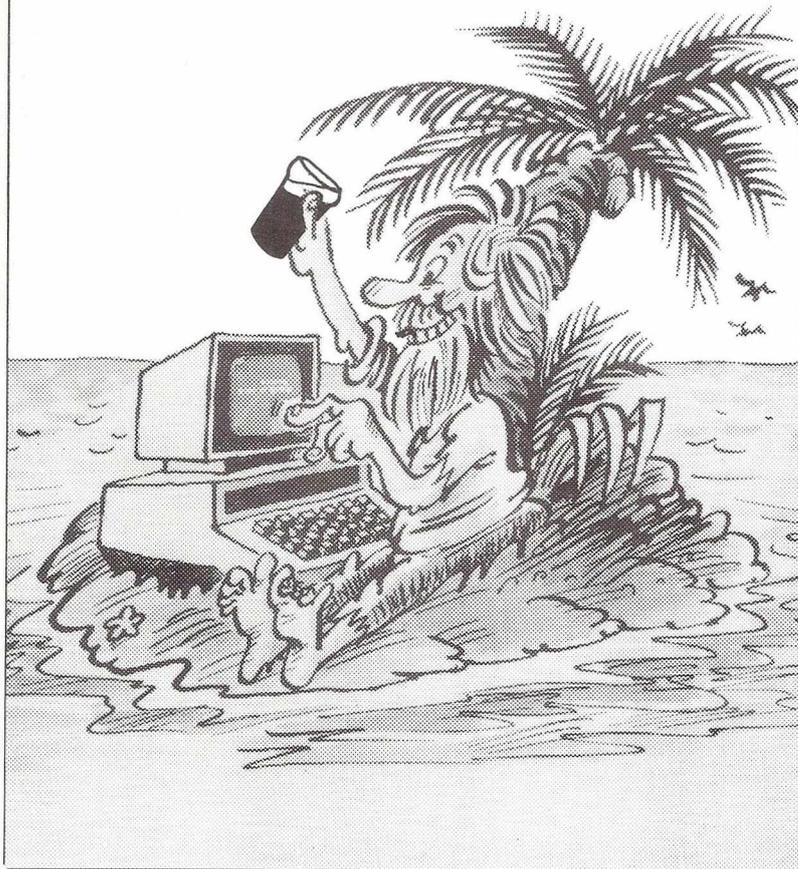


EDUCATION
EVALUATION OF A
SUPERPET

COMMODORE CLUB NEWS

January 1982

you're never
alone
with a....



Volume 4 Issue 1

DAY TRIP TO I.M.I.

Over 60 PETs in action !!

THE MANAGER

New software product
evaluated

DISK USE FOR BEGINNERS

Back by popular demand!

PERIPHERAL SPOT

With the accent on
Communications

EDUCATION

Special Centre section
"pullout"

 **commodore**
COMPUTER

“Every PET owner should read it”

Chuck Peddle, Inventor of the PET

“The PET Companion” is a fascinating collection of essential PET information from the pages of *Microcomputer Printout*. It contains **all** of the editorial from the 1979 & 1980 issues, including 105 PET programming hints and tips, 116 news reports, reviews of 54 peripherals ranging from light pens to printers and 27 major articles on PET programming. All of it written in straightforward English.

Some of the topics covered:

PROGRAMMING THE PET

Double Density Plotting
Modular Programming
Programming Style
Graphics
Subroutines
Sorting Out Sorts
Tokens
The Game of LIFE
Tommy's Tips
ROM Addresses

THE SOFTWARE

Business Software Survey
Cosmic Invaders
Superchip
PETAID Do-It-Yourself Database
What's Wrong with WORDPRO?
Screen Display Aids
Keyboard Tutor
Photography Course
Who Do You Want To Be?: Fantasy Games
Commodore's Assembler Development System
Programming Aids & Utilities Survey
PET Games

HARDWARE REPORTS

The New ROM Set
CBM 8032 SuperPET
CompuThink Disk Drives
Hardware Repeat Key
High Resolution Graphics
The Commodore Printer
How the Keyboard Works
AIM161 A to D Converter
Commodore's 3040 Disk Drive
PET's Video Logic
Colour for PET: The Chromadaptor

THE SPECIAL REPORTS

PET in Education
PET Show Report
The Jim Butterfield Seminar
Hanover Fair Report
PET In Public Relations
Local User Groups
High Resolution Graphics
Commodore's New Technology
Future Shock: Forecasting The Future
Speech Synthesis
PET As Secret Agent
A Visit to the Commodore

plus news, letters, gossip and regular columns by leading PET experts.



To: Printout Publications, P.O. Box 48, Newbury RG16 0BD. Tel: 0635-201131

Please rush me a copy of the PET Companion

I enclose cheque/postal order/money order

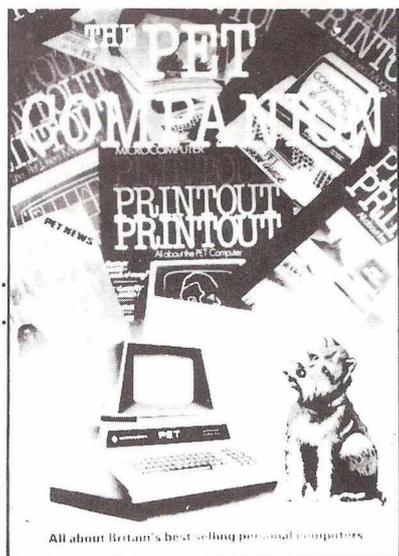
Please charge my Access/Eurocard/Mastercharge
or Barclaycard/Visa No:.....

UK £9.50 plus 45p postage USA \$25 Overseas £12
(Credit card orders accepted by telephone on 0635-201131)

NAME:

ADDRESS:

Postcode:



All about Britain's best-selling personal computers.

Contents

Editorial — Hello 1982.....	2
Day Trip to I.M.I. — On the road again.....	3/4
Second Hand Equipment — More good buys.....	5
The Manager — Close encounters of the programming kind.....	6/8
Basic Programming — More hints and tips.....	9/12
Machine Code Programming — Help and advice.....	13/16
Supermon 4 — Finding the Mean.....	17/18
Disk use for Beginners — Back by popular demand!.....	19/22
Peripheral Spot — Communications this time.....	23/26

Educational Supplement

Editorial — Back again.....	1
Open University — Horses for courses.....	2/3
1991 — A look to the future.....	3
Independent User Groups — Listings.....	3
Educational Workshops — A complete up-to-the-minute list.....	5/12

MICROCOMPUTER EXPERIENCE?

Comac Systems is a specialist Commodore Dealership with a national reputation for its Maintenance Management Systems. Increasing demand has created the following opportunities —

SENIOR SYSTEMS PROGRAMMER

If you have substantial computing experience and the ability to develop a complex package from initial design through to error free coding, then this is the job for you. You must be able to program in both BASIC and Assembler and should be thoroughly familiar with CBM hardware. A background of commercial data processing is desirable.

A substantial salary together with a computer car will be offered to the right applicant.

PROGRAMMER

This vacancy is offered to someone with limited experience - perhaps a hobbyist looking for an opportunity to break into the business. You must be a competent BASIC programmer and good at Maths and English.

Salary is negotiable and depends upon qualifications and experience.

Interested? - then post your C.V. to —

Peter Antill
Comac Systems
Spaceregal House
Coln Estate
Old Bath Road
Colnbrook
Bucks
SL3 0NJ

FREE PET/CBM COMAL

“The excitement in Europe (over COMAL) seems to be growing by the hour and we look forward to America being able to share in the good fortune of having an easy-to-use, structured, planning language at last.”

The power of PASCAL and the ease of BASIC can now be yours with Commodore COMAL, a new programming language from DENMARK. It is being distributed in the USA by the COMAL USERS GROUP. To find out more about COMAL and how you can get a free disk copy of Commodore COMAL, send a large self-addressed stamped (35 cents) envelope to:

COMAL USERS GROUP
5501 GROVELAND TER., MADISON, WI 53716.
Outside USA please add \$2.00 for airmail and handling.
*PET & CBM are trademarks of Commodore Business Machines.

Editorial

1982 ?! Whatever happened to 1981 ? Come to that, whatever happened to New Year's Day ? I remember the day before very well, but for some reason my knowledge of the first day of this new year is a little hazy. But then, why change the habits of a lifetime?

So now that you too have written your first ten cheques with 1981 on instead of 1982, and tried to alter it but ending up only with a distinct 'mess' in the top right hand corner of the cheque, it's time to return to normal.

Well, not quite. Commodore Club News is here, 1982 version. The magazine comes out much the same as before : I know one or two series of articles managed to 'lose' themselves at the tail end of last year, but consistency is the name of the game this time around. So, What have we for you this time?

1) The editorial. This, as well as containing the usual introduction to the magazine, will also have little snippets (SnipPETs ?!) of information. News, views, newspaper items, press releases, second-hand equipment, and so on. A number of pages covering a broad spectrum of Commodore related information.

2) General Articles. We've featured a number of these in the past, including such articles as Commodore Splits the Proton, VIC meets the Stars, and so on. This month we give you the story of a journey I made to I.M.I. in Witton, near Birmingham, a company with over 50 PETs at the time I wrote the article (getting nearer 60 now I believe). There is also an article on 'The New Building', in which we are ensconced at this precise moment in time.

3) Reviews. Commodore's relatively new software product 'The Manager' gets our unique treatment this time.

4) Basic Programming. A selection of hints and tips, programs and letters, all designed to help you get the most out of your Commodore system. Being in Basic, these programs will normally quite happily transfer from PET to CBM to VIC etc., and will hopefully all prove of interest. If you've developed any programs of

your own that you'd like considered, feel free to contact the editor at the address given at the end of every editorial. The programs cover many areas, and are not always just programs : for instance, this month we have, by kind permission of Henry Budgett and Computing Today, an item on 'standards' to be used when printing program listings e.g. the normal cursor down symbol appears as cd etc. Thanks to Henry for an interesting article, and hopefully this will lead to more people conforming to these standards.

5) Machine Code Programming. Again a selection of programs, along with a number of articles on the art of machine code programming. To give an example, this month there is a further note to our earlier publication of Supermon 4, the classic extension to the on-board PET monitor, detailing how to get the best out of the monitor.

6) Disk Use for Beginners. Due to circumstances beyond my control (Christmas!) we didn't see this for two months at the tail end of last year. Please accept my apologies for that, and an assurance that David J. Pocock is back with us once again!

7) Writing your own Compiler. Again a series that disappeared at the end of last year, but which will be a regular contributor this year. Essentially a guide to writing, as opposed to buying, a compiler for your PET system. For those of you who are new to the computing world, a compiler is a program that converts source code (that Basic program you've sat down and typed in) to object code. If that doesn't make sense, look at it this way : your program will run significantly faster after it has been compiled!

8) The Peripheral Spot. A number of programs devoted to a particular peripheral, be it a disk drive, printer etc. Since most of the programs in existence require a disk drive, it is this area that tends to be concentrated on, and thus most of the programs we've published relate to that particular peripheral. There is, after all, significantly more that you can do with a disk drive than you can with a tape deck, for instance. Agreed you

can turn both of them upside down and they'll carry on working, but (as they say) you get my drift.

And that, apart from the central, is the format for January 1982. The central section started off life in a simple way: we used to produce a magazine called 'Microcomputers in Schools and Colleges'. I say used to, we published one in fact. Not the high spot of the Commodore publishing world, that little number. In order to bring 'MISAC' back to life, it was decided to have a special central section in Commodore Club News devoted every other month to the role of Commodore in the world of education. This led to the problem of what to do every alternate month. This was nicely solved in the last six months by three superb articles from Dr. Phillip Barker, of the Teeside Polytechnic, on communications. This month we're on education again, with, amongst other things, an evaluation article of the new SuperPET.

So that's it. I hope you have a successful 1982, and that you enjoy this year with Commodore Club News.

If you've any news, views, letters, material you'd like considered for publication, or indeed anything for that matter, you can contact me at the address below:-

The Editor
P.O Box 13,
Yeovil,
Somerset.
Note new address!

And now a footnote. Your friendly editor is moving on. But don't panic (or cheer for that matter!). I will be going to join Nick Hampshire Publications (you know the name, The PET Revealed, that kind of thing), and we will continue to provide you with a magazine on the entire Commodore range of equipment, covering all spectrums of interest. If you already subscribe, you're on our list, and the magazine will simply pop through the door. Known as **Commodore Computing**, we hope you enjoy it. First one coming out soon!

A Day Trip to IMI

"Get me more applications stories!" my boss screamed. "You should get out and about more, get me more stories!"

Fair enough, I thought, days out of the office, the big expense claim at the end of it, the whole works, why not? So, I was on the lookout.

Coincidentally, not long after this, our information centre's 'phone rang and the conversation duly revealed that the enquirer was a Mr. Kingscott of IMI Ltd., and that they had over 45 PETs! 45 PETs in a company like IMI just had to have a story attached to it, and I subsequently found myself journeying North to Witton, not a million miles from Birmingham.

The journey, as ever, was not without difficulties. Taplow Station, whilst having a charming person serving behind the counter, is not the quickest acting station in the world. Consequently, when you see your train arriving and there's still three people in front of you, there's only one course open for you: run! I ran, and jumped on one of the few trains to Reading. It was at Reading (and subsequently at Birmingham) that I discovered one of the reasons why British Rail is losing money.

At Reading, without having to hand in my non-existent ticket, I boarded the train to Birmingham without purchasing another one: I didn't even have to jump past any guards, there just weren't any there. On the train there were no ticket inspectors, so I arrived at Birmingham still without having bought a ticket. There my journey ended, but if I'd wanted to I could have happily boarded another train, once more without buying a ticket. Being an honest chap, when I was leaving Birmingham Station I stated precisely where I'd come from; however, I could have gone on to Glasgow (or somewhere like that!) and said I'd got on at the previous stop. British Rail, start investigating!

How to find IMI

Once in Birmingham I wasn't quite sure where it was. The first word of warning here: if you're ever in the Birmingham city centre, and you're near a pub called The Old Contempibles, don't go in it! I went in there, and went under the theory that if you buy a pint and then ask where somewhere is, they'll be more likely to tell you.

So I uttered the immortal words "Pint of Guinness please", and got this Niagara Fall of Brummie coming at me. Assuming he'd said something like 'the Guinness is off today', I asked for a pint of bitter, and got another Niagara Falls. In despair, I asked him to speak slowly, and in English. The result was ...

"We don't serve people wearing training shoes, sir". Stunned silence, and I walked sadly out.

Rescue was at hand however, in terms of The Royal Oak (I think), not too far away. I still couldn't believe the training shoes bit, and asked in the Oak about it, and the reply was "I know, I know, we get 'em in here all the time. I can't believe it". So there you go: a point (pint ?!) to bear in mind.

I found out where IMI was, the point of the exercise, and set off. When I got there, I found an immense number of buildings, and a policeman politely requesting what was I doing there. Having told him, I was then shown to IMI HQ and Reception, where I went through the procedure again, and got handed a 'Safety and Security' card full of rules and regulations, all for my own good I might add. Well, with one exception: rule 2. 'Smoking is not permitted on the Factory roadways. In buildings and offices permission must be obtained'. Aaargh! Never being one to put people out I didn't particularly feel like asking permission, and spent a harrowing time not smoking. Still, I survived.

Use of PETs in IMI

My contact was a Mr. Tom Kingscott, in the Central Office Block, and it was at this point I discovered how harrowing it was not to be allowed to smoke on the roadways. It was like walking to the wicket from the pavilion, last man in, 50 runs to get and their fastest bowler is on. It seemed MILES!

I met Mr. Kingscott, who introduced me to Roy Pagett, manager of IMI Operational Research Department, and was shown into a room with a welcoming sight, a PET sitting happily in the corner. Next door was the actual office, and after the cup of coffee (charming secretary) we found out a little bit about the history of IMI and their involvement with computers.

They have, for a long time, been us-

ing IBM mainframes, and over 40 Datapoint minis have been installed over ten years. It is IMI policy to use these suppliers in their respective fields, but the time came to review micros and the possible extension of computing power provided by the entry-point price levels. Their first contact was with Tandy, but after 3 fairly unsuccessful months went onto investigate PETs. PETs were chosen originally because of the 'all-on-one' configuration, and ease of use.

The first arrival

The first PET arrived on the 5th May 1979, and they quickly came to realise one of the major advantages of using a micro: it's much cheaper to find out where you're wrong on a micro, and it's very easy to develop programs quickly and effectively.

The next move was from the original cassette based system to a disk based one, and 12 months ago onto 8050's to take advantage of the greater range of commercial software available.

One thing they noticed happening time and time again. IMI has two large sites at Witton and Leeds, as well as many others around the U.K. What was happening was this: one group would get a PET system, start actually using it, and then others on their site would want to acquire one. A popular computer!

At this point, a word of praise for IMI's local agents, namely Taylor Wilson. IMI now have 60 PETs, and assorted cassette decks, disk drives, printers etc., and never a single complaint against Taylor Wilson. Well done!

As IMI acquired all these PETs, it was natural that the O.R. Department became a central source of information, and whenever a query arose from another department or branch of IMI they could come back to Witton. Consequently a fair degree of PET 'expertise' is centered here, and this led to the next move: in-house training courses. After several of these the only way they could go was outwards, and now they've started opening their training courses to outsiders.

Training Courses

These courses cost 150 pounds, last for two and a half days, and are based at the Witton site. They promise to teach you PET BASIC programming in that time, and the contents extend

to such subjects as keyboard controls, special functions and so on, as well as numerous 'hands-on' exercises. 1982 starting dates for these courses are 2nd February, 6th April, 5th May and 22nd November. Alternatively, if you've got 6 to 8 people to be trained, a course may be run at your premises. Ring John Hind for further details, on 021 356 4848 extension 2215.

So what are all these PETs being used for?

The PETs are used for a wide variety of applications, ranging from technical calculations to traditional commercial data processing, in situations where computers were previously not justified. Programs and systems range from small to large, from simple to complex. Many are written by the users themselves, but some of the more involved have been developed by the O.R. department.

