFS by default.
- IDE interface. (420 Mb)
- ARM 7500 processor. (40 MHz)
- 12 MHz RAM.
- 4096k OS in ROM.
- ~35.9 MIPS.

**StrongARM RiscPC** (Acorn Risc Technologies) basically the 700 series machine with larger hard drives, RISC OS 3.7 and a StrongARM (202 MHz) in them. Released September 1996.

By Philip R. Banks (banksie@khantazi.wn.planet.gen.nz)



# THE ACORN CODE

Are YOU serious about your Acorn? If so, proudly display the AcornCode information in your signatures. Then, every other proud Acorn owner can see how serious YOU are about yours.

This code has been designed by Quintin Parker. The latest version of this, and a fuller text, is available at <http://www.werewolf.demon.co.uk/quintin/code.html> so pop by later on.

**v1.01 (14-Jun-1997)**

By Quintin Parker

## Introduction

Ever seen Geek Code? The list of funny symbols often put at the end of sigs or in peoples pages??? If you haven't, you can visit the Geek Code homepage (<http://hrypton.mankato.msus.edu/~kayden/geek.html>), where it is explained much more eloquently than here. The problem is, for us Acorn users, is that it asks you quantify all your opinions of UNIX and PC programs, whereas many of us wouldn't even touch them with a bargepole! You may have seen Brit Code, which is fine, but only contains one single little mention about Acorns and doesn't contain any extra information... This situation has gone unresolved until now ... the Acorn Code, a supplement you can use on its own or with your original Code!

Hurrah! It's thwogging brilliant!

## How These Codes Work

In this code and the Geek Code, you have a list of particular flags, plus a number of +s or -s, or other symbols, which demonstrate your opinion towards various things. Let's take bc meaning Bouncy Castles, as a convenient example. Somebody who rated themselves bc++ would be a bouncy castle enthusiast, but not nearly as enthusiastic as a bc+++. Similarly, somebody who was bc--- would detest the things. No flag - just bc on its own - would mean an impartial/apathetic view. There are also other things you can add to qualify these flags...

- ! When this placed before the flag it means `not', therefore indicating a total refusal to participate in the particular activity or holding no opinion whatsoever. eg. !bc I've never gone on a bouncy castle in my life!
- > This means that the user is one category but is aspiring to be another. eg. bc+++>-- I have a bouncy castle addiction. I would so much like to stop going on them and see what it's like to dislike them for a change.
- @ This means the opinion is likely to change due to circumstance. eg. bc@ I used to like bouncy castles as a child, but now I can't abide by them. Sometimes I go on them when I'm plastered, though.
- (...) parentheses You may place these after your main opinion, and contain secondary opinions that

you may also have. These are different to @ because @ means that the opinion you hold is likely to change to anything, whereas (...) is specific. eg. bc(+++) Normally I don't really mind about bouncy castles. But I absolutely adore bouncy castles that are bright pink. eg. bc--(---) I hate bouncy castles. Sometimes, though, I absolutely detest them.

- ? Implies a lack of knowledge or a lack of information about the particular subject. Don't feel ashamed about using it - sometimes it can actually be a bonus not to be acquainted with a certain thing :) Placed after a flag. eg. bc? Bouncy castle? What good are they when faced with a marauding Angevinbesieging army?
- \* This character is often called a `splat`, and means just that : a sort of all over the place alternative sort of view. It's different to @ in that @ is in the boundaries of the code that is defined, whereas \* means lots of other things. May be used in conjunction with other flags. eg. bc+++\* Not only do I like bouncy castles, I like bouncy sharks, bouncy life-belts, bouncy shopping malls, bouncy giant chess boards, bouncy... eg. bc---\* I utterly despise not only bouncy castles but also all things bouncy. I carry a sharp pin with me at all times so I can stop infernal bouncy demons annoying me.

You will find full descriptions of what exactly each inflection on each flag means - choose the one which fits your opinion most. There are also flags you can use that describe what computers you own, and so on. These will be dealt with in the next two sections, where the Acorn Code is described properly...