The first job they tackled was in control of a jobbing toolroom, where they used the PET to keep track of each particular order. Planning the workload of the toolroom had previously involved experienced seminar people, but doing the job on the PET was found to be so much easier, and they could also instantly find out where every job was, when it was out, and what the forward loading of the various machines would be.

Interesting Applications

At the Witton site there's an interesting application involving a PET, with IMI Titanium. There a PET is permanently 'on-line' as customers call, to deal with sheet, billet and other types of castings in the sales department. The program used leads the sales person through a very comprehensive series of questions and answers, rapidly getting the information required.

Some of the jobs they are now looking at involve the monitoring and controlling of equipment. This got under way when they were asked to link a PET to a weighbridge, recording (obviously) the weights of items passing over it, and this will provide management with an awful lot of information on movements on the site. Basically they will know where everything is at any one time, because the PET remains permanently on.

Another area IMI have been investigating is the linking of PETs to mainframes, and in particular 8032s

to IBM machines. This is being done via a package from one of our communications specialists, Cortex, in Bedford. To speed up the communications links, a fibre optic system is being looked at by Cortex at the moment, and if and when that gets off the ground I'll bring you the story.

IMI are examining the feasibility of using VICs industrially, with a specific project in mind at present, namely the measuring of the thickness of copper strip, and printing out the results in graphical form. Another triumph for Commodore machines!

Other Software Packages

As well as all their own software, they are using a number of 'off the shelf' programs. The program most commonly used is, perhaps not surprisingly, Visicalc. Several of the IMI sites are using this program, and at Witton the program is in regular use in the company planning Department. Another dedicated machine is for Word Processing, using the Wordcraft package: very favourable reports on both programs. With the news that their 8032s could be upgraded to become 8096s, and that with Visicalc in particular you'll then have 69K of usable RAM, a significant advantage, it looks like IMI will be staying with Commodore for a long while to come.

Finally, I'd like to express my thanks to all the people I met at IMI, and the hospitality I was shown there.

New Premises

During the Summer and Autumn of last year Commodore decided to tread boldly where no company had gone before, and took over new premises. The need for new premises had been apparent for quite a while, as the company was getting larger and larger. New departments were springing up daily (or so it appeared), and it was obvious that we had to have somewhere to put all these people. Similarly, as the range of equipment we supplied grew, the need arose to have more warehouse space in which to put everything.

After reviewing the situation with their heads two inches from the ceiling atop a pile of PET boxes crammed into the warehouse the management decision was taken: we must have new premises!

The search was on. Personally I

had my eyes on the Slough Estates Building opposite our old premises: a strange looking building, somewhat like an upside down pyramid, but bearing an incredibly modern appearance. It would go with the image that I always try and project, I thought. However, it was not to be, and we ended up moving to 675 Ajax Avenue.

I would like at this point to dispel an image that some people have of the Slough Trading Estate. A vision of incredibly long streets, which makes it very difficult to find the place you're looking for. This is not the case: we are simply on plot 675 on the great map of the estate, and plot 675 happens to be in Ajax Avenue.

The Image

So what is Ajax Avenue like? Not the cultural high spot of Slough, but then Slough, as Sir John Betjeman so wisely observed, isn't exactly the centre of the Cosmos either. But one must not complain: one must take one's work site as one finds it. The 125's hurtling past the window every now and again create a bit of a nuisance, as does the obligatory ambulance zooming past at full speed, sirens blaring etc. Always seems to go past at the same time for some reason. Usually about ten minutes after the pubs have closed: is there a connection?!

That is but one of the saving graces of the new building: the pub is nearer! Not that we editors drink at lunchtime, you understand, but after a hard day's work, with 5.30 looming up, it's nice to know you can make it in time.

The building is in fact quite modern in appearance (just four years old they tell me), both inside and outside, and we now have a pleasant reception area as opposed to the older, rather off putting appearance of the other building. Very pleasant reception staff as well! A large car park is the other main point that hits you as you enter the premises, and if you're walking in when a certain accountant is driving in you could very well have something else hit you.

The one off-putting sight is the Colditz-style barbed wire fence around the place, that looks like it was designed to keep people in, rather than out. I understand that the Technical Services tunnel is progressing well, under the leadership of Rod "Biggles" Wellburn. When the first

successful escape is made, I'll let you know.

Innovation

Once inside the building you're standing very close to an innovation in this area, namely the telephone system. Apparently, we're the first company on the Estate to install such a system. Going under the glorious name of Monarch 120A, the heart of this system is the switchboard itself. This is really well worth a look at : it looks more complicated to use than it is to program in machine code, but they tell me it's very simple once in actual operation. All touch sensitive keys (of course), and it comes complete with a screen giving you little messages of information whenever anything happens, complete with the obligatory bleep.

Consequently we now have a superb telephone network. Even our own individual 'phones are not without their wonders. Every button you press has it's own little tone, so consequently ringing up various numbers can be quite musical : ringing up my printer sounds like 'Yankee Doodle'! Every newcomer to the office invariably spends some time playing with the 'phones, playing Beethoven's Seventh, the usual things, and this is made a lot simpler by the rather extensive manual that accompanies your 'phone. Transferring calls, three party conversations, automatic re-dialling of the last number dialled, diverting all calls to another 'phone, and more, are all remarkably easy.

As far as the rest of the building goes, the key-word is LARGE. There is an awful lot of warehouse space, which should be more than enough to cope with our future requirements for the next few years, and unlike the old days of Euston Road, there is even enough office space now. The office I'm in has got seven people in it : in Euston Road days that same amount of space would have had about 15 people in!

Office Plan

The place is essentially open plan, apart from the top brass of course, who each have their own office. The rest of us revel in conditions that wouldn't have been expected in the old days : we've got room to move around in. Admittedly my desk space is slightly less than it should be, with telephone, cigarettes, ash tray and

cups of coffee elbowing each other for room. Piles of paper take up much of the rest of the desk top, as prospective newsletters emerge from the old Wordpro.

To sum up what this new building actually looks like, the pictures here

should help. Taken on a summer morn, with myself and my printer happily wandering around the building annoying everyone by taking surprise pictures, I've been threatened with all kinds of things if these get published : oh well!

Do you want to advertise Second-Hand equipment?

Name

Address

.....

.....

Equipment

.....

Please fill out the form above if you wish to advertise second hand equipment for sale in Commodore Club News. Entry costs just £1.00 (cheque or postal order), and return it to: Peter Gerrard, Commodore Computing, PO Box 13 Yeovil Somerset

Commodore Computing do not accept any responsibility for the products advertised hereunder and prospective purchasers should satisfy themselves in respect of any representation made.

Second Hand PETs

8032 + 8050 + Eprom MX80 printer + cassette + cables with manuals. Mint condition - hardly used. Cost over £2500 will accept £2000 or near offer for the lot.

Name and address to contact :-

Chirs Lawless,
17 Aston View,
Woodhall Farm,
Hemel Hempstead
Tel no: Home: (0442) 40953
Work: 01-903 1333 Ext 359

CBM 3032 with Toolkit and manuals Teletype 43KSR All necessary leads etc. PETAID file-handling software Price £1500.00 or near offer

Name and address to contact :-

John Lea,
1 Tudor Avenue,
Hampton,
Middx TW12 2ND
Tel. 01-735 5756 - 01 979 6627

PEt 3032 + Toolkit, Superchip, Arrow, Faster Basic Rom's + Rom X + Tensai Cassette Deck/Sound Box. £650

CBM 3032 and Computhink 800K disc unit. Price £850.

Name and address to contact :-

E. Kilty
3, Town Street,
Marple Bridge,
Stockport SK6 SAA
Tel. 061 449 8101 (Day)
061 427 4084 (Eve)

Commodore Pet 3032 computer with a basic programmers toolkit and a Tensai cassette recorder (all about six months old). Price between £450 and £500 pounds.

Name and address to contact :-

Mrs A M Witcher,
15 Hunters Way
Bishopstoke,
Eastleigh,
Hants
S05 6NN.
Tel. Southampton 695388 (Eve)

One small-keyboard PET for sale. It has been upgraded to 32K from 8K.

Name and Address to contact :-

Jeremy Jacob,
30 New Bond Street,
London W1Y 9HD
Tel (01) 499 0963
Telex 28735

Through the post to the Commodore Training Office, a hidden location deep in the North West, came our review copy of "The Manager". Mike, our boss, prodded about with it over one week-end, mumbled "Velly inteltesting" on the Monday morning and promptly appointed me Training Department Spokesman on "The Manager"!

To be fair he did give me a ten minute introduction to the package. "It manages information on the disk much in the same way you would on paper in filing cabinets.", he said. "You create a format file which acts like a preprinted form for the records you want to store. Then you enter your information using the format file as a prompt, much as you would with any other form. You can add another file which dictates what arithmetic will be performed on the items in each record - you might multiply item 1 on the form by item 2 and place the answer in item 3, grand for extending invoices."

"Those are the three fundamental files", he finished off, "but you can print reports designed as you want them, index files which process the records in a particular sequence and sub-files which are selections of records from the main file - You might create an estate agent's main file with details of all the properties on the agent's books and create from it a subfile of all properties within 10 miles of Slough with 2 bathrooms and 4 bedrooms costing less than 50,000 pounds. "That'll be a pretty small file!", I thought, staggering under the weight of the preliminary reference manual he threw at me.

Following the excellent principle of walking before trying to run, I decided to see if the Manager could cope with the season's statistics for the cricket team I belong to. I'd written a BASIC program to get myself familiar with PET BASIC, and thought it would prove an interesting comparison of the two approaches.

For this application I needed a file with one record per member of the team. The Manager actually prints out work sheets for you to design your record layout, so I spent a little time on this. The Manager en-

courages you to follow the design method we teach in the Training Department, "Consider output first, then input then the calculations." The worksheet I produced is shown in illustration 1.

Notice that several items, or fields as they are called in Manager terminology, are intended to receive the results of calculations. I needed to create an arithmetic file for this, but left it for the time being.

The MANAGER is loaded from a power up situation by pressing SHIFT/RUN. The program is divided into modules, only one of which is held in memory at a time. The first module is a menu by which you can select the various options.

My first task was to produce a 'LAYOUT' file. I thought of this as a mould for the information in each record. It contains all the required text laid out as if it were the input document, plus details of the positions and lengths of the various items or fields on the form.

To create this layout I selected option C from the menu. Straight away I was asked for a file name. I found out by the end of the exercise that this name is pretty important, because you use it to name the format file, the main file, the arithmetic file and goodness knows what else. In future I will keep the file name as short as is reasonable to avoid loads of repetitive typing. Perhaps version 2 of the Manager will let you use a single character such as back slash to use last file name typed in.

The Create module gives you the option of creating a new file or revising an existing one. In my case of course it was a new file.

You are then asked how many screen pages your record occupies, 1 or 2. My application as worked out on the worksheet, only needed one.

You are then offered the option of Lower/Upper case or Upper Case/graphics. Since I'm not the world's best typist, I decided to stick to Upper case. Doing this also avoids problems when using the Search facility - The Manager distinguishes between Commodore and COM-MODORE for example.

Having survived this option ques-

tionnaire, I was finally faced with a clear screen, and a rapidly blinking cursor. To create the layout, one uses cursor control keys to position the cursor, and the normal keys to type text prompts and descriptions. To specify the positions and lengths of fields upwards arrows are used in pairs, with a backlash for a single character. It didn't take me long to type in the form straight from the worksheet. When I made an error, it was easy to go over it with the cursor and over-type, delete or insert extra characters.

When all was correct, a press of one button, followed by an "Are you sure?" message from the Manager, started the Format file recording process. You are then asked how many records you want to create. This was easy for me to answer because I knew how many people there were in my team, but how do you know in advance how many invoices will be created? In the short time I spent reading the manual, I didn't notice any mention of adding records.

One more nitty gritty here. The Manager works out the maximum number of records you could create and sets this figure up underneath the cursor. Suppose it was 3000. Both Mike and I fell into the trap of typing say, 20 and pressing return. What the manager found on the screen was 20 typed over the 30 followed by the original last two zeros. Instead of creating 20 empty record slots in a few seconds, 2000 were created over several minutes, with no get-out, such as the stop key.

Once the format had been stored and the empty record slots created, the Manager returned to the main menu. Operation E loads the module that allows you to enter data into the records.

Once again the program asks you for the name of the file. When you're given it, the appropriate 'mold' appears on the screen with the flashing cursor, but this time without the upward arrows. Instead, the cursor is automatically moved from field to field when you press Cursor Up or Down, Return or Home Screen. Cursor Left and Right, Delete and Insert still work but only within the confines of the field as you have specified

it during the create option. In my case it was just a matter of entering in the data from the teams record books, and skipping over the average fields which were to be calculated later.

By pressing P at any time during this Enter phase, the current contents of the screen are sent to the printer. Illustration 2 shows an entry for one of the team. When all was correct for a particular team member, pressing one key sent the record to disk. The Manager takes care of where it goes for you, finding the first empty or deleted record slot in the file.

The Enter phase has a large number of sub-options. I played about for a while, searching through the file looking for team members who had a batting average of 50 or over for example. Then back to work!

The next task was to specify the arithmetic to work out the averages. To do this I had to create an Arithmetic file. So 'Q' to quit the Enter phase brought back the main menu and 'A' then selected the Arithmetic option.

Once again I was asked for the file name of the main file. A series of questions concerning display positions was then presented. I got stuck on this and had to ask Mike. He explained that these are items, normally calculated, which appear on the screen and can thus be printed using the P sub command, but which are not stored in the main record file. I typed in zero for the number of fields required, and the quiz went on to the actual arithmetic.

It works using field values and constants. For example in this file I needed to calculate 6 different sets of averages. So I needed to do 6 separate calculations, using the following general equation:-

FX / FY TO DFZ

Wher the 'F's stand for field.

The variables X, Y and Z are the field numbers.

'TO' tells the MANAGER to place the result of the calculation in the field number that follows.

'D' is the number of decimal places to be displayed.

It wasn't too hard, counting fields on the work sheet, to derive the following arithmetic expressions :-

F7/F5TO2F10

F7/F6TO2F9

F7/F8TO3F11

F15/F14TO2F16

F15/F13TO3F17

F13/F14TO2F18

You use * for multiply, / for divide. You can't use brackets, but there are 99 registers like those on expensive calculators to help with accumulation and more complex claculations. You can also use an upward arrow to represent "To the power of."

After the last calculation line, the manual told me to type STOP. The disk whirred and once again I was presented with the main menu. Selecting the Enter option, I recalled the main records, and "just like that!", the averages had been calculated and entered.

Updating the input data on a record didn't have an immediate effect on the calculated fields like Visicalc does,

which is a pity, but once you have recorded the modification record and pulled it back, lo and behold the averages are recalculated.

A complete record appears in illustration 3.

This was my first attempt at using the MANAGER. I found it much easier to design and process this admittedly simple application using the package rather than a BASIC program, particularly with formatting for input and output. Now I'm looking into ways of using the MANAGER for more involved ideas, calling for some of the other facilities provided such as SORT, SEARCH and REPORT GENERATE.

Figure 2.

```

SURNAME      TWEATS
INITIALS     D.N.

                DEBUE (DATE)      15/08/81
                DEBUE (PLACE)    CRANAGE

BATTING STATICS
NUMBER OF MATCHES      12
NUMBER OF INNINGS     10
TOTAL RUNS SCORED     100
NUMBER OF BALLS FACED 250
AVERAGE PER INNINGS
AVERAGE PER MATCH
AVERAGE RUNS PER BALL

BOW. STATS.
NUMBER OF MATCHES      12
NUMBER OF BALLS        200
NUMBER OF WICKETS      12
NUMBER OF RUNS         100
AVERAGE PER WICKET
AVERAGE RUNS PER BALL
AV. BALLS PER WICKET

```

Figure 3.

```

SURNAME      BLOR
INITIALS     G.

                DEBUE (DATE)      15/5/80
                DEBUE (PLACE)    CRANAGE

BATTING STATICS
NUMBER OF MATCHES      14
NUMBER OF INNINGS     12
TOTAL RUNS SCORED     3000
NUMBER OF BALLS FACED 3500
AVERAGE PER INNINGS      250.00
AVERAGE PER MATCH      214.29
AVERAGE RUNS PER BALL    .857

BOW. STATS.
NUMBER OF MATCHES      14
NUMBER OF BALLS        2587
NUMBER OF WICKETS      350
NUMBER OF RUNS         3589
AVERAGE PER WICKET      10.25
AVERAGE RUNS PER BALL    1.387
AV. BALLS PER WICKET    256.36

```

Basic Programming

Unit to Unit Copying

Ever needed to copy one or two programs from 4040 to 8050 (or vice versa)? So you pull out your Change Unit Address program, then Copy All, and after answering all the prompts, the files you've selected get transferred. Seems like a lot of work for only a couple of files, doesn't it.

There is another way! Connect your 4040 and 8050 to the PET/CBM and leave the device numbers alone (ie. both device 8). Assuming we're going from 4040 to 8050, insert your source disk in drive 0 of the 4040 and place the destination disk in drive 1 of the 8050. Now DLOAD the program from 4040 drive 0. Both disks units will fire up, but the 8050 will give an error since drive 0 is empty... So What! DSAVE to drive 1 and the 8050 error light will go off and the 4040 error light comes on. Again... So What! Check the directory and you'll find the transfer took place without a hitch!

This would also work for SEQ files with a small bit of software that knows how to ignore anticipated errors. Of course, for a lot of files or programs, the Copy All approach would probably be quicker, but isn't it nice to know you can deliberately cause disk errors and still accomplish something!

IEEE Timeout Defeat

IEEE Timeout is a condition that simply says, "this bus operation took too long!". There are two such situations; Timeout on Read and Timeout on Write.

Timeout on Read has occurred when ST=2. This usually happens when you try to read past the end of an SEQ file. More generally, the controller (PET) has asked a peripheral for a byte and the peripheral did not respond with that byte within the 64 ms. time limit.