### How to Begin Your Acorn Code

First of all, you must declare that it is Acorn Code you are writing. This you do by starting with;

```
AcornCodev1.01:
```

The version number is important to put in, as inevitably extra things will be added to the code. You can now go on, if you like, to describing your setup. None of the flags are completely compulsory, and you can leave some out if you want, but it's nice to have the full picture. By way of example, here is my (Quintin's) own Acorn Code. Don't feel daunted! It's very easy once you get the hang of it...

```
AcornCodev1.01: A2 rpc7sa/I1/10/Icd12 a441/I350/4/ro+/sa BB*
                ed++ dc200 nl+ EC pc-- pr+
```

### Describing Your Computers

After you've declared you are writing Acorn Code, it is important to say just how many Acorns you own, using the A flag..

- Ax        Instead of x, put in the number of Acorns/Arcs you own, excluding BBCs (but including A3000s). BBCs are dealt with later..
- !A        Never used an Acorn! (In that case, why are you using acorn code, you pillock?)
- A?        I lost count...

Now describe your computers. The one you use most you should put in, but you can put another one if you like. Just don't put all twenty in...

- rpcsam Risc PC, with StrongARM on the motherboard. (not avail. yet :))
- rpc7sa Risc PC 700, with StrongARM card.
- rpc6sa Risc PC 600, with StrongARM card.
- If you've cut the tracks and changed the switches to increase the speed on any of these three, tack on a \*.
- rpc7 Risc PC 700
- rpc6 Risc PC 600
- a7000 A7000
- a5000 A5000
- a4000 A4000...
- not too difficult to work out the rest, is it?
- a3010 A3010.
- Wow. That was hard.
- a3000 A haddock. No, only joking. Though a haddock would probably be faster.
- a540 A540
- a441 A440/1
- a305 A305
- a310 A310
- a440 A440

If I missed any out, mail me first for clearance.

To show how much HD space in total you have, put an oblique afterwards and then the number of megabytes or gigabytes. (It's pretty obvious which is which... but if you want to make absolutely sure, add a g or m afterwards.) If there's no HD at all, put a 0.

Prefix the number with the interface type.

- P Parallel port type thing
- I IDE
- S SCSI
- T ST506

If there is more than one interface, divide it up.

Examples:

- rpc7sa\*/I512 Risc PC 700, doctored StrongARM, with 512Mb IDE HD.
- a5000/S2.2/I100 A5000 with 2.2Gb SCSI HD and 100Mb IDE.
- a3010/T20m An A3010, with 20Mb ST506 HD.
- a3000/0 A3000, no HD.

If you've upgraded to an ARM3 or another processor (excluding StrongARM) then add another oblique and put in:

- A7 ARM 700
- A3 ARM 3

Finally, if it's not immediately obvious, add another oblique and put in the total memory (including VRAM) you have.

A StrongARM Risc PC with 512mb HD 5Mb memory, would be `rpcsa/512/5`.

An A305 with 40mb St506 HD and ARM 3, as it's obvious it's only got half a meg memory, you can just leave it as `a305/T40/A3`.

You should also put what version of Risc OS you use on it, if that isn't obvious.

```

ro++++ Risc OS 3.70
ro+++  Risc OS 3.60
ro++   Risc OS 3.50
ro+    Risc OS 3.11
ro     Risc OS 3.10
ro-    Risc OS 3.00
ro--   Risc OS 2.01, Risc OS 2.00
ro---  Risc OS? Pah! Arthur is fine for my needs.
ro*    I wrote my own version...
!ro    I don't have any Risc OS chips in my computer. I program using the LED switches on my
        Altair 8800.
    
```

So :

```

a5000/I720/4      A5000 4Mb memory 720Mb total IDE HD space.
a441/I350/A3/4ro+ A440/1 4Mb, ARM3, IDE 350Mb HD space and RO3.11
rpc6/S1/5        Risc PC 600, 5Mb,1Gb SCSI HD space.
    
```

HD space is BIGGER, so it comes FIRST!

Any extra expansions should be added on with obliques separating them.

```

tv      TV tuner/teletext card
mi      MIDI
cc      Colour card
(S|I)cdx SCSI/IDE CD ROM (x=speed). If you have more than one, add them all separately.
sp      Serial card.(Add sps as appropriate to the number of cards you have.)
sa      Sound sampler (if separate to a MIDI card)
dg      Digitiser (if separate to TV tuner card)
    
```

If I missed anything out (and I probably have), mail me first for clearance.

## Opinions and Character

An Acorn user is not just defined by his/her computer setup. Many hold quite strong and conflicting opinions as well... These flags are designed to show these opinions so the user can stand up and feel proud s/he is declaring them to the public...

### BBCs, Electrons, and other relics

See, told you I would deal with them :) Many Acorn users have stayed with Acorn right from the early days, and have supported them, like a beloved football team. Often you will find many early Acorn computers in an Acorn users house, because the user is too attached to them to throw them away. Let us know how many of these old faithfuls you own, with the BB flag.

BBx	Number of BBCs, Masters, Electrons, Atoms, etc.. working or not!
!BB	I haven't got any!
BB?	BBC? I prefer ITV.
BB*	I haven't got any BBCs... but I do have a Spectrum or two. :-)

### The Editor Wars

One of the most contentious and controversial issues in the Acorn world is which editor is the best - Zap? StrongED? Edit? Arguments, usenet flamewars, bar brawls, nuclear wars have started over this one small difference of opinion. I'm saying nothing. What do you say? Let people know and use ed to express your view.

ed+++	Edit's too simple! StrongED's for pussy cats. I use Zap because it's got more features.
ed++	Edit sucks. I use StrongED. Zap's way too obscure.
ed+	I use Edit once or twice, like when it doesn't matter when things stuff up. The very fact people have written extra editors must mean that Edit is lacking something.
ed	Edit I use for some things, the more sophisticated ones for others - I don't really have a preference, they've all got different purposes.
ed-	Edit's usually loaded. It's much easier for my fuddled brain.
ed--	StrongED? Zap?? Bleurgh! They're not even on my hard drive.
ed---	I trust Edit implicitly. In fact, I'm writing my novel in it, all 700K of it, it's perfectly fast, and it hasn't crashed once.
-ed+	I really don't give a row of buttons about the whole issue..
!ed	Edit? Get lost. *Build is fine for me...
ed?	I haven't got a clue what editor I use... what day is it today?
ed*	I wrote and use my own editor. :-)

### Discs ... Disks ...

The issue of how to spell this simplest of words has divided the Acorn Community right down the middle. Is `c' a disgusting and misleading Americanism? Is `k' a signal to conformity and an ugly spelling? What do you think?

dc spell `disc`  
 dk spell `disk`  
 d@ I spell it either way.  
 d\* That's boring. I normally spell it `teapot`.  
 !d I don't have any discs. I type everything out from memory.

If you like, you can put afterwards approx. the number of discs you own (dc200 means 200 discs, spelt with a c.)

## NewerLook

Doggysoft's desktop enhancing program has caused more ructions in the Acorn world over one single principle ever since, er, the last ructions in the Acorn world over one single principle. Is it a life-saver, and are you unable to function without it? Or is it unnecessary makeup on an already perfect visage? Where do you stand?

nl++++ Blib! I was one of the people who wrote it! Tee hee!  
 nl+++ I always make sure I have the latest version. I use all of the features and bells and whistles, I even install things that are already implemented, and all the memory it takes up is worth it, believe me.  
 nl++ Yep. I got it installed. It's cool.  
 nl+ Yeah, I've got one or two icons from it, and nicked a few thingies from it, but I don't have it all installed.  
 nl Hmm. I really don't have a preference either way.  
 nl- I don't like NewerLook much. I prefer NewLook, to be honest.  
 nl-- I don't like NewerLook at all. I much prefer the plain vanilla desktop.  
 nl--- NewerLook is an abhorration, an ugly boil on the face of any Acorn unfortunate enough to have it installed, and each Doggysoft member should die a painful death, one by one, slowly, bloodily.  
 nl---- I am Glenn Richards.  
 !nl I've never used it.  
 nl? I've never even heard of it...

## Econet

Networking, using the native Acorn networking hardware Econet, is quite often a past-time of Acorn users, particularly Acorn Geeks. Use the EC flag to tell us whether you take part!

EC++++ I am Glenn Richards.  
 EC+++ I have networked several Acorns, Arcs and BBCs all around my house. I spend a lot of time comparing network times with serial rates. My mission in life is to network every single Acorn in existence. I have written, or I'm using, an alternative network manager replacing Level 4.  
 EC++ I'm the syst, and have more than two computers networked. Running the network takes a lot of my time.  
 EC+ I'm the syst on an Econet system... don't spend a lot of time with it though.  
 EC Yep. I have an account on an Econet system. Somewhere. What was my password again?