Timeout on Write (ST=1) will happen if you write to a file that is open in the PET but not open in the peripheral. For example, OPEN an SEQ disk file for writing and reset the disk or perform a CATALOG command. PET still considers this file open while the disk has gone and closed it. Try a PRINT// now and Timeout on Write will be flagged.

This protocol works great with Commodore peripherals, but unfortunately there exist devices with much slower response times. The biggest offenders are X-Y Plotters.

Fortunately, in all BASIC 4.0 PET/CBMs, there is a feature that allows you to disable IEEE Timeout. When the bus flags timeout, BASIC 4.0 checks location 1020 (hex 03fc) to determine how to handle it.

POKE 1020, 255

If 1020 contains a negative number (bit 7 set), PET will wait forever for the peripheral to accept the byte last delivered to the bus. Be careful though... the PET can hang up if your peripheral doesn't accept this character. This might sound somewhat precarious but it has its advantages. If your peripheral is receiving characters successfully, but just not fast enough, disable timeout and no data will be lost.

To enable timeouts POKE 1020, 0.

Reproduced for Transactor

Dear Editor,

Here's something that might be worth mentioning in Commodore Club News. The timing of BASIC programs is quite strongly (about 10%) affected by an active 'toolkit'-type chip. This is presumably because, when the jump to the 'toolkit' is downloaded into the CHARGOT area, the extensions to BASIC are searched for **every** instruction that has to be interpreted. So, although the 'toolkits' are most useful for editing and debugging, they had better not be loaded when long programs are being run.

Yours truly,

M.J. Smyth
Senior Lecturer
Royal Observatory
Edinburgh

Machine code

by Roger Davis

Many people would like to get involved with machine-code programming, but lack the incentive !! I think this routine will effectively demonstrate the speed of machine-code, and may even get some people

so interested, that they start looking more seriously at 'getting into' machine-code. Anyway, you can at least try it !!!

- 1) Power up your machine, and enter the monitor with SYS 1024
- 2) Type 'm 033a-0352' (RETURN)
- 3) Enter the following:-
033a a2 00 a9 fd 9d 00 80 9d
0342 00 81 9d 00 82 9d 00 83
034a ca d0 f1 60 00 00 00 00
- 4) Exit the monitor by typing x (RETURN)
- 5) Enter the following as a normal BASIC program:-
10 a=1
20 Poke829,a:a=a
+1:gosub100:goto20
- 6) Type RUN RETURN , and see what happens!!

P.S. If you want to save this program, enter the monitor as before, and then type:-

(for disk)

's "O:screen demo",
08,033a,0500'

(for tape)

's "screen demo",01,033a,0500'

(Ed Note: Next issue we will detail the use of the machine language monitor.)

Petit points

by Steven Darnold

A tidy way of putting a time delay into a program is to use the WAIT command. First set TI\$="000000". Then choose your delay from the following table:

1 sec	WAIT 143,64
2 sec	WAIT 143,128
4 sec	WAIT 142,1
8 sec	WAIT 142,2

Longer and shorter delays may be set by altering the value in location 143 & 142. For example, WAIT 143,32 would give a half second delay.

R2D2

by Gary Walker.

Here is a routine which PET users with a sound box on the user port, or electronic bell, may find amusing. It runs for about 20 seconds.

```
10 FOR T = 1 TO 50: POKE 59459,255
15 POKE 59467,16: POKE 59466,85
20 FOR I = 1 TO 15
25 X = RND(TI)*26
30 IF X 20 OR X 255 THEN 25
35 POKE 58464,X: NEXT: NEXT
40 POKE 59466,0: POKE 59467,0
```

Dear Editor,

Just a short note with nothing to do with the compiler series.

There appears to be a bug in the TCL Pascal system distributed free to purchasers of Commodore disk systems. The following program crashes:

```
var I,J : integer;
begin
  for I:= 1 to 255 do
    begin
      wrhex(output,I);
      writeln;
      for J := 1 to 50 do
        end
    end.
end.
```

If this is run using the resident compiler the PET breaks into the monitor with the stack pointer set at \$02. The stack is filled with the return address \$393E, which comes from a JSR \$3942. \$3942 contains a JMP (\$0024), where \$24 and \$25 are set up from a table at \$35AA/\$35BB. This is set up when the P-code interpreter in-

terprets what looks like a CSP (Call Special Procedure) P-code, in this case for the wrhex procedure, where the value of the X register used as index to the table is \$02. This gives (\$24)=\$67 and (\$25)=\$36.

AT \$3667 we have the following code:

```
JSR $367F
JSR $1DCE
JSR $FFCC
JMP $2169
```

where the JMP \$2169 does not execute in RTS instruction but just jumps back into the main P-interpreter loop. Thus the return address pushed to the stack by the JSR \$3492 is never removed from the stack. Replacing the JMP \$2169 with a JMP \$3AA6 appears to cure the problem, but this must be done each time the resident compiler is loaded. Perhaps there is some way to alter the sequential file in the PASCAL LIB 0-series which contains the P-interpreter so that the alteration need only be done once.

Yours sincerely,

John Stout

FIND

J.J. Brown

FIND is a program which displays the number of the lines which contain specified TOKENS (reserved words) and/or STRINGS. This program works with all 3 versions of COMMODORE Basic. (Basic 1,2, and 4)

FIND can be screen merged (appended from the screen to a program loaded from Cassette) in the following manner.

Load or type in the FIND program. List FIND on the screen.

Without clearing the screen load the program to which FIND is to be merged. (Ensure the screen does not become cluttered with the titles of unwanted programs.)

Without clearing the screen load the program to which FIND is to be merged. (Ensure the screen does not become cluttered with the titles of unwanted programs.)

Without clearing the screen, move the cursor to the HOME position. Using the CURSOR DOWN key move the cursor to the start of line 44444. Now press return so that the cursor progresses down the screen until it is BELOW the last line of FIND. (Line 44464)

FIND will now be merged to the end of your program.

NB: If your program already uses line numbers 44444-44464 then renumber FIND before use. (In line 44452 ensure that R is equated to the new start line number.)

To use the FIND.

Type in RUN 44444 (RETURN) In response to the prompts enter the TOKEN NUMBER (from the list below and the required STRING. If you do not want to search for a TOKEN then enter 127.

If you do not want to search for a string then enter #*\$ or some other unlikely group of characters.

As each line is examined the line number will be listed on the screen.

Whenever the specified Token or String is located in a line, a message will be printed alongside the line number.

The line number and the message will be repeated for each occurrence of the Token or String.

Provided the string is input within quotation marks it is possible to locate graphic symbols, characters

TOKENS.

Numerical Order. (By Rows)

END	128	FOR	129	NEXT	130	DATA	131
INPUT#	132	INPUT	133	DIM	134	READ	135
LET	136	GOTO	137	RUN	138	IF	139
RESTORE	140	GOSUB	141	RETURN	142	REM	143
STOP	144	ON	145	WAIT	146	LOAD	147
SAVE	148	VERIFY	149	DEF	150	POKE	151
PRINT#	152	PRINT	153	CONT	154	LIST	155
CLR	156	CMD	157	SYS	158	OPEN	159
CLOSE	160	GET	161	NEW	162	TAB(163
TO	164	FN	165	SPC(166	THEN	167
NOT	168	STEP	169	+	170	-	171
*	172	/	173	^	174	AND	175
OR	176	>	177	=	178	<	179
SGN	180	INT	181	ABS	182	USR	183
FRE	184	POS	185	SQR	186	RND	187
LOG	188	EXP	189	COS	190	SIN	191
TAN	192	ATN	193	PEEK	194	LEN	195
STR#	196	VAL	197	ASC	198	CHR#	199
LEFT#	200	RIGHT#	201	MID#	202		

Alphabetical Order. (By Columns)

ABS	182	IF	139	READ	135	TAB(163
AND	175	INPUT	133	REM	143	TAN	192
ASC	198	INPUT#	132	RESTORE	140	THEN	167
ATN	193	INT	181	RETURN	142	TO	164
CHR#	199	LEFT#	200	RIGHT#	201	USR	183
CLOSE	160	LEN	195	RND	187	VAL	197
CLR	156	LET	136	RUN	138	VERIFY	149
CMD	157	LIST	155	SAVE	148	WAIT	146
CONT	154	LOAD	147	SGN	180	+	170
COS	190	LOG	188	SIN	191	-	171
DATA	131	MID#	202	SPC(166	*	172
DEF	150	NEW	162	SQR	186	/	173
DIM	134	NEXT	130	STEP	169	^	174
END	128	NOT	168	STOP	144	>	177
EXP	189	ON	145	STR#	196	=	178
FN	165	OPEN	159	SYS	158	<	179
FOR	129	OR	176				
FRE	184	PEEK	194				
GET	161	POKE	151				
GOSUB	141	POS	185				
GOTO	137	PRINT	153				
		PRINT#	152				

with leading or trailing blanks and cursor control characters, etc.

This is particularly useful when searching out variables.

FIND is very useful in locating all specified occurrences of PEEK or POKE when upgrading programs from one version of Basic to another.

```

1 rem "J #1" "ka l "D???" (*)

```

I do not know how many of you are familiar with the above West-Sanskrit idiom. Actually, it is only rarely used nowadays, in spite of its effectiveness. Even I almost forgot it, - one has to be really frustrated to remember such things...

Well, to make a long story short, I was working on one of those complicated programs, with a lot of disk handling, when suddenly the error-detection sub-routines begun reporting errors, and stopping the program, of course. Now my little Pet has always pointed out the fact, that the program stopped right in the middle of the subroutine, - but where was the routine called from??

Fortunately, I remembered my kindergarten Sanskrit, so I typed in the following (nasty) remark (highlighting the "la" and the "ka" to make it more forceful):-

```

1 rem " (RVS)la(RVS OFF)
(RVS)ka(RVS OFF) 1 D"

```

At the start of my program I inserted:
poke1,8:poke2,4
and all I had to do within the subroutine, was to put
print usr(0)
to have the calling line number printed on the Pet screen...

By the way, for those with Graphic keyboards, the " " is a shifted ":", And another thing. In order to be able to print reverse characters, you have to start the rem statement with two quotation marks, deleting the second one. Do not forget the space after "rem". It is important. That's all Folks. And watch your language!...

Peter R. Gabor

(*) Where the Hell have you been

```

FIND
-----
44444 J=1024:PRINT"ENTER TOKEN#,ITEM";
44446 INPUTE,Q#:FF=LEN(Q#)
44448 B=PEEK(J+1)+PEEK(J+2)*256
44450 R=PEEK(J+3)+PEEK(J+4)*256:PRINTR
44452 IFR=44444THENPRINT,"END":END
44454 FORO=JTOB-1:W=PEEK(O):A#=CHR$(W)
44456 IFW=ETHENPRINT,"TOKEN IN LINE":R
44458 B#=B#+A#:NEXTO:FORN=1TOLEN(B#)
44460 IFMID$(B#,N,FF)C>Q$THEN44464
44462 PRINT,"LINE":R;"CONTAINS ";Q$
44464 NEXTN:B#="" :J=B-1:GOTO44448

```

```

1 REM "J #1" "ka l "D???" (*)
2 :
10 REM *****
20 REM * DEMO PRGM FOR SBR SOURCE # *
30 REM * 16/11/81 *
40 REM *****
50 :
60 REM ENTER REM STATEMENT AS FOLLOWS:
70 REM ",",DEL,↑,↓,←,→,↖,↗,↘,↙,↕,↔
75 REM RVS,K,A,↑RVS,L,↖,↗,↘,↙,↕,↔
90 :
100 POKE1,8:POKE2,4:A#="SBR CALLED "
200 A#=A#+ "FROM LNE #":PRINT:GOSUB1000
210 PRINT" " # # # # "
250 GOSUB 1000
300 END
400 :
1000 PRINTA#:USR(0)
1010 GOSUB2000:RETURN
1100 :
2000 PRINT"2ND ";A#:USR(0)
2010 RETURN

```

Dear Editor,

I have just discovered an interesting feature of the PET's method of deleting lines. It seems that the PET searches the BASIC program in order, looking for the line number it has to delete, until it finds a line with a higher number, at which point it gives up and assumes the line did not exist. This allows certain lines of a BASIC program to be protected from deletion e.g. copyright notices, routines to ask for passwords, etc.

To see how this works enter the following dummy program :

```

1 REM TEST PROGRAM
6000 REM TEST PROGRAM
60001 REM PROTECTED
LINE

```

Now type

```

P = PEEK (1025) + 1024 : P1 =
PEEK (P) + 1026:POKE P1, 0 :
POKE P1+1 , 0

```

LIST

Line 60001 will have been re-numbered as 0, and any attempt to delete line 0 will fail, unless the whole of the rest of the program is deleted first (or the line number is poked back to 60001). A real program can now be entered where these three lines are still in storage, and then lines 1 and 60001 may be deleted to leave the protected line as part of the new program.

Yours Faithfully,

David Lewis.

CBM 8010 Auto-Answer Circuit

The schematic on the following page is an automatic answering device for the CBM 8010 IEEE Modem. It was designed by Dieter Demmer of Oakville Ontario. The circuit hasn't been tested by us, but presumably it works for Dieter!

Machine language monitor

When a CBM or PET computer is turned on the first thing is displayed is confirmation that one of the three Commodore BASIC versions has been loaded and is now running. This is confirmed by the machine displaying 'READY' mode and displaying a flashing cursor. The machine is now ready to accept and run the BASIC interpreter, which has the task of converting BASIC text into machine code instructions which the 6502 processor can respond to. It can be seen that if we write programs directly in machine code, faster running speeds will be obtained as the interpreter will not be required to operate. Assembly language lends itself to such tasks, as performing I/O operations.

A Machine language Monitor is provided on all CBM/PET microcomputers (except original 2001) for use in editing, displaying and changing memory locations. TIM (Terminal Interface monitor) is the program for MOS Technology's 65XX microprocessors that allows the use of these commands. This program has been expanded and adapted to function on the Commodore PET/CBM range of microcomputers. With the original PET with calculator keyboard and using BASIC 1.0, TIM is loaded from cassette. All other Commodore versions of BASIC have the monitor in ROM.

Commands directed to TIM via the CBM keyboard and then displayed on the monitor screen allow TIM to commence program execution, display, or modify registers and memory locations and for LOADING or SAVEing data via any of the devices available for use by the processor. TIM also performs an automatic read write verification to ensure that addressed memory exists, is of R/W (read/write) type and is responding correctly to operations. These provide the basic facilities needed to operate and construct a machine code program. TIM also provides several subroutines which may be called by user program. These include reading and writing characters on the video display, typing a byte in hexadecimal and typing a CRLF sequence.

TIM monitor calls and special locations.

Kernal Name	Vector Pointer	Description
WRT	\$FFD2	type a character
RDT	\$FFCF	input a character
GET	\$FFE4	get a character
CRLF	\$FDD0	type a CR
SPACE	\$FDCD	type a space
WROB	\$E775	type a byte
ROOB	\$E7B6	read a byte
HEX-		
IT	\$E7E0	ASCII to HEX in A

TIM offers six basic commands that provide the ability to use machine code without the use of an Assembler/Disassembler.

The six TIM commands are:

R	display registers
M	display memory
X	exit to BASIC operating system
G	begin execution of any routine
L	load data into memory
S	save data from memory

These commands allow simple handling of machine code text or routines and facilitate scanning of BASIC text, the SAVEing and LOADING of programs, altering and/or amending programs and to aid general program development.

The way to activate TIM is to make the CPU execute a break (BRK) command, which will then transfer the operating system to TIM and TIM conventions.

This is done via a SYS command which is normally used to transfer control to machine code routines. By typing either SYS 4, SYS 1024 or SYS 64795, the processor is forced to execute the instruction/s held at these addresses. In these cases HEX 00 comand, (a break) is encountered, so activating TIM. The monitor will respond with a Register display preceded by a 'B*' which indicates that entry to the monitor was via a breakpoint, rather than 'C*' denoting entry via a call.

The cursor is also returned, denoting that TIM is ready to accept commands. The flashing cursor is

preceded by a full stop (.) which is displayed at the beginning of all lines when TIM is in operation. The various registers are also displayed in the following form.

B*

```
PC IRQ SR AC XR YR SP  
.: 0401 E455 32 04 5E 04 F8
```

R — DISPLAY REGISTERS

This format is also shared by the 'R' command which stands for display registers and is executed by typing 'R' and pressing RETURN. The display consists of:

1. PC (the program counter) indicates where program execution would start if a RUN command was executed, which is why this value changes, depending upon which entry point was chosen.

2. The IRQ is a representation of locations 0090-0091 which are indirect vectors to the 1/60 second interrupt. This interrupt scans the keyboard and updates the clock, amongst other operations needed for computers to operate (these operations are sometimes referred to as 'internal house keeping').

The purpose of this is to show if the interrupt is being vectored to a machine code routine, which is being executed every interrupt and may be the cause of corruption.

3. The processor status register (SR). This is the status flag which the processor uses to convey information based on calculations performed. These are in turn used by branch commands which check for zero, negative and other such answers to calculations. The flag can be used to check branching based upon this flag.

4. This is a display of the contents of the accumulator which is an 8 bit memory location dedicated for variable storage (as are the X and Y registers). This register is used for many mathematical calculations, general storage and manipulation of numbers.

5. & 6. (XR and YR) These are representations of the two registers which are at the disposal of both the processor and machine code programs, (the X and Y index registers,)

which can be manipulated in the same way as the accumulator.

7. (SP) The stack pointer is responsible for the handling of interrupts and subroutines. It is used to point to addresses in page 1 (i.e. 0000-00FF) and is used to generate addresses in the same way as the program counter does for calculating the address of the next line to be executed.

M — MEMORY DISPLAY

This command displays sections of memory for either verification of memory/machine code text or for modifying or entering text. Both the start and ending addresses must be completely specified as a 4 digit Hexadecimal address. If just one location is desired the instruction will operate, but the eight following bytes will also be displayed as shown.

```
.M 0400 0400
```

```
.: 0400 00 00 00 AA AA AA AA AA
```

This format is used in all memory displays. When displays will not fit onto the screen, the screen scrolls in the same manner as LISTings do. These screen displays can be slowed down in the same way as LISTings. TIM does not specify that a space be entered between the two addresses, in fact any character maybe used as a separator.

X — EXIT

The 'X' comand does nothing but engage the BASIC Interpreter and then fall into the READY mode; with the flashing cursor displayed below the 'READY'. The stack values saved when entering will be restored. Care should be taken that these values are the same as when the monitor was entered. A CLR executed in BASIC will re-adjust all pointers, (including the stack), if any problems are incurred. The 'X' instruction does not alter in any way the contents of memory.

```
.X  
READY
```

G — BEGIN EXECUTION

This command begins execution of machine code at either; A) the address held in the PC register (all the registers will be replaced with the displayed values) or (B) at a 4 digit Hexadecimal address.

```
A)  
PC IRQ SR AC XR YR SP  
.; 033A E455 XX XX XX XX F8  
.G  
B)  
.G 033A
```

L & S — LOADING and SAVING

The loading and saving procedures are in much the same format as when using BASIC, with the exception that all normal default values must be entered. To save a BASIC program you must specify the device number required, the address to start saving and the end address; which one normally accepted as the pointers to the start and end of BASIC text.

If an error incurred during this or any TIM command, a question mark will be positioned just in front of the error. For example, if the filename is too long (over sixteen characters) a '?' will be located one character after the correct file name length.

```
.: 0408 48 45 4C 4C 4F 22 00 00  
.S'this filename is ?o long'',  
01,0406,040F
```

To rectify this mistake, use the cursor control keys to move the quotes to the position before the question mark. This when executed will then respond with the messages PRESS PLAY & RECORD ON TAPE #1 as in BASIC. This is because we have specified device 01, as being cassette unit #1. Once the operation is complete, the cursor will be returned after the (.) denoting TIM is still in operation. However, if the run/stop key is pressed, the machine will drop into BASIC mode after indicating a break has occurred.

This command is very flexible, allowing any section of memory to be saved to any of the devices on the IEEE or cassettes, excepting the screen and keyboard. The 'L' or Load command only requires the character L and return to be pressed as when using cassettes as in example A), or with the filename to be searched for with device number as in B).

```
A ) '  
. L  
PRESS PLAY ON TAPE #1  
B)  
. L "FILENAME",01
```

Here is a small sub-routine which may be of interest to some of your readers. Also, a word of thanks for an excellent magazine/newsletter - I find it most informative and interesting.

With the advent of Relative Files and the large storage capacity of the CBM 8050 Disk drive, some form of 'ISAM' (Indexed Sequential Access

Method) would be useful to make full use of these facilities. Here is a version which meets most of the specifications of ISAM, but is relatively (excuse the pun!) easy to use. It works as follows:-'

An ordinary sequential file is used to store a 'key-file' of all records held within a System (eg: Stock, Accounts, etc.) while the main records are held in a Relative file. The key-file would normally contain the first 10 characters or so of a key-field in each relative record (eg: the Customer's Name when dealing with some sort of Client File). Attached to each of these key-records in the key-file **would be the Relative Record Number of the main Customer's Record.** Right - now all you have to do is search through this key-file until you find the record you're looking for (using the customer's name or part thereof as the search-string); retrieve the relative record number and you have access to the main record. The only problem in doing this in BASIC is time - especially if you have 500 or 1000 records or more.

Here is a machine-code routine which will do the above significantly faster (it searches thru 500 ten-character records in approximately 4 seconds - not including the rather slow start-up times of the 8050 Drive). This routine may only be used with BASIC 4.X and DOS 2.X. Here is how you use it:-

The length of each record in the key-file (SEQ) is not important and it may contain any valid ASCII character (for safety's sake, stick to alpha-numerics only). To separate the key-record from the relative record number a delimiter must be used. In this case the delimiter is a '#/' symbol (if preferred, another symbol may be used - just change the character in line 0053 of the source-list). Therefore a record in the key-file should look something like this: SMITH# 1234. The space between the delimiter and the rel/rec number is the sign associated with the number and can be suppressed if space-saving on the disk is necessary. It is important that the records in the key-file are separated by Carriage Return symbols -CHR\$(13). This should'nt present any problems as the PET automatically sends this symbol after each 'PRINT#' command.

The ISAM machine code program must be located at the top of memory and protected in the usual way (ie:

Educational Editorial

Welcome to the first educational supplement of 1982. The first of many, and the promise of a lot of interesting articles for you over the next 12 months. Many of these were brought about as a result of the very successful PET in Education conference in November last year, and quite a few are copies of the actual talks given by the numerous distinguished speakers who attended that conference : these will be reproduced in subsequent issues of the magazine.

The conference was so successful in fact that another is planned for this year, although where and when I don't, as yet, know. Details will be given in the education section of the magazine as soon as I get them.

Among the talks that I plan to reproduce will be ones from such luminaries as Nick Green, Commodore's Special Projects Manager, and famous outsiders like Danny Doyle, Chris Smith, Borge Christensen, Bob Lewis, Trevor Lusty, Nick Hampshire et al. These talks cover many topics, but, as you might surmise, all are connected in some way with the role of microcomputers in education. Feedback on these articles would be most valuable in assisting for the planning of next year's conference, so I would appreciate any comments you might have. You can reach me at the new address given in the magazine editorial.

But that's all intended for future months : What have we got in January 1982 ?

The first contribution comes from the Open University, that august body of people who specialise in teaching people out of working hours. To keep pace with the continuing growth of microcomputers in industry and commerce, they're launching a complete series of courses aimed at a) managers in industry and b) engineers who want to know how to use microprocessors in product design. The article inside goes into more detail on the courses, and also gives a general run down on managing microprocessors.

Just to prove that the SuperPET 9000 exists, and is not just a Commodore marketing stunt, we have a



report from Westminster City School in London concerning the beast. The fact that they have three SuperPETs there is perhaps some indication of the way Westminster City feel about the machine, and as their principal use of it is using the APL programming language option, that must say something about the implementation of the language. But read on, for the views of Chris Beattie, computer science teacher at the school.

A scene from the future next, with a humorous look at educating a child in 1991. Somehow, I can't imagine it being quite like this, but you never know ...

Magazine of lists time

The last two sections in this educational supplement are slightly different, in that both are lists of names and addresses. The first will probably require some explanation, so here we go.

Almost as soon as the PET appeared, a body of people set themselves up as the Independent PET Users Group. Not through any great dissatisfaction with the Commodore Users Group, but just so that they could be truly independent. Thus when any member of the group spoke at meeting etc. the people knew they were getting an unbiased point of view. However much I personally try and maintain neutrality in talking

about Commodore, the fact that I've worked here for over 3 years means that I can never be truly unbiased, so I can appreciate the I.P.U.G. viewpoint.

Commodore and I.P.U.G. have shared and differed opinions many times over the years, but generally speaking we've enjoyed a mutually close working relationship. To this end we like to help each other out wherever possible, and it is this 'help' that brings me to the list.

Maintaining one national independent group was soon found, administratively, to be at best an irksome task, and at worst an impossible one. Consequently a number of regional groups were set up to help share the load, as it were. These groups exist to assist PET users both new and old, in the form of magazines in one respect, but far more importantly by holding regular meetings where people can turn up and air their views, ask questions, and so on. The list towards the back of this supplement is a collection of these independent groups. If you live anywhere near any of the, you'll find it well worth going along.

Finally for January 1982 we have the complete, up-to-the-minute list of educational workshops. Each and every one of them, a total of 293 in all. Good luck to them all!

Managing Microprocessors

How big is the gap in British industry and commerce between talking about the new microelectronic technology and actually applying it.

According to a Department of Industry survey in 1977, done just when the word chip was beginning to disassociate itself from fish and vinegar, only some five per cent of firms were taking action to apply the new technology. A further 45% were broadly aware but not taking action.

Five years later, there is new survey evidence from the Policy Studies Institute (PSI) that the situation has much improved - half a national sample of 1,200 manufacturing points are using microelectronics in their products and processes or were planning to do so.

But the other side of this coin is that half of these manufacturing points were **not** using microelectronics and were **not** planning to do so.

The PSI survey represents about 70% of all manufacturing establishments in the UK employing 20 people or more and about 45% of total employment in manufacturing industry.

Over half of these non-users agree that there is scope for applications in their area but are not doing anything about it now or planning to in future.

And of those who see potential scope for micro applications in their products but are not using it, some 37% think their overseas competitors are using them already. The figure for process applications is 24%.

So while there is clearly a process of adaptation going on to what is going to be the dominant technology for the remainder of this century and much of the next, there is still much awareness creation and education to be done.

It was against this background that four years ago the Department of Industry and the Open University agreed to create courses to help along this process. The decision was in line with the University's declared policy of providing education and training opportunities to help with the updating and re-skilling of British industry. To this end the OU will pro-

duce in the near future scientific and technological updating packages with a £2 million grant from the Science and Engineering Research Council and also enter the management education field.

But three years ago, as an initial step, the OU launched its self-study package about managing microprocessors entitled 'Microprocessors and Product Development' designed for decision-takers and this summer added a similar package for engineers called 'Microprocessors and Product Design'.

Both courses were written by electronic experts from the OU's Technology Faculty and funded under the Microprocessor Application Project, a national attempt to facilitate the introduction of microelectronics. These packages include specially written books; technical files; and experimental kits to illustrate theory and develop practice.

The OU feels the first package, the one for managers, has done well with some 4,000 sale and an estimated pass-on-rate among work professional colleagues of the actual purchaser of one to five. Thus some 20,000 British managers have studied the effects and implications of the new technology from an all-round stance.

This has been achieved without managers away from their workplace with all the attendant disruption and expense. Instead the OU's method of distance teaching gives the individual the choice of study time and study place, for the packages are self-contained and free of the obligation to watch fixed-time broadcasts or to travel to a centre.

A survey of the managers package by the University's Survey Research Department showed that it was considered relevant by users and of high quality.

Against this generally favourable user reaction, the OU has launched its package for engineers and it is hopeful that it is using a proven formula which goes a long way to meet substantial need.

Below the two courses are briefly described. Besides authoritative written material, the courses come com-

plete with hardware and guides for its use and suggested experiments. In the case of the engineers course, this amounts to a fully assembled microcomputer based on the INTEL 8085 microprocessor. The overall aim is to illustrate theory, integrate it with practice and give a measure of 'hands on' experience.

'Microprocessors and Product Design: A course for Engineers' shows how to use microprocessors in product design and covers the complete sequence from customers' specification to final design stage.

It does not assume or require previous knowledge of microprocessors and involves about 150 hours study.

It comprises five books specially written for self-study; a file of data sheets and technical literature; a fully assembled microcomputer system based on the INTEL 8085 microprocessor with full alphanumeric keyboard which interfaces with students' own TV sets and cassette recorders; a prototype development board to be driven by the microcomputer in various configurations; a user manual for the microcomputer and experiment books containing practical work. The course fee is £395. There are no TV broadcasts or tutorials, so allowing students to fit study in as best suits them.

The course follows through the complete design sequence for microprocessor-based products; customer specification; overall system design; hardware and software development; prototype evaluation and production design. Case studies are brought in to illustrate points made and short self-assessment questions allow students to check progress.

The experimental work does not have to be done at the same time as reading texts. The initial experiments familiarise students with use of the microcomputer and peripherals, while later ones follow the design sequence for a microprocessor-based product.

'Microprocessors and Product Development: A course for Industry' is designed to give managers in in-

dustry an understanding of how the process of developing a product is affected by the introduction of a microprocessor. Previous knowledge of microprocessors is neither required nor assumed.

This self-study package from the Open University explains what microprocessors are, shows how the new technology can be applied and illustrates what is involved in developing a microprocessor-based product.

As well as the development process, the course looks at some of the per-

sonnel, financial and other effects on industry of introducing microprocessors, and the effects on products themselves.

The course comprises six specially prepared teaching texts, a small, ready-to-use microprocessor system designed to show you how a microprocessor is used and the tasks it can perform and an experiment book describing the microprocessor systems and the practical work. The fee is £150.

For a brochure and order form, please write to the Learning materials

Service, Open University, P.O. Box 188, Milton Keynes, MK3 6HW or telephone (0908) 79058 (24 hour service).

NOTE TO EDITORS: The Policy Studies Institute report referred to is "Microelectronics in Industry: Extent of Use". It is available from 1/2 Castle Lane, London SW1E 6DR. Telephone 01-828-7055. Enquiries to: Kevin Moloney Telephone Milton Keynes (0908) 653248/653931 (Office) Leighton Buzzard (0525) 371011 (Home)

Super Pet Scene - 1991 - Domestic Division

"If you can spare a minute Son, your mother and I thought it would be a good idea to try and chat to you about a few facts of life. After all you ARE fourteen".

"Go sub Dad. All disky checksum".

"Err.... yes! Could you try and imagine you are talking to a human being for a moment and not those machines upstairs. We wonder now whether we did the right thing five years ago when you asked for a Super Pet. We had in mind a nice homely dog or cat".

"O.K. Dad, what's scrolling?"

Now, these days, voice recognition and command control of personal computers are fairly common. Also impressed codes on sound from the radio or television can automatically load a home program on disk without distracting from the entertainment value of what was being heard or soon.

Most children are learning computers at the age normally associated with listening to bed-time stories.

When a child learns to input questions, the computer verifies any statement, especially those of parents. Parents are updated frequently. A child's early struggles are with understanding the illogical language of most adults.

No self-respecting parent is without one disk serial, preferably situated prominently. This symbol overtook the status of new cars.

"Well! Son! Your mother and I have bent over backwards for nine years keeping your Super Pets fed with peripherals whatever that means. We haven't complained about

our cars transformed into mobile receivers and transmitters or even your take over of the top floor of the house. We sleep better downstairs especially in winter. Its warmer. We appreciate your total rejection of alcohol, drugs, pornography and smoking. We do worry though at not seeing you in the flesh for days. Communicating on our screens doesn't seem quite the same, especially to your mother. She worries about the 'send along meals' you order via satellite. Foreign food is a little dicey you know. We do value the health check-ups every week but we think we'll skip the personalised predicted life spans if you don't mind. The biorythms confuse your mother. Can you understand what I've been saying so far Son?.

"Pointered. Long string. Re-programmed. Traced. Renumbered. Yes Dad! Er! Did not you learn any language at school?

"I have to talk in old English because its the main point in earning my living. However, very soon you will have to think of earning money.

We can't support you for ever. That means in basic - a job. Do you read me?"

"Synchronised. CMD opened. Coursed COC⁵. Result wait. If no error then go to Zivania or Pandask on eighteen. Confirm hard copy Hand shake Commodore".

"Do I guess or would you enlighten the long suffering generation?"

"Well here goes. Bear with me. I've taken the Commodore Open Course for a degree in space communications. I expect a rush from Boston any day on the dish although I'm sure I've passed. If I do I shall spend a year or two in the U.K. and U.S.A. learning the newer developments. I shall go to Zivania or Pandask to open up Technical Sales there. I know what the places look like and how they tick from a disher mate in Africa. Commodore will put this in writing to you."

"Good God! You did all that upstairs just sitting on your behind. You can even talk sense. By the way, I'd like your advice. Your sister has just asked us for a pet. Do we buy a dog, a cat or a tortoise?"

Independant Commodore Product Users Groups

Region K — Birmingham Area: J.A. McKain P.P.I Ltd., 177 Lozells Rd, Birmingham. Tel: 021-554 0202.

Region K — South Midlands: M.J. Merriman, 12 York St., Stourport-on-Severn.

Region L — Derby Area: R. Davies,

105, Normanton Rd, Derby, Tel: 0332 41025.

Region N — Manchester Area - We have the venue, but need someone willing to organize the group.

Region N — Blackpool Area: David Jewett, 197 Victoria Road, East,

Continued on page 5.

Westminster City School's Computer Experiment: 1

A staid London boys school is about the last place one would expect to find a radical experiment in computers being undertaken.

Yet Westminster City School's computer science course includes the "Westminster Project" - a programme to evaluate the "claims made on behalf of APL", which is unique in the U.K.

Westminster computer science teacher, Christopher Beattie's objectives are to evaluate the suitability of APL (A Programming Language) as a language and to develop a mathematics laboratory based on it for use in British schools.

APL is a powerful computer language which is claimed by its inventor, American Ken Iverson, to be ideal for teaching mathematics - although it has additional uses.

While APL is a common computer language which enables the programmer to write short concise programmes, it is a problem solving language for people who don't want to understand how computers work.

"APL is perhaps, 10 times more powerful than Cobol or Basic. It will do in one line, what may require 10 lines in Basic. Despite this, or may be because of it, APL is not popular with computer people in this country who have been brought up on other languages.

"For the Westminster Project we are working with first form pupils - 11 year olds - and introducing them direct to APL, and while it is a complex language they are having no trouble coping with it."

Mr. Beattie's primary interest in APL lies in its potential as a language to teach mathematics, instead of using standard maths text books, and his ultimate aim is a government sponsored programme of maths teaching using APL and computer-based maths laboratories.

Few micro-computers use APL as a language so that obtaining suitable hardware for the evaluation programme was a problem, then Commodore heard of the project and supplied three special SUPER PET SP9000 machines for the test.

These are specially modified versions of the PET 8000 series machines, developed in conjunction

with Waterloo University in Toronto, Canada.

Most Commodore PETS use a Basic as their language but these SP9000 machines can be programmed in Basic, Waterloo Basic, Pascal, Fortran or APL - with the possibility of Cobol being added in the future.

They differ from the 8000 series in being fitted with a special keyboard, and they are designed to interface as "intelligent" terminals with a mainframe computer.

The three pre-production machines being used at Westminster School are being given the most thorough work-out imaginable, as work often starts at 6.30am and goes right through until 7.30pm, 50 weeks a year - the computer department doesn't shut for the holidays.

During school hours the computers are used by sixth formers, but in the evening the 11 year olds have priority working through until 6.30/7.00pm.