- EC- I've never used Econet. And what on earth is a syst? Isn't that some horrible kind of blister?
- EC-- I've used Econet, and not liked it ... There are much more fun things to do with your Acorn. Like designing knitting patterns on it.
- EC--- I flamethrower every Econet card I see. Econet spreads diseases!
- !EC I've never used any kind of network at all.
- EC\* I network/used a network, but not with Econet. (In that case, add +s or -s as appropriate.)
- EC? Econet? That's like Greenpeace, isn't it?

## PCs

Acorn Users have traditionally hated PCs. What about you? How much dislike do you hold for Gates and his Imperial Windows? Do you even like it? Tell us!

- pc+++ I wish my Acorn was more like a PC! I think Windows 95 is such a lovely, smooth, easy to use GUI and I can't stand the way in which the RISC OS desktop doesn't shield you from the actual files on the hard drive.
- pc++ I quite like PCs - I use them at work/school/have another one, and think they're quite fun - they are the industry standard after all. But I still use my Acorn.
- pc+ Yeah, PCs are fine. What's everybody else's problem with them? It's just My Computer Is Better Than Your Computer all over again.
- pc I don't really care. I've got an Acorn, I've used a PC, and I don't see the difference in quality, quite frankly.
- pc- They've got little niggles in, and aren't particularly intuitive. I would use one, if forced, but I prefer my Acorn.
- pc-- I don't like them at all. The software's too expensive, they crash perpetually, it's all too corporate and Bill Gates is a nerd with enough cash to fund a small country's defence budget. That's why I bought an Acorn, that's why I'm sticking, and I often get into fights with PC users.
- pc--- PCs are scum. Anybody who buys them has got to have something wrong with them : in fact, I'm sure I read somewhere that the mark of the Antichrist, 666, actually means `Windows'! It makes sense - nobody can buy or sell without it etc etc, and it turns people into slaving minions of the night. Bill Gates is the Devil Incarnate. Acorns are the saving Light of the world.
- pc---- I am Glenn Richards.

## Programming

An integral part of being a geek, and indeed an Acorn user, is how good you are at telling your computer what to do? Are you king of the keyboard or programming peasant?

- pr++++ I am Richard Sergeant.
- pr+++ I challenge you to find a programming language I don't know. I use PASCAL regularly, and either have written or in the process of writing my own language compiler.
- pr++ I can do ARM assembler, but do C and C++ instead. I only use BASIC for stupid things.
- pr+ Hmm. I can do C or ARM code, with BASIC, but that's about it.
- pr BASIC is cool.
- pr- I'm just learning it. Um, what does ENVELOPE do?

pr-- All my knowledge of programming is typing in programs in electrical stores that print  
`Dixons is Crap`.  
pr--- Not a clue. Sorry.  
pr---- I work for Generation Design.  
!pr Programs? My God! I thought computers just used instinct!  
-pr I'm a dab hand at Commodore 64 BASIC.

### **Publishing your Acorn Code**

You can do this in a number of ways - put it at the end of your internet sig (it's not very big at the moment, so you probably won't get people upset), or in conjunction with your Geek Code or Brit Code if people finger you (don't for heavens sake add all this to your sig!)...

For all of those parts that Acorn Code doesn't reach (ie. the non Acorn bits), you could do a hell of a lot worse than add in Brit Code, seeing as if you're an Acorn user you're likely to be living in Britain ;). It contains much more information, such as the H flag for hair, and so on.. Do what you like. As long as you use Acorn Code!

### **Contacting the Author**

This is the first version of the Acorn code, so if you have any new ideas or comments please don't hesitate to mail me at *quintin@digibank.demon.co.uk*. I'm particularly looking for new flags to add ! Let me know, and don't go mad!

Thanks to everybody who's e-mailed me giving support and extra ideas. I can't list you now because my mail program crashed and I lost your names - sorreee! :(

### **Credits**

Original idea : Robert Hayden  
Text : Quintin Parker  
Graphics : Background by Brain Soup, graphics by Quintin Parker (using the Artworks Acorn)

---

THE END

Directed by R.Murray  
Visual effects by B.Hartshorn  
Camera by P.R.Banks  
Special Effects by Q.Parker

No animals were harmed in the making of this production.

Copyright © 1997 Richard Murray  
A Hissing Spinach Presentation