"The sixth formers, most of whom have done some work with computers, mostly use Basic, although two are looking at APL, and one is keen on Fortran. The first formers who are new to computers are using APL as a matter of course, and working confidently with it, after three or four days. "While we have other computers in the classroom, most of the boys prefer to work with the Commodores, even though they have very intricate keyboards to cope with the different languages.

"The Commodore SP9000 are just about ideal for our purposes," Mr. Beattie said, "and the indications are that they will be cheaper than the other micro-computers which can use APL - and money is very important when one talks about using them in schools."

The Westminster Project is supported by a working party which includes representatives from a number of outside companies which use APL in their work. So part of the project includes visits by guest speakers who talk on the practical applications of the languages in commerce and industry.

"Practical applications are important. APL is a very contemplative language. It teaches students to think, recognise complex patterns and

motivates them to solve problems - but we need to show that it has practical uses beyond playing games.

"We are the first, and so far the only, school to use APL. It is used in business, but not in the universities. The fact that the Cambridge "O" Level examinations may now be taken in any computer language may give APL a boost - although it will need moderators to examine the projects." Mr. Beattie is concerned with "creating a learning environment", and is doing it while undertaking work which could be important to the future of computer education in British schools.

His long term objective is the application of APL to the teaching of mathematics - "APL is good for simulation. It is used in industry by electrical engineers and others who need this capability - but it will simulate mathematical and other problems. I can see uses for it outside of the computer sciences umbrella.

"Iverson invented it to teach maths, not as a computer language, originally, but as far as I know we are the only school experimenting with it in this context - certainly in the United Kingdom.

"To do the job properly we'd need to set up what I call Mathematics Laboratories - though they would be nothing like the old language teaching laboratories - using maybe a dozen SUPERPETS... the more you have the more you need."

Mr. Beattie estimates that it would require about £40,000 to set up a satisfactory working arrangement. Westminster City School doesn't have that sort of money. The project would need government support, but if APL proves to be as effective as Westminster's early evaluations suggest it could be of great benefit to the country, helping Britain to keep up with the need for trained computer personnel caused by the rapid development of computers in industry.

If the enthusiasm of the students of Westminster City School who queue to use the Commodore SUPERPETS, is any indication the Westminster Project will be a success - regardless of the results of their evaluation of APL.

Continued from page 3.

Thornton-Cleverleys, Blackpool, Tel. Cleverleys, 869108.

Region N — Liverpool Area: Tony Bond, 27 Ince Rd., Liverpool 23, Tel; 051-924-1505.

Region P — London - Barry Miles, Department of Business Studies, Polytechnic of North London, Holloway Road, London N7. This is the PET group of the Amateur Computing Club which meets every other Tuesday at the Poly at 6p.m.

Region P — West London - Please contact Geoff Squibb, 108 Teddington Park Road, Teddington, Middlesex. Tel: 91-977-2346.

Regions S — Durham Area: Jim Cocallis, 20 Worchester Rd., Newton Hall Estate, Durham tel (0385) 67045. Meetings at Lawson School, Burtley.

Region T — Scotland: Dr. Jim MacBrayne, 27, Paidmyre Crescent, Newton Mearns, Glasgow. Tel 041-639-5696.

Region W — Yorkshire: Bob Wood, 13 Bowland Crescent, Ward Green, Barnsley, S. Yorks. Tel (0246) 811585.

Any queries, offers of help, information etc. please get in touch with Mrs. E. Pamphlett, 7 Lower Green, Tewin, Welwyn, Herts. Tel:(043871) 7325.

Region B — Glos./Bristol Area: Mrs. Janet Rich, 23, Sheppard Leaze, Wotton-under-Edge, Glos.

Region C — Hampshire Area: Ron Geere, 109, York Rd, Farnborough, Hants.

Region D — South East Region: Wing Cdr. Mike Ryan, 164 Chesterfield Drive, Riverhead, Sevenoaks, Kent.

Region D — Canterbury Group - For details please contact John Bickerstaff, 48 Martin Down Road, Whitstable, Kent, Home tel. 0227 272702 Office tel. No. 01-499-9102.

Region G — Essex Area - Please contact Walter Green, 151, The Hatherley, Basildon, Essex for further details.

Region H — Norfolk/suffolk Area -We need a volunteer in this area.

Region I — Herts/Watford Group -Stephen Rebagliati, c/o Institute of Grocery Distribution, Grange Lane, Letchmore Heath, Watford.

Region I — Herts./Stevenage Group: Brian Grainger, Tel: Stevenage 727925.

Region J — Wales - Contact - F.J. Townsend, The Mill, Rhydowen, llandyssul, Dyfed, SA44 4QD. Tel: 05455 5291.

Educational Workshops

College Name and Address	No of PETs	Courses for Teachers
LONDON		
Mr J Bennett, Computer Services, Middlesex Polytechnic, Bounds Green Rd, London N11	3	Yes
Mr A Lenney, L B Haringey, Chief Education Officer, 48/42 Station Road London N22	6	Yes
Mr R Bradbeer, Poly of N London, Holloway Road, London N7	20	Yes
Mr A T Ley, Moat Mount School Worchester Cresent, Mill Hill, London N7	1	Yes
Mr R C Lowe, Thames Poly, School of Surveying, Wellington Street, Wollwich London SE18	3	Yes
Ms C Pope, CHelsea College, Bridges Place, London SW6	2	Yes
Mr P Redman, Peckham Rye School, Whorlton Road, London SE15	2	
Mr D Simpson, Whitefield School, Claremont Rd, Cricklewood, London NW2 1BY	3	
Mr P J Wayth, St Andrews CE Juniors, Belton Road, Willesden, London NW2	1	Yes
Mr T Lyth, South Thames College, Wandsworth High Street, London SW18 2PP	6	Many
Ms Z Isaacson, Poly of N London, Prince of Wales Road, London NW5	5	Yes
Mr D St J Gray, Sidney Chaplin High School, Folly Lane, Walthamstow, London E7	3	H + A
Mr P Gibson, Rowan High School, Rowan Rd, London SW16	4	Internal
Dr C R Mitchell, Walmsley House, The City University, St John St, London EC1	2	No
Mrs Tayler, Woodhouse Vith Form College, Woodhouse Road, London N12	4	Internal
Mr J R de Boer, N E London Poly, Forest Road, London E17 4JB	15	Yes
ESSEX		
Mr F Bonner, Kingsdown School, Snake Lane, Southend-on-Sea, Essex	5	Yes
Mr V Buhagiari, Bishop Ward Comp School, Beacontree Heath, Dagenham, Essex	2	Yes
Mr R W Stamp, Culverhouse School, Daiglen Drive, S Ockendon, Essex RM15 5QH	2	H + A
Mr B Richardson, Janet Duke Junior School, Markhams Chase, Basildon, Essex	1	Yes
Mr V Petch, Harlow Technical College, Harlow, Essex CM20 1LT	5	Yes
Mr K O'Shea, S E Essex VI Form College Kiln Rd, Thundersley S Benfleet	3	Yes
Mr P Clarke, Barking College of Tech, Dagenham Rd, Romford, Essex	5	Yes
Mr L Parren, Advisor for Computer Ed, 4th Floor, Lynton House, 255/259 High Road, Ilford, Essex	12	Yes
Mr B W Ready, Chemistry Dept, University of Essex, Colchester, Essex	6	H + A

SURREY AND MIDDLESEX

Mr M Bawtree, Royal Grammar School, High street, Guildford Surrey	5	Yes
Dr R Hancock, St Peters & Merrow Grange Comp School, Horseshoe Lane, Guildford, Surrey	15	-
Mr D R Marlow, Yateley School, School Lane, Yateley, Camberley, Surrey	6	
Mrs McKen, Dept Hotel & Catering, Guildford Co College of Technology Guildford, Surrey	6	Yes
Mr W A Smith, Redhill Technical College, Gatton Point, Redhill, Surrey RH1 2JX	16	Yes
Mr D S Williams, Dept Biochemistry, Brunel University, Uxbridge, Middlesex		
Mr J C Alderton, Eversham High School, Four Pools Road, Eversham. Surrey	2	Internal
Mr R Bird, King Edward's School, Witley, Godalming, Surrey GU8 5GG	3	H + A
Dr K W Glasson, 11 Lancaser Cottages, Lancaster Park, Richmond, Surrey TW10 6AE	1	H + A
Mr S J Ball, Dept of Technology, Brooklands TC, Heath Road, Weybridge, Surrey	1 +	Yes

KENT, SUSSEX AND HANTS

Mr P J Banbury, Sir William Nottidge School, Bellvue Road, Whitstable, Kent	1	Yes
Mrs T Bloodworth, Gosport, Fareham & District Teacher's Centre, 1 Spring Garden Lane, Gosport, Hants, PO12 1HY		
Mrs S M Douse, Hillside School, Marsden Road, Paulsgrove, Portsmouth	1	No
Mr G M Earney, Oak Farm School, Chaucer Road, Farnborough, Hants	1	Yes
Mrs S J Gray, Heron Wood Comprehensive School, Tongham Road, Aldershot, Hants	1	
S W Hampshire Teacher's Centre, Highwood Road, Brockenhurst, Hants		Yes
Mrs S Kinahan, St. George RC School, Leaside Way, Swaythling, Southampton	1	Staff Only
Mr D C Morton, N E Hants Teachers Centre, Lynchford Road, Aldershot Gull 2HH	1	YES
Mr P R Neate, Curriculum Dev Centre, Grosvenor Road, Portswood, Southampton SO2 3RR	7	Yes
Mr R M Homer, Farnborough Coll of Tech, Boundary Road, Farnborough, Hants GU14 6SB		
Bohunt School, Longmoor Road, Liphook, Hants	4	Internal
Mr J F Wright, Chineham Park Jun School, Shakespeare Rd, Basingstoke, Hants	2	Yes
Mr G Anderston, Itchen Cottage, Middle Road, Southampton, S09 3AX	5	H + A
Mr J Ralston, Computer Science, Queen Mary's College Cliddesden Road, Basingstoke, Hants	5	Yes
Mr J Rosbottom, South Downs College Road, Havant, Hants P07 8AA	10	Yes
Mr K J Shorey, Fernhill School, Neville Duke Road, Farnborough, Hants GU14 9BY		
Mr R Thornton, Portsmouth Grammar School, High Street, Old Portsmouth, Hants	2	
Mr A J Oldknow, W Sussex Inst of H E, The Dome, Upper Bognor Rd Bognor Regis, Sussex	9	Yes
Wing Com Ryan, 164 Chesterfield Drive, Sevenoaks, Kent	1	No
Mr J O'Donoghue, Stanley Deason High School, Wilson Avenue, Brighton BN2 5PB	3	Yes
Mr J Hewitt, Broomfield School, Middle Park Way Leigh Park, Havant, Hants PO9 4DA	2	Internal
Mr D L Bothwell, Education Dept, Hampshire CC, The Castle, Winchester Hants SO23 8UG	100+	Yes
Mr A French, Crawley Teachers Centre, Thomas Bennett School, Ashdown Drive, Tilgate, Crawley, Sussex RH10 5AD	1+	Yes
Mr Wellard, Maths Dept, Brighton Poly, Watts Building, Lewes Rd, Brighton, Sussex	7	H + A
Mr G L Holmes, Lewes Technical College, Mountfield Rd, Lewes, E Sussex BN7 2XH	10	Yes
Mr A Clark, Computer Studies Unit, Chichester College of Technology, Westgate Fields, Chichester, West Sussex	12	Yes
Mr J Sansom, Computer Centre, Brighton Poly, Moulsecomb, Brighton 1BN2 4GL	50+	Yes
Mr M D Meredith, Dept of Education, The University of Southampton, Southampton S09 5NH	4	Yes

BERKS, BUCKS AND OXFORDSHIRE

Mr M Flinn, School of Management Studies, Newland Park College, Gorelands Lane, Chalfont St Giles, Bucks	8	Yes
Mrs P V Wright, John Mason School, Wootton Road, Abindon OX14 1JB	3	Internal
Mr A Smith, Wallingford School, Station Road, Wallingford, Oxon	3	Yes
Mr R E Jones, N Oxon Technical College, Broughton Road, Banbury Oxon	3	Yes
Mr M Wells, Amersham College of F E, Art and Design, Stanley Hill, Amersham, Bucks	1	Yes
Mr B Jones, Slough College of HE, Wellington Street, Slough Berkshire	8	Yes
Mr T L Collings, Denbigh School, Cornwall Grove, Bletchley, Milton Keynes	1+6 Vics	H + A
Dr R F Cooper, Dept of Applied Sciences, Langley College of FE, Station Road, Langley, Slough, Bucks	24	Yes

HERTS AND BEDS

Mr J A Crombie, Watford College, Hampstead Road Watford, Herts	12	Yes
Mr G Hubbard, Park Wood School, Hawk Drive, Bedford	1	Yes
Mr. R A Wood, Stockwood High School, Cutenhoe School, Luton, Beds	1	Internal
Mrs E Ramsden, Dacorum College, Marlowes, Hemel Hempstead, Herts	6	Yes
Mr R Leggart, Bedford College, Piehill Avenue, Bedford	2	H + A

EAST ANGLIA

Mr I Birnbaum, Ramsey Abbey School, Ramsey, Huntingdon, Cambs PE17 1OH	2	Yes
Mr A Harrison, Manor School, Arbury Road, Cambridge	1	
Mr W R Howell, The Grammar School, Salsbury Road, Gt Yarmouth, Norfolk	1	Yes
Mrs S Tutt, Coleridge Community Centre, Reagund Road, Cambridge CB1 3RJ	2	For adults
Mr P Tyler, Springwood High School, Queensway King's Lynn PE30 4AW	4	Yes
Messers Luton & Walton, Houghton County Primary School, Houghton, Huntingdon, Cambs PE17 2AY	2	Yes
Mr R A MacDougall, 37 Northgate Way, Terrington St Clement, Kings Lynn	3	Yes
Mr J R Bailey, Costessey High School, Richmond Rd, New Costessey, Norwich	7	Internal
MR M V Smith, Stalham High School, Brumstead Road, Stalham Norwich	2	Yes
Mr A Goodhew, Sir Harry Smith Community College, Eastree Road, Whittlesey, Peterborough PE7 1XB	1	H + A
Mr J F White, West Suffolk College of FE, Out Risbygate, Bury St Edmunds, Suffolk	10	Yes

NORTHANTS, LEICS, NOTTS AN LINCS

Ms M Holloway, Beanfield Infant School, Farmstead Road, Corby, Northants	1	
Dr G A Thompson, Wyggeston & Queen Elizabeth I College, University Road, Leicester	2	Yes
Mr S C Tompkin, Croft Primary School, Station Road, Sutton in Ashfield, Nottinghamshire	2	
Dr D J Yates, Univ of Nottingham, Sutton Bonington, Loughborough LE12 5RD	3	Internal
Mr P T Aldridge, Bishop Stortford School, The Headlands, Kettering, Northants	4	Internal
Mr R Broadley, Dept of Social Studies, Trent Poly, Burton Street, Nottingham	4	Internal
Mr R Broadley, Dept of Social Studies, Trent Poly, Burton Street, Nottingham	15	Yes
Mr C R Du Feu, Queen Elizabeth's Grammar School, Morton terrace, Gainsborough, Lincs DN21 2ST	1	Yes

WEST MIDLANDS AND STAFFORDSHIRE

Mr R Coxon, 7 Chadbury Road, Halesowen, West Midlands B63 3HB	10	Help and advice
Mr D Dixon, Arden School, Station Road, Knowle, Solihull, West Midlands	1	
Mr A Fawdon, Production Engineering Department, Coventry (Lanchester) Polytechnic, Coventry	6	
Mr A Henry, Garretts Green Technical College, Garretts Green Lane, Sheldon, Birmingham	1	No
Mr R M Homer, W Bromwich College, Management Dept, Wood Green Road, Wednesday, West Midlands	5	
Mrs D Hughes, Holy Child Convent, 39 Sir Harry Road, Edgbaston, Birmingham B15	1	NO
Mrs E Moody, Maths/Computing Department, Westhill College, Weoley Park Road, Birmingham B29 6LL	1	Yes
Mr R Pountney, Sharmans Cross Comp School, Radbourne Road, Shirley, Solihull	4	Internal

Computer Dept, Richmond High School for Boys, Richmond Street, Halesowen, West Midlands	1	No
Mr J T Shaw, Archbishop Williams School, East Medway, Tile Cross, Birmingham B33 OAR	1	Willing
Mr M Smith, Charlemont Teachers Centre, Hollyhedge Road, West Bromwich B71 3DL	4	Possibly
Mrs M Duffy, Oaklands, Smeaton Lane, Stretton-Under-Fosse, Nr Rugby CV23 OPS	1	H + A
Dr E F Slade, Physics Dept, University of Keele, Keele, Staffs ST5 5BS	5	No
Mrs L Watts, Sans Souci, Teachers Centre, Tanworth Lane, Shirely, Solihull	1	Yes
Mr D H Readman, Stoke-on-Trent Tech College, Moorland Road, Burslem, Stoke-on-Trent, Staffs	6	Yes
Mr A Wood, 2 St James Road, Dudley, West Midlands	6	Yes
Mr R A Clarke, Computer Centre, Birmingham Poly, Franchise Street, Birmingham B42 2SU	4	Yes
Mr J Aspinall, West Midlands College of HE, Gorway Walsall	12	Yes
Mr M N Cooke, Henley College of FE, Macro Unit, Henley Road, Bell Green, Coventry CV2 1ED	2	Yes
Mr K A Farthing, Business & General Studies Dept, Tile Hill College of FE, Tile Hill Lane, Coventry	11	Yes
S A L O P		
Mr K J Delaney, Adams Grammar School, Newport, Shropshire	2	Not Yet
Mr T Lusty, The Upper School Stirchley, Telford, Salop	3	
Mr R Duncan, Meole Brace Modern School, Longden Road, Shrewsbury	2	Internal
Mr J Burn, The Manor School, Hadley, Telford, Shropshire	1	Yes
Mr J I Meardon, Adams School, Lowe Hill, Wem, Shropshire, SYS 5UB	2	Internal
GREATER MANCHESTER		
Mr S Allanson, St Albans School, Warren Lane, Oldham, Greater Manchester	1	No
Mr R Bates, Lancasterian School, Elizabeth Slinger Road, Manchester M20 8XA	1	No
Dr P Birkett, Science Department, Bolton Insititute of Technology, Deane Road, Bolton, Greater Manchester	4	Yes
Messrs D Brown/A Roscoe, Rose Bridge School, Holt Street, Ince, Wigan, Greater Manchester	1	Yes
Mrs J Fiddies, The Deane High School, New York, Junction Road, Bolton BL3 4NG, Greater Manchester	1	
Mr A Goodall, Xaverian College, Lower Park Road, Manchester 14	3	Evens esp Wed
Mr J G Harris, Salford Resources Centre, Half Edge Lane, Eccles,	2	Yes
Miss A Howlett, Leight College, Marshall Street, Leigh, Greater Manchester	8	Yes
Mr M Jasper, Church of England High School, Northfleet Road, Peel Green, Eccles, Manchester	1	Internally
Mr R K Johnson, Science Inspector, Education Offices (1), Crown Square, Manchester M60 3BB	50	Many
Mr E Leonard, Loreto College, Chichester Road, Manchester M15 5PB	1	Yes
Mr P Murphy, Computer Resource Centre, Manchester Teachers Centre, Barlow Moor Road, Manchester	3	Yes
Mr B Phillips, St. Albans Catholic High School, Holmcroft Road, Gorton, Manchester 18	1	Yes
Mr B E Rant, Wythenshawe College, Moor Road, Manchester M23 9BQ	1	Yes
Mr T Robertshaw, Hope High School, Eccles Old Road, Salford M6 8BJ, Greater Manchester	1	
Mr B H Stead, Education Office, Gateway House, Standishgate, Wigan WN1 1XL, Greater Manchester	25	Yes
Mr D R Stretch, Chorlton High School, Corkland Road, Manchester M21 2XD	3	No
Mr G Beech, NCC, Manchester		
Mr E Purcell, Abraham Moss Centre, Crescent Road, Manchester M8 6HU		
Mr I G Fanning, Holy Cross College, Manchester Road, Burly, Greater Manchester	1	Yes
Mrs E M Heseltine, Leigh CE High School, Leigh Road, Leigh, Greater Manchester	2	Inservice to staff
Mr A M Woods, Mount St Joseph School, Hawthorne Road, Bolton, Greater Manchester	1	Yes
Messrs Leather & Oulton, Salford College of Technology, Frederick Road, Salford,		

Greater Manchester M6 6PU	4	Yes
Mrs S P Buckthorpe, Turton High School, Chapelton Road, Bromley Cross, Bolton, Greater Manchester	1	Yes
Mr P Mangall, Greenhill High School, Fallingford, Rochdale, Greater Manchester	3	Yes
Mr A McGowan, Matthew Moss School, Marland, Rochdale, Greater Manchester OL11 3LU	3	Internal
Mrs F Harris, Bolton Technical College, Manchester Rod, Bolton, Greater Mancheste	Many	Possibly

MERSEYSIDE

Mr G Catty, Formby High School, Freshfield Road, Formby, Liverpool	4	Adult Ed. Class
Mr R P Stevens, Knowsley Hey Comprehensive School, Seel Road, Huyton, Liverpool	1	No
Mr M Trotter, Microelectronics Advisor, 10 Hatchmere Close, Birkenhead Merseyside	1	Runs courses
Mr R E Marsh, Science Department, Anfield Comprehensive School, Breckside Park, Liverpool L5 4DN	1	Note Yet
Mr R Hogg, Liverpool Inst of H E, Christs & Notre Dame College, Woolton Rd, Liverpool	4	Yes
Mr Bailey, Dept of Education, Met Bor of Wirral, Municipal Offices, Cleveland St, Birkenhead, Merseyside L41 6NH	Many	Yes
Mr R D Seddon, Maths Dept, Liverpool Poly, Byrom St, Liverpool L3 3AF	4	Yes
Mr A G Waring, Mosscroft School, Bedford Close, Huyton, Liverpool L36 1XH	1	Yes
Mr J Stout, 6 College Avenue, Formby, Merseyside L37 3JJ		
Mr P R Hope, Ruffwood Comp School, Kirkby, Liverpool L33 8XF	2	Possibly
Mr B Langley, Southport Tech College, Mornington Road, Southport, Merseyside		
Mr B Wainman, Ormonde High School, Ormonde Drive, Maghull, Merseyside	3	Yes
Mr J R Park, Merchant Taylors' Liverpool Road, Crosby, Liverpool L24 OQP	9	Yes
Mr J Battisi, City of Liverpool, College of HE, Liverpool Road, Prescot, Merseyside L34 1NP	3	Yes

LANCASHIRE & CHESHIRE

Mr K S Clark, Palatine High School, St. Annes Road, Blackpool, Lancashire	1	Yes
Mr P Martin, St Nicholas RC School, Greenbank Lane, Hartford, Northwich, Cheshire	1	No
Miss M. Mills, Computer Services Department, Preston Polytechnic, Corporation Street, Preston, Lancs	21	Yes
Mr P D Sudlow, Arnold School, Blackpool, Lancashire	1	No
Mr G Thomas, Maths Department, Chester College, Cheyney Road, Chester CH1 4BJ	6	Yes
Mr T L Thurogood, Blackpool College, Electronics Department, Blackpool, Lancashire	15	Yes
Mr D Webb, Edge Hill College HE, St Helens Road, Ormskirk, Lancashire L39 4QP	10	
Mr A Potts, Sir John Deane's College, Northwich, Cheshire CW9 8AF	5	Yes
Mr G W Cockrell, Crewe & Alsage College, Crewe Road, Crewe CW1 1DU	14	Yes
Mrs J Guenault, S Martin's College, Lancaster LA1 3JD	4	Yes
Mr L M Davies, Ashton-On-Ribble High School, Aldwych Drive, Preston, Lancs	2	No
Mr S R Lloyd, Halto C of FE, Kingsway, Widnes, Cheshire	6	Yes
Mr K M Roberts, Nantwich Science Centre, Manor Road, Nantwich, Cheshire	3	Yes
Mr R Barham, School of Construction & Urban Studies, Preston Poly, Corporation Street, Preston	2	Yes
Mr E J Mullineux, Wellington Boys School, Moss Lane, Tiperley, Altrincham Cheshire	1	Internal
Mr R J Wilson, Baines School, Highcross Road, Poulton-Le Fylde, Blackpool, Lancashire	3	H + A
Mr S J Rogers, Darwen Vale School, Blackburn Road, Darwen, Lancashire	5	Internal

DERBY

Mr P Radcliffe, Buxton College, College Road, Buxton, Derbyshire	3	Internal
Mr A F Thoday, Earnest Bailey School, Matlock Derbyshire DE4 3FE	3	At local teacher centres

Mr L J Want, John Port School, Etwall, Derbyshire	5	No
Mr R Gale, Noel-Baker School, Bracknell Drive, Alvaston, Derbyshire	3	Internal
Mr P Dobson, Henry Fanshawe School, Dronfield, Derbys S18 6XD	1	Yes
Mr N W Rutherford, Derbyshire County Council, County Offices, Matlock, Derbyshire DE4 3AG	Many	Yes
Mr R A Parker, Bolsover School, Mooracre Lane, Bolsover, Chesterfield, Derbys	2	Yes
Ms L Blanchard, N E Derbyshire College of FE, Rectory Road, Clowne, Nr Chesterfield, Derbyshire	11	Yes
YORKSHIRE		
Mr R T Ashford, York College of Arts & Technology, York	8	Yes
Dr R M Campbell, Education Department, University of York, Heslington, York	2	Yes
Mr K R Tomlinson, Computing Centre, Bradford Schools, Britannia House, Hall Ings, Bradford BD1 1HX	55	Yes
Mr B Davies, Whitby School, Prospect Hill, Whitby, N Yorkshire	2	Informal
Mr J Dewhirst, N Farm Middle School, Thorn Walk, Leeds 8, West Yorkshire	1	Yes
Mr D W Ellingham, Marcliffe Jun & Infant School, Marcliffe Road, Sheffield S6 4AJ	1	Yes
Mr D Greatorex, College of Ripon and York St John, Lord Mayors Walk, York	2	Yes
Mr R Irvine Smith, Wentworth College, University of York, Hesligton, York	1	Yes
Mr D Jinkinson, Sheffield Region Centre for Science/Technology, 55 Broomgrove Road, Sheffield	1	Yes
Mr P A Riding, Computer Unit, Leeds Polytechnic, Calverley Street, Leeds LS1 3HE	1	Yes
Mrs J Roberts, The Park High School, Park Lane, Pontefract, W Yorkshire	1	No
Mr P D Robinson, South Grove Comprehensive School, Morrgate, Rotherham, S Yorkshire	1	Internal
Mr B S Wilson, Leeds Poly, School of Education, Beckett Park, Leeds LS6 3QS	1	Yes
Mr P J Auckland, Stocksbridge College, Hole House Lane, Stocksbridge Sheffield S30 5BN	6	Yes
Computer Unit, Bradford College, Old Building, Great Horton Road, Bradford BD7 1AY	7	Yes
HUMBERSIDE		
Messrs Thomas & Moore, Univ of Hull, 173 Cottingham Road, Hull HU5 2EH	1	Yes
Mr P Thompson, Schoolsoft, 67 Anderby Drive, Grimsby, S Humberside DN37 9ET	3	Internal
CUMBRIA		
Mr A Sutcliffe, Charlotte Mason College, Ambleside, Cumbria LA22 9BB	3	Yes
Mr A L Watters, Business/Mnagement, West Cumbria College, Park Lane, Workington, Cumbria	12	Yes
Mr A Thompson, Moorclose School, Needham Drive, Workington CA14 3SE	2	Yes
MR P Foulkes, Netherhall School, Maryport, Cumbria	3	Yes
Mr C R Dean, Parkview School, West Avenue, Barrow-in-Furness, Cumbria	5	H + A
HEREFORD & WORCESTER		
Mr J F Foster, Microproject, Country Resources Centre, Oakfield, Claines, Worcester	1	Yes
Mr D C F Chaundy, Malvern College, Malvern Worcs WR14 3DF	13	Yes
DURHAM AND TYNE & WEAR		
Mr W G C Austin, Geography Department, Newcastle Polytechnic, Sandyford Road, Newcastle Upon Tyne	30	Yes
Mr J M Cocalis, Wolsingham Comprehensive School, Wolsingham, Bishop Auckland, Durham	1	No
Mr B Jefferson, Branksome Comprehensive School, Eggleston View, Darlington, Co Durham	3	Internal
Mr T Mead, St Aidans RC School, Willow Bank School, Ashbrooke, Sunderland SR2 7HJ		
Mr L A Willis, Inservice Training Unit, Faculty of Education, Sunderland Polytechnic, Sunderland	12	Yes
Mr D Wardill, 7 Ashtree Close, Rowlands Gill, Tyne & Wear NE39 1RA	Several	Yes
Mr B Nattres, Dept of Educ Studies, Teeside Polytechnic, Cleveland	16+	Yes

Mr R Reed, Microelectronics Educ Centre, Newcastle Upon Tyne Poly, Coach Lane, Benton, Newcastle Upon Tyne	3	Yes
Mr C Dixon, Walker Comp School, Middle St, Walker, Newcastle-Upon-Tyne	5	Yes
ASCENTS, North Tyneside Micro Education Centre, King Edward School, Preston Avenue, North Shields NE30 2BD	2	Yes
Mr W R Houston, Peterlee Technical College, Peterlee, Co Durham	22	Yes
Mr W J Best, Durham School, Durham City DH1 4S2	5	H + A

NORTHUMBERLAND

Mr A B Durnace, King Edward VI School, Cottingwood Lane, Morpeth, Northumberland	Many	Yes
--	------	-----

AVON AND GLOUCESTERSHIRE

Dr D Blane, College of St Paul & St Mary, The Park, Cheltenham, Glos	4	Yes
Mr F H Garlick Bristol Polytechnic, Cabot House, Askley Down, Bristol	24	Yes
Major T Hardie, Ladies' College, Cheltenham, Gloucestershire	Consider- ing	
8		
Mr N Larcombe, West Glos College of F E, College Road, Cinderford, Gloucestershire	8	Yes
Dr P N Richards, Bath University, School of Education, Claverton Down, Bath BA2 7AY	1	Yes
Dr R Sullivan, Physics Dept, Bath University, Clavertown Down, Bath, BA2 7AY	6	Yes
Dr A R Rogers, Dept of Physiology, The Medical School, University Walk, Bristol BS8 1TD	Many	Internal
Mr P White, Dept of Business Studies, Bristol Poly, Coldharbour Lane, Bristol Avon	9	Yes
Mr D A Hawkswell, Rendcomb College, Cirencester, Glos GL7 7E2	4	H + A

DORSET AND WILTSHIRE

Mr R C Cory, County Advisor, Eastern Area Ed Offices, Portman House, Richmond Hill, Bournemouth	12	Yes
Mr B Taylor, General Ed Dept, Chippenham Technical College, Coklebury Road, Chippenham, Wilts	4	Yes
Mr D Bale, Dorset Institute of HE, Wallisdown Road, Poole, Dorset BH12 5BB	5+3 Vics	Yes
Mr F D Pickles, Toft Computer Training Centre, 13 Central Drive, Meyrick Park, Bournemouth BH2 6LQ	3	Internal
Mrs A Straker, Education Dept, County Hall, Trowbridge Wilts	2	Yes
Mr K H Dyson, Gillingham School, Gillingham, Dorset SP7 8DW	3	Yes
Mr L Ruddick, Sheldon School, Chippenham, Wilts	2	Yes
Mr G A Wickham, Dorset Institute of H E, Cranford Avenue, Weymouth, Dorset	2+2 VICs	Yes

WALES

Mr L A Dawson, Rhonda College of Further Education, Llwynypia, Mid Glamorgan, Wales	6	
Mr J M Ferguson, Educational Psych., Ladywell House, Newton, Powys SW16 1JB	7	Yes
Mr B Jeff, Head of Maths, Llanidloes High School, Llanidloes, Powys SY18 6EX	2	Yes
Mr B J Roberts, Trinity College, Camarthen, Dyfed	4	Yes
MNr J R Wilson, Maths Adviser, County Offices, Llandrindod Wells, Powys	6	Time to time
Mr M Bushell, Llanidloes High School, Llanidloes, Powys SY18 6EX	2	Yes
Dr D Wharry, Dept of Education, University College, Cardiff	3	Yes
Mr J R Fletcher, Atlantic College, St Donat's Castle, Llantwit Major, S Glam	3	Yes
Miss G M Williams, Gwent College of HE, Allt-Yr-Yn-Avenue, Newport, Gwent	8	Yes
Mr A E Owen, Pontypridd Technical College, Ynys Terrace, Rhydyfelin, Pontypridd, Mid-Glam	8	Yes
Mr W Jeffrey Rees, Amman Valley Comp School, Ammanford, Dyfed	3	Yes
Mr R C Bennett, Whitland Grammar School, North Road, Whitland, Dyfed	3	H + A
Mr J G Francis, Computer Centre, Poly of Wales, Llantivit Road, Treforest, Pontypridd Mid-Glam CF37 1DL Wales		

DEVON & CORNWALL

Mr R Randall, Newquay Tretherras School, Trevenson Road, Newquay Cornwall	2	Internal
Prof D Burghess, Dept of Education, Exeter University, Stocker Road, Exeter EX4 4QD		

SOMERSET

Mr P J Fry, Millfield School, Street, Somerset	5	Internal
Mr P Johnson, Wells Cathedral School, Wells, Somerset BA5 3SX	6	Internal

SCOTLAND

Mr J Barton, Stevensons College, Computer Department, Edinburgh	4	Yes
Mr J L Bowie, Adviser Business Studies, Floor 8, Tayside House, Crichton Street, Dundee	1	Not Yet
Dr I Bushell, Department of Psychology, University of Glasgow, Glasgow	2	Informal
Mr C W Crooks, Newton Centre, Green St Lane, Ayr	5	Yes
Mr P R Cutforth, George Watson's College, Colington Road, Edinburgh EH10 5EG	3	No
Mr J R S Davidson, Maths Department, Stow Cottage, 43 Shamrock Street, Glasgow G4 9LD	9	Not Yet
Mr R D Instrell, Forrester High School, Broomhouse Road, Edinburgh EH12 9AE	1	Yes
Mr D McGill, Biological Science, Stow College, 43 Shamrock Street, Glasgow G4 9LD	10	
Mr M J Prowse, Adviser - Sec Education, Education Offices, 30 Edinburgh Road, Dumfries	11	Yes
Mr I Robertson, Science Department, St James Annexe, Reid Kerr College, Paisley	6	Possibly
Mr W Sarson, Douglas Ewart High School, Newton Stewart, Wigtownshire DG8 6HT	3	Yes
Mr R M Tindall, Floor & Tayside House, 28 Crichton Street, Dundee	3	Yes
Mr R Wills, Dundee College of Education, Gardyne Road, Broughty Ferry, Dundee	17	Yes
Mr A B Wilson, Gourrock High School, Fletcher Avenue, Gourrock, Renfrewshire	1	Internal
Mr D Muir, Napier College, Colinton Rd, Edinburgh EH10 5DT	3	Yes
Dr J Chalmers, Queen's College, 1 Park Drive, Glasgow	6	Yes
Mr A H Livingstone, Williamwood High School, Seres Rd, Clarkston, Glasgow	2	H + A
Mr W Patterson, Paisley College of Technology, High St, Paisley PA1 2BE	6	Yes
Mr A Redpath, Anniesland College, 19 Hatfield Drive, Glasgow G12 DYE	3	No
Mr E A Martin, Dunfermline College of PE, Cramond Rd North, Edinburgh	2	Possibly

ISLE OF WIGHT

Mr R E Smith, Mayfield CE Middle School, St Vincents' Road, Ryde, Isle of Wight	1	No
---	---	----

ISLE OF MAN

Mr N Roberts, Douglas High School, St Ninians, Douglas, Isle of Man		
---	--	--

IRELAND

Mr J P MacDougal, Limerick Technical College, Athenaeum Building, Limerick, Eire	10	Soon
Mr D McCarthy, College of Marketing & Design, 18 Parnell Sq, Dublin 1	13	Yes
Dr J P Chester, Dept of Applied Science, Moylish Park, Limerick, Ireland	20	H + A
Mr W McSparron, Faughan Valley, Sec School Drumahoe, Londonderry	2	H + A

CHANNEL ISLANDS

Mrs L Lucas, La Mare de Carteret Secondary School, Rue de Galaad, Castel, Guernsey	3	Yes
Miss B Sheen, Jersey College of Girls, La Pouquelaye, St Helier, Jersey	1	Yes
Mr C R Miller, St Joseph's School, SPP, Guernsey	2	Yes

OVERSEAS

Mr J Smilde, Copytronics, Burg v Suchtelenstraat 46, 7413 XP Deventer, Netherlands	1	H + A
Mr V Klafki, 39 AM Forsthaus, D-6078 Neu-Isenburg 2, West Germany	3	Yes
Mr J W Greenacre, Peterhouse, Private Bag 3741, Maraandellas, Zimbabwe	12	Internal
Mr T G J de Both, ITP Computer Products SA, PO Box 424, 6500 AK Nijmegen, Holland		
Mr C Schlezinger, Danciger Comp School, Kiryat Shmona 10200, Israel	15	Yes
Mr P Malmberg, dataromics AB, Vretenborgsc 8, S-12612 Stockholm, Sweeden		
Mr D Voulgaridou, Vas Georgiou 16, Thessaloniki, Greece	5	Possibly
Mr C L Churms, Dept of Physics, University of Stellenbosch, Stellenbosch, South Africa	3	H + A

The original BASIC/M.C. listing is in Commodore Club News, August 1981.

The listing is for a BASIC/M.C. "loader" which when entered into the PET and RUN loads the running Assembler/Disassembler into the top of the BASIC RAM area, and prints to the screen the access command to this loaded machine code program. I.E. on a 32K machine SYS31283 (equivalent to \$7a33), leaving control in BASIC.

By entering the monitor by SYS31283 the "loaded" machine code program can be saved as follows :-

```
.s"sup/mon4.bin",01,7a33,8000
```

but cannot be verified through the monitor. Verifying, once saved, can be done by exiting the monitor to BASIC by .x (RETURN), rewinding the tape to the beginning of the program, and typing :-

verify (RETURN)
With both "loader" and "loaded" versions type 'new' to reset pointers etc. before entering any basic program writing exercises.

When using the M.C./BASIC "loader" as listed, PET RAM locations 48, 49 and 52, 53 are POKed by the basic "loader" program. When using the "loaded" machine code program only locations 52 and 53 to protect Supermon 4 from BASIC. Also, the "loaded" machine code program, when loaded to the PET from tape, loses the value stored at the last location used i.e. 7fff. This value, which for the 32K machine is 7a, is the high byte of the address to be stored in location 53 on entry with SYS31283. The value after loading from tape remained as at power-up i.e. \$aa, and hence far too much of the RAM is protected from BASIC.

To overcome these two minor faults, the M.C./BASIC "loader" version was modified to :-

- Store the two bytes of the address to be stored in location 0680.
- Add two extra STA instructions at the beginning of the "loaded" part of the machine code program to store the two bytes of the address at locations 48, 49.

The modifications to the listing are as in figure 1.

Figure 1.

```
a) Machine Code program :-
..: 0678 aa 2d fa 00 ad 2d fa 00
..: 0680 85 30 85 34 ad 2e fa 00
..: 0688 85 31 85 35 ad fc ff 00
..: 0690 a9 79 c5 1f a9 06 e5 20
```

Compare this to the original program, and you'll find the bytes that need changing.

b) BASIC program
Change line 180 from :-
180 poke52,m:poke53,n:poke48,m:
poke49,n:n=m+n*256
to :-

```
180 poke52,m:poke53,n:poke48,m:  
poke49,n:n=m+n*256+2
```

The "loaded" version of the machine code program is named "SUP/MON#2.BIN" in the tape (or disk) directory, and is accessed by SYS31279 for all functions except the "print disassembler".

"Print disassembler" requires that the printer is engaged before entering the monitor. This is done by entering the direct command :-
open4,4:cmd4

Then enter the monitor (again as a direct command) by :-
sys 54935
on the same line i.e.
open4,4:cmd4:sys 54935

The final alteration we have to make is to the program giving the instructions on how to operate Supermon 4. This was not published in the August issue of the newsletter, but has been published earlier when we gave you the listing for Supermon 2. So, for those of you with the instructions program, lines 1780 and 1785 need changing to :-

```
1780 print"and access the monitor  
via a call :-"  
1785 print" [ rvs] sys 54935[off],  
after engaging printer"
```

After making the above changes, your Supermon program is completely up to date!

Finding the mean of a 1000 element array

Michael J. Smyth

MEANOFARRAY.....PAGE 0001

LINE# LOC CODE LINE

```
0001 0000                               **=$033A
0002 033A                               POINT1=$01           ;LO,HI POINTER TO ARRAY ELEMENTS
0003 033A                               POINT2=$02
0004 033A                               COUNT1=$BA          ;LO,HI COUNTER FOR SUMMATION
0005 033A                               COUNT2=$BB
0006 033A                               STEP=$10             ;POINTER STEP
0007 033A                               MLO=$BC
0008 033A                               MHI=$BD             ;LO,HI ADDRESS OF START-OF-M

0010 033A 78                               SEI
0011 033B A6 77                               LDX $77             ;DUMP CHARACTER POINTER
0012 033D 86 BA                               STX COUNT1          ;TEMPORARILY IN COUNTER
0013 033F A6 78                               LDX $78
0014 0341 86 BB                               STX COUNT2
0015 0343 A2 F0                               LDX #<NAME          ;LO BYTE OF ADDRESS OF 'M'
0016 0345 86 77                               STX $77
0017 0347 A2 03                               LDX #>NAME          ;HI BYTE OF ADDRESS OF 'M'
0018 0349 86 78                               STX $78
0019 034B 20 6D CF                               JSR $CF6D           ;FIND M IN MEMORY
0020 034E 18                               CLC
0021 034F A5 5C                               LDA $5C
0022 0351 69 02                               ADC #02             ;STEP 2 TO START-OF-M
0023 0353 85 BC                               STA MLO
0024 0355 A5 5D                               LDA $5D
0025 0357 69 00                               ADC #00
0026 0359 85 BD                               STA MHI
0027 035B A6 BA                               LDX COUNT1          ;RESTORE CHARACTER POINTER
0028 035D 86 77                               STX $77
0029 035F A6 BB                               LDX COUNT2
0030 0361 86 78                               STX $78
```

```

0031 0363 20 DB 03      JSR COUNT      ;GET NO. OF ELEMENTS IN ARRAY
0032 0366 A2 07      LDX #07      ;AND MOVE POINTER TO FIRST ELEMENT
0033 0368 86 10      STX STEP
0034 036A 20 BF 03      JSR INCR
0035 036D A2 05      LDX #05
0036 036F 86 10      STX STEP      ;SET STEP = 5
0037 0371 A5 01      AGAIN LDA POINT1
0038 0373 A4 02      LDY POINT2
0039 0375 20 AE DA      JSR $DAAE      ;LOAD FAC#1 FROM ARRAY
0040 0378 A5 BC      LDA MLO
0041 037A A4 BD      LDY MHI
0042 037C 20 73 D7      JSR $D773      ;ADD SUM TO FAC#1
0043 037F A6 BC      LDX MLO
0044 0381 86 46      STX $46
0045 0383 A6 BD      LDX MHI
0046 0385 86 47      STX $47
0047 0387 20 DC DA      JSR $DADC      ;STORE FAC#1 INTO M
0048 038A 20 CD 03      JSR DECR      ;DECREMENT COUNTER
0049 038D A5 BA      LDA COUNT1     ;AND BRANCH TO NEXT
0050 038F D0 07      BNE NEXT      ;IF NON-ZERO
0051 0391 A5 BB      LDA COUNT2
0052 0393 D0 03      BNE NEXT
0053 0395 4C 9E 03      JMP LAST
0054 0398 20 BF 03      NEXT JSR INCR    ;INCREMENT POINTER
0055 039B 4C 71 03      JMP AGAIN
0057 039E 20 DB 03      LAST JSR COUNT   ;GET NO. OF ELEMENTS IN ARRAY
0058 03A1 A5 BB      LDA COUNT2
0059 03A3 A4 BA      LDY COUNT1
0060 03A5 20 6D D2      JSR $D26D     ;INTEGER TO FLOATING POINT IN FAC#1
0061 03A8 A5 BC      LDA MLO
0062 03AA A4 BD      LDY MHI
0063 03AC 20 98 D9      JSR $D998     ;LOAD FAC#2 WITH SUM
0064 03AF 20 1E DA      JSR $DA1E     ;DIVIDE FAC#2 BY FAC#1
0065 03B2 A6 BC      LDX MLO
0066 03B4 86 46      STX $46
0067 03B6 A6 BD      LDX MHI
0068 03B8 86 47      STX $47
0069 03BA 20 DC DA      JSR $DADC     ;STORE RESULT IN M
0070 03BD 58          CLI
0071 03BE 60          RTS      ;RETURN TO BASIC

0073 03BF 18          INCR CLC      ;SUBROUTINE TO
0074 03C0 A5 01      LDA POINT1    ;STEP POINTER
0075 03C2 65 10      ADC STEP
0076 03C4 85 01      STA POINT1
0077 03C6 A5 02      LDA POINT2
0078 03C8 69 00      ADC #00
0079 03CA 85 02      STA POINT2
0080 03CC 60          RTS

0082 03CD 38          DECR SEC      ;SUBROUTINE TO
0083 03CE A5 BA      LDA COUNT1    ;DECREMENT COUNTER
0084 03D0 E9 01      SBC #01
0085 03D2 85 BA      STA COUNT1
0086 03D4 A5 BB      LDA COUNT2
0087 03D6 E9 00      SBC #00
0088 03D8 85 BB      STA COUNT2
0089 03DA 60          RTS

0091 03DB A6 2C      COUNT LDX $2C    ;SUBROUTINE TO
0092 03DD 86 01      STX POINT1    ;LOAD COUNTER WITH NUMBER
0093 03DF A6 2D      LDX $2D      ;OF ELEMENTS IN ARRAY
0094 03E1 86 02      STX POINT2
0095 03E3 A0 05      LDY #05
0096 03E5 B1 01      LDA (POINT1),Y
0097 03E7 85 BB      STA COUNT2
0098 03E9 A0 06      LDY #06
0099 03EB B1 01      LDA (POINT1),Y
0100 03ED 85 BA      STA COUNT1
0101 03EF 60          RTS

0103 03F0 4D          NAME .BYTE 'M'
0104 03F1 00          BRK
0105 03F2          .END

```

ERRORS = 0000

Let me start with an apology for the lack of an article last month. This is because that hard working tyrant Rod Wellburn, affectionately known as the Technical Services Manager, actually made me do some work. However, the Christmas spirit must have got to him this month, so here I am again, talking about RELative files this time.

RELative files are another way of handling Direct Access, and are only available on the Commodore 4040 and 8050 disk drives. (Last minute insert : RELative files are also available on the new 2031 disk drive, the 8422 22 megabyte hard disk, and also on the VIC disk drive. Interestingly enough, that latter drive is 4040 Read/Write compatible!)

As mentioned in an earlier article, it is possible to upgrade a 3040 to a 4040 and thus get RELative files. If you do upgrade your disk unit it is recommended that you also upgrade your PET to Basic 4.0. However, if you do upgrade your PET a lot of the programs you may already have, notably Invaders, will no longer work. DON'T PANIC !! It is possible to operate RELative files without the use of BASIC 4.0, although obviously it will be more difficult. Nonetheless it can be done, and I will be showing you how in a future article.

The example program which can be seen at the end of this article performs a similar function to the D-A example in the last article (Volume 3 Issue 9). This is so that you can compare the two programs.

Before we can start writing data into RELative file we must know two extra pieces of information. The first is the name of the RELative file ; unlike D-A files which access blocks on the disk REL files have a name which will appear in the directory. The second piece of information required is the length of the record ; this is the maximum length (number of characters) that a single record can hold (each record occupies the same space on the disk). The record length can be anything from 1 to 254 characters. The Disk Operating System will automatically make the best use of disk space (as it does for PRoGram and SEQuential files) ; the

only waste space is space which you define and don't use, so it makes sense NOT to define a record length of 254 characters if you are only going to have a maximum of 20 characters in a record.

Before we examine the program in detail, let us look at the commands used. Note : Square brackets indicate that a section of the command is [optional!]

DOPEN

Syntax : DOPEN#(NF),(NF\$) [, D(DR)] [, L(LN)] [, W] N.B. No spaces!

This is used to open a file on the disk unit.

NF is the Logical File Number
NF\$ is the name of the file, and can also be a literal string.

[, D(DR)] is used to specify the drive number (DR) of the file. If this is omitted, drive 0 is assumed.

[, L(LN)] is used to specify the length of the records within a file (LN). If this is omitted the file must already exist, or be a SEQuential write file.

[, W] is only used to create (write) a SEQuential file. When omitted the file will only be opened for READ or as a RELative file.

e . g .

DOPEN#(NF),(NF\$),D(DR),L(LN)
As in all DOPEN# commands, there must be NO SPACES left.

This opens a RELative file called (NF\$) on drive (DR) with a length of (LN) characters.

DOPEN#(NF),NF\$,W

This opens a SEQuential file called (NF\$) on drive 0 for writing.

DOPEN#(NF),(NF\$)

This opens any type of file on drive 0 called (NF\$) for reading. If the file is RELative it may also be written to, provided that it already existed.

NOTE : With these and the other disk commands any variables must be in brackets e.g. (variable)

NOTE : If a RELative file is opened with a length (LN) when it is first created, subsequent references to that file need not contain the L(LN) section in the DOPEN. If they do LN must be the same otherwise the error message:

50,RECORD NOT
PRESENT,00,00
will be generated.

RECORD

Syntax : RECORD
#(NF),(RC) [,(PT)] N.B.
No spaces!!

This is used to position the file pointer to the desired record, and position within that record.

RC is the record number.

PT is the position within the record at which reading or writing is to start (usually position 1). If this is omitted the default value of PT is 1.

NOTE : I always tend to use RECORD#(NF),(RC),1 if I want the first character on the record. This is not necessary, but does help to remind me that I have the option to go somewhere other than the first character.

DCLOSE

Syntax : DCLOSE#(NF)

This is used to close a file after it is finished with.

SCRATCH

Syntax : SCRAT
-CH(NF\$) [, D(DR)]

This is used to erase the specified file. If the D(DR) is omitted an attempt will be made to search the file from drive 0

NOTE : When used in direct mode (i.e. typed in and executed immediately) the command causes an 'ARE YOU SURE ?' question prompt. This question is not asked if performed under program control.

If you look at the listing 'REL-FILE EXAMPLE' at the end of this article and compare it with 'D-A EXAMPLE PROG' in Volume 3 Issue 9, you will notice various similarities. The major difference (apart from the use of REL file commands) is the fact that the index array I\$() is now only a single dimensional array (we can use the element number to point to a record). The program also has checks in it to ensure that file information (file name and record length) has been entered before data can itself be entered.

Now for an explanation of how the program does what it does.

LINE ====	EXPLANATION =====
2005	Assign drive number and file number. If the file name is null ("", no file has been assigned) Jump to 2300.
2080	Check that the compiled string is not longer than the record length. If it is, data must be re-entered.
2110	Open RELative file.
2120	Position record pointer to position 1 in the correct record (array elements are numbered 0 upwards, records 1 downwards)
2130	Write data to the disk.
2140	Close file.
2170-2180	Add key to index, increment index pointer.
2300-2380	No file data has been assigned. Get file name and record length.
2390	There are F+1 fields of information, each of which must contain at least one character, plus a RETURN. Thus the minimum record length must be (F+1)*2.
2400-2450	Creates the RELative file with MR (maximum number of records) records with one spare record. Maximum number = MR + 1 I will explain the reason for this in a subsequent article. (Next time perhaps).
3110	Open RELative file.
3120	Position record pointer ready for read.
3150	Read F+1 fields of information.
3160	Close the RELative file.
4040	Open the SEQuential file for the index. This is a sequential WRITE file. The name of the RELative file is used with the suffix '.IND'
4050	Check error message. If the file already existed, scratch it and re-start the Index Write option.
4060	Write the record length and the number of records used to the Index file, separated by a carriage RETURN.
4080	Write the index array (key names) to the disk.
4090	Close the file.
5030	Open the SEQuential Index file for reading.
5040	If the file does not exist, ask for the file name again.
5050	Read the record length and number of records used from the file.
5070	Read the Index array from the file.
5090	Close the file.
7000-7360	From the above information you should be able to work out what each line of this routine does without any trouble.
10005	DS contains the Error or Message number from the disk. It is transferred to EN to preserve it after other disk operations.
10020	An error of 50 means 'RECORD NOT PRESENT' and is only fatal if followed by an INPUT#. PRINT# will create the record (and any needed up to that record) and then write the data to that record. (Lines 2400 - 2450).
10030	63 - FILE EXISTS. 62 - FILE NOT FOUND.
10040	52 - FILE TOO LARGE. 72 - DISK FULL. This means there is no more room in the relative file, or on the disk. (See next month for more details).
10110 and 10310	LEFT\$(MID\$(DS\$,5),LEN(DS\$-10) prints out the message part of the disk error message (DS\$).

```

READY.
2200 REM INDEX FULL
2210 PRINT "INDEX IS FULL"
2220 FORA=0 TO 2000: NEXT: RETURN

READY.
2300 REM NO FILE DATA SET UP
2310 PRINT "YOU HAVE NOT SET UP ANY FILE INFORMATION"
2320 PRINT "YOU WANT TO CREATE A NEW FILE OR DO"
2330 PRINT "YOU WANT TO READ AN INDEX FROM DISK"
2340 GET A$: IF A$ <> "N" AND A$ <> "R" THEN 2340
2350 PRINT "J": IF A$ = "R" THEN 5000
2360 INPUT "RELATIVE NAME OF RELATIVE FILE": NF$
2370 IF LEN(NF$) > 12 THEN PRINT "MAX LENGTH 12 CHARACTERS": GOTO 2360
2380 INPUT "WHAT IS THE LENGTH OF EACH RECORD": LN
2390 IFLN < (F+1) * 2 OR LN > 254 THEN PRINT "RECORD LENGTH": (F+1) * 2, "- 254": GOTO 2380
2400 DOPE#(NF), (NF$), D<DR>, L<LN>
2420 RECORD#(NF), (MR+2), 1: REM CREATE MR+2 BLANK RECORDS
2430 PRINT #NF, "*"
2440 DCLOSE#(NF)
2450 RETURN

READY.
3000 REM READ A REL RECORD
3010 REM GET KEY AND SEARCH
3020 IFX=0 THEN 4100
3030 PRINT "READ A RECORD FROM "; NF$: R=-1
3040 INPUT "KEY (FIELD 0) ": KY$
3050 FORA=0 TO X-1: IF KY$ = I$(A) THEN R=A: A=X
3060 NEXT A: IF R < 0 THEN 3200
3070 RC=R+1: REM GET RECORD NUMBER

READY.
3100 REM READ AND DISPLAY RECORD
3110 DR=1: NF=3: DOPE#(NF), (NF$), D<DR>
3120 RECORD#(NF), (RC), 1
3140 FOR A=0 TO F
3150 INPUT #NF, F$(A): PRINT "FIELD"; A; F$(A)
3160 NEXT A: DCLOSE#(NF)
3170 GOSUB 12000: RETURN

READY.
3200 REM KEY NOT FOUND
3210 PRINT "NO KEY NOT IN INDEX"
3220 FORA=0 TO 2000: NEXT: RETURN

READY.
4000 REM WRITE INDEX TO DISK
4010 IFX=0 THEN 4100
4030 PRINT "WRITING INDEX TO DISK FOR "; NF$
4040 DR=0: NF=4: DOPE#(NF), (NF$), IND", D<DR>, W
4050 GOSUB 10000: IF EN<> 0 THEN SCRATCH(NF$, IND), 0: GOTO 4000
4060 PRINT #NF, LN; CR$: X; CR$:
4070 FORA=0 TO X-1
4080 PRINT #NF, I$(A); CR$:
4090 NEXT A: DCLOSE#(NF): GOSUB 10000: RETURN

READY.
4100 REM NO INDEX

```

```

10 REM REL FILE EXAMPLE PROGRAM
20 REM BY DAVID J. COCK
100 REM DR = DRIVE NUMBER
110 REM NF = FILE NUMBER
120 REM I$ = INDEX ARRAY
130 REM LN = RECORD LENGTH
140 REM X, RC = RECORD POINTER
150 REM F = NUMBER OF FIELDS (ADD 1)
160 REM D$ = CURSOR DOMAINS
170 REM MR = MAX NUMBER OF RECORDS
200 REM SETUP ARRAY AND START VARIABLES
210 DR=1: NF=2: X=0: F=3: CR$=CHR$(13): MR=20
220 D$=" "
230 DIM I$(MR): REM MAX MR+1 RECORDS IN AN INDEX
240 DIM F$(F): REM MAX F FIELDS PER RECORD

READY.
1000 REM MENU
1010 PRINT "RELATIVE FILE EXAMPLE"
1020 PRINT "1. ADD A RECORD"
1030 PRINT "2. LOOK AT A RECORD"
1040 PRINT "3. WRITE INDEX TO DISK"
1050 PRINT "4. READ INDEX FROM DISK"
1060 PRINT "5. LIST INDEX ON SCREEN"
1070 PRINT "6. AMEND A RECORD"
1080 PRINT "7. QUIT"

READY.
1100 REM SELECT OPTION
1110 GETA$: IF A$ = "" THEN 1110
1120 IF A$ < "1" OR A$ > "7" THEN 1110
1130 IF A$ = "7" THEN 1200
1140 ON VAL(A$) GOSUB 2000, 3000, 4000, 5000, 6000, 7000
1150 GOTO 1000

READY.
1200 REM CLOSE DOWN SYSTEM
1210 END

READY.
2000 REM ADD A REL RECORD
2005 DR=1: NF=2: IF NF$ = "" THEN 2300
2010 REM GET INFORMATION AND COMPILE STRING FOR DISK
2020 IFX=0 THEN 2200
2030 PRINT "ADD A REL RECORD TO "; NF$
2040 FOR A=0 TO F: PRINT "FIELD"; A;
2050 INPUT "CONTENTS "; F$(A)
2060 NEXT A
2070 S$="": FOR A=0 TO F: S$=S$+F$(A)+CR$: NEXT A
2080 IF LEN(S$) > LN THEN PRINT "DATA TOO LONG: RE-ENTER": GOSUB 12000: GOTO 2010

READY.
2100 REM OPEN CHANNEL RELATIVE FILE AND WRITE RECORD TO DISK
2110 DOPE#(NF), (NF$), D<DR>: REM OPEN REL FILE
2120 RECORD#(NF), (X+1), 1: REM POINT TO RECORD NUMBER X+1
2130 PRINT #NF, S$;
2140 DCLOSE#(NF)
2170 I$(X)=F$(0)
2180 X=X+1: RETURN

```

```

4110 PRINT "SEARCHING INDEX IN MEMORY"
4120 FORA=0TO2000:NEXT:RETURN
READY.

5000 REM READ INDEX FROM DISK
5010 INPUT "SEARCHNAME OF FILE (MAX 12 CHARS) ";NF$
5020 PRINT "SEARCHING INDEX FROM DISK FOR ";NF$
5030 DR=0:NF=5:DOOPEN#(NF),<NF$+ ".IND">,D<DR>
5040 GOSUB10000:IFEN<>0THENNF$="":GOTO5000
5050 INPUT#NF,LN,X
5060 FORA=0TOX-1
5070 INPUT#NF,I$(A)
5080 NEXTA
5090 DCLOSE#(NF):GOSUB10000:RETURN
READY.

6000 REM LIST INDEX TO SCREEN
6010 IFX=0THEN4100
6020 PRINT "SEARCH INDEX FOR ";NF$
6030 FORA=0TOX-1:PRINTA+1,I$(A):NEXT
6040 GOSUB12000:RETURN
READY.

7000 REM AMEND RECORD
7010 REM GET KEY AND SEARCH
7020 IFX=0THEN4100
7030 PRINT "SEARCH AMEND A RECORD IN ";NF$:R=-1
7040 INPUT "KEY (FIELD 0) ";KY$
7050 FORA=0TOX-1:IFKY$=I$(A)THENR=A:A=X
7060 NEXT A:IF R<0 THEN 7400
7070 RC=R+1:REM RECORD NUMBER FROM INDEX
READY.

7080 REM READ RECORD
7090 DR=1:NF=6:DOOPEN#(NF),<NF$>,D<DR>
7100 RECORD#(NF),<RC>,1
7120 FOR A=0 TO F
7130 INPUT#NF,F$(A)
7140 NEXT A
7150 PRINT "SEARCH AMEND A RECORD IN ";NF$
READY.

7200 REM DISPLAY OLD RECORD AND UPDATE
7210 FOR A=0 TO F
7220 PRINT "SEARCH";LEFT$(D$,2*#A);A,F$(A)
7230 PRINT "SEARCH";LEFT$(D$,2*#A);"SEARCH";INPUT"ALTER";F$(A)
7240 NEXT A
7250 S$="":FOR A=0 TO F:S$=S$+F$(A)+CR$:NEXT A
7260 IFLEN(S$)>LN THEN PRINT "DATA NOW TOO LONG : RE-EDIT":GOSUB12000:GOTO7100
READY.

7300 REM WRITE RECORD TO DISK
7310 RECORD#(NF),<RC>,1
7320 PRINT#NF,S$
7340 DCLOSE#(NF)
7350 I$(R)=F$(0)
7360 RETURN
READY.

10 REM REL FILE EXAMPLE PROGRAM
20 REM BY DAVID J. POCOCK
100 REM DR = DRIVE NUMBER
110 REM NF = FILE NUMBER
120 REM I$ = INDEX ARRAY
130 REM LN = RECORD LENGTH
140 REM X , RC = RECORD POINTER
150 REM F = NUMBER OF FIELDS (ADD 1)
160 REM D$ = CURSOR DOWNS
170 REM MR = MAX NUMBER OF RECORDS
200 REM SETUP ARRAY AND START VARIABLES
210 DR=1:NF=2:X=0:F=3:CR$=CHR$(13):MR=20
220 D$="SEARCHING INDEX FROM DISK"
230 DIM I$(MR) : REM MAX MR+1 RECORDS IN AN INDEX
240 DIM F$(F) : REM MAX F FIELDS PER RECORD
READY.

1000 REM MENU
1010 PRINT "SEARCH RELATIVE FILE EXAMPLE"
1020 PRINT "001. ADD A RECORD"
1030 PRINT "002. LOOK AT A RECORD"
1040 PRINT "003. WRITE INDEX TO DISK"
1050 PRINT "004. READ INDEX FROM DISK"
1060 PRINT "005. LIST INDEX ON SCREEN"
1070 PRINT "006. AMEND A RECORD"
1080 PRINT "007. QUIT"
READY.

1100 REM SELECT OPTION
1110 GETA$:IFA$="":THEN1110
1120 IFA$<"1"ORAS$>"7"THEN1110
1130 IFA$="7"THEN1200
1140 ON VAL(A$) GOSUB 2000,3000,4000,5000,6000,7000
1150 GOTO1000
READY.

1200 REM CLOSE DOWN SYSTEM
1210 END
READY.

2000 REM ADD A REL RECORD
2005 DR=1:NF=2:IFNF$="":THEN2300
2010 REM GET INFORMATION AND COMPILE STRING FOR DISK
2020 IFX>MRTHEN2200
2030 PRINT "SEARCH ADD A REL RECORD TO ";NF$
2040 FOR A=0 TO F:PRINT"FIELD";A;
2050 INPUT"CONTENTS ";F$(A)
2060 NEXT A
2070 S$="":FOR A=0 TO F:S$=S$+F$(A)+CR$:NEXT A
2080 IF LEN(S$)>LN THEN PRINT "DATA TOO LONG : RE-ENTER":GOSUB12000:GOTO2010
READY.

2100 REM OPEN CHANNEL RELATIVE FILE AND WRITE RECORD TO DISK
2110 DOOPEN#(NF),<NF$>,D<DR> : REM OPEN REL FILE
2120 RECORD#(NF),<X+1>,1 : REM POINT TO RECORD NUMBER X+1
2130 PRINT#NF,S$
2140 DCLOSE#(NF)
2170 I$(X)=F$(0)
2180 X=X+1:RETURN

```

Peripheral Spot

Beginners guide to RS232

from articles by Michael E. Day (Compute) and W. T. Garbutt (Transactor).

In the broadest of terms there are two basic methods any computer can use to communicate with its peripherals and with the outside world. One is parallel data transfer, the other is serial data transfer.

Parallel data transfer involves sending out 8 bits of data (in the case of CBM machines) down 8 separate wires simultaneously and thus transmitting a full byte of data with each transfer. Just sending out a byte of information is of very little value, as the receiving device has no way of knowing if the signals on the bus are valid or they are simply noise. Nor is the device even sure that the information being sent is actually for it and not for some other device. Clearly some other wires must be included to make these matters clear to the receiving device. The function of these wires in controlling the data bus is referred to as "hand-shaking". In the current Commodore machines the "hand-shaking" used is the IEEE-488 (1978) standard. Other standards for "hand-shaking" abound, one of the more common is the Centronics Parallel Interface standard. This standard can be found in a great many of the printers currently available.

Using a parallel bus over a long distance introduces many problems not the least of which is that electric currents will travel down each wire at a slightly different speed. This has the effect over a long distance of causing data bits to arrive at their destination at different times.

While a great many manufacturers provide equipment with an IEEE-488 interface, a far more common method is the serial RS232C standard. A serial interface is one in which only one bit of information is transmitted at any particular time. The transmitting device will take a byte and send out each bit of the byte, one at a time, down a single wire. The receiving device will collect these bits and build them up into a byte again. The major

advantage of this type of arrangement is that data can be transmitted over long distances. With suitable modification this method can be used to transmit data over a telephone line or via satellite.

Essentially RS232C is the title for a standard formulated by the Electronic Industries Association (EIA). As a standard it describes a set of conditions that must exist to provide the housekeeping necessary to interface two devices over a serial data bus.

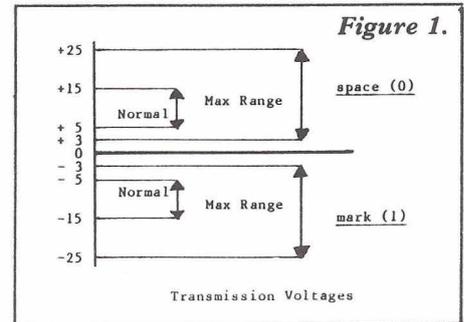
RS232 has existed in three different forms since its formulation in the early 1960's when set down a standard to allow for the orderly interconnection of peripherals to the then newly developing mini-computers. Prior to EIA's RS232 standard what communication did take place was, in the vast majority of cases, handled by the 60 or 20ma current loop teletypes.

The earliest RS232 standard was the RS232A which is totally obsolete, and equipment using this standard is almost non-existent. The RS232B standard is also non-existent. The RS232B is also obsolete but there is still some old equipment around that conforms to this standard. The current RS232C is almost the same as RS232B except that the data signals are inverted.

The standard has gained widespread use not only in the original area of intent, communication between terminal and modems, but also for the interconnection of computer peripherals such as printers, plotters, etc.

Electrical Signal Characteristics

Many of the terms used in communications come from their Teletype ancestors. The terms for logical one and zero, for instance, are Mark and Space respectively. These are the terms that were used when data communications consisted of the presence of or absence of a 20 milliampere current in a circuit. Some teletype equipment still uses



this technique.

The RS232C standard, while still retaining these names, uses Voltage to transmit data instead of current. A space condition (0) is represented by a voltage between +3 and +25 volts, while a marking condition is represented by a voltage of between -3 and -25 volts. In a working situation a range of 5 to 15 volts is more usual. If the plus and/or minus voltage of any transmission line falls below 3 volts an indeterminate state results and the receiving interface will not be able to detect the condition of that signal. See figure 1.

When transmitting over long distance the voltage in the wires will become "attenuated" i.e., there is a voltage loss proportional to the distance travelled. The practical limit for a cable is usually little more than 1000 feet. This can be increased by the use of special line drivers which will boost the signals. Another method of achieving greater distances is the use of a MODulator/DEMODulator (MODEM) at each end of the line. Modems will be the topic of a future article.

Another problem that can be encountered using long cables is interference. Most interference in this case will be electromagnetic in nature. Some examples of the sources of this are fluorescent lighting, laying cable near to mains cables, a nearby radio or television transmitter, proximity to radar, electric motors, transformers, and many more. The easiest way to minimise their effect is to use shielded cable i.e., a cable with braid surrounding the conductors. This braid should be earthed at one end only and should not be used as a ground return wire.

Transmission Types

Two types of data transmission are possible:-

1. Synchronous
2. Asynchronous

Synchronous transmissions are those that have a fixed time interval between the bits of a transmission. The time periods are defined by the transmitting station by means of a Transmit Clock which tells the receiving device exactly when a data bit is valid. Synchronous data is normally organised into rigid groups of characters which are collectively called "protocol".

The advantage of using synchronous communications is that the controlling device has very rigid control over its receiver and because the format of each type of message can be anticipated and checksums are used error detection is very accurate. The prime disadvantage in this type of transfer is the overhead of maintaining all the message types and responses. Because of this limitation it is very rare to see synchronous transmission on a micro-computer.

Asynchronous transmission is by far the more common of the two. It does not rely on a clock but simply assumes where the centre of each should be after detecting the "start" bit which is a one bit-time space or 0 bit at the beginning of each character. This start bit is used to signify the start of a character and can also be used to determine the length of the bit-time. After this bit has been received the bits from the character are transmitted one at a time starting with the least significant bit.

The number of bits in a character can vary. The original 5 bit or "Baudot" code is the original teletype code and is still used on many telex facilities (international telex is 5 bit), 6 bit "Baudot" is also used in some

instances but 7 bit code (usually ASCII) is by far the most common code used. Normally with seven bit code a parity bit will be sent which is why this code is often called 8 bit code. The parity can either be odd or even or can be unused as long as both transmitting and receiving devices are expecting the same thing.

The final bit sent in the asynchronous character is a "stop" bit. The length of this may vary but its minimum length will be stipulated as either 1, 1.5 or 2 bit-times. This bit is a mark of logical 1 and is terminated by the start bit of the next character. See figure 2.

The major advantage of asynchronous communication is the relative ease of use and the fact that no set groups of characters are required. This is particularly convenient for use in micro-computers where memory space is at an optimum. The disadvantage is that parity is the only protection against error. Another feature of some modes of transmission is that the receiving station will echo the message back as a check of the validity of the data. In an important data link or where a long distance is to be covered asynchronous is not a fail-safe as synchronous.

Control Signals

There are three major groups of control codes. Variations from these groups abound so the appropriate technical manuals should be referenced when interfacing devices.

Group 1 consists of:-

- Pin 1 Protective Ground
- Pin 2 Transmit Data
- Pin 3 Receive Data
- Pin 4 Request to Send
- Pin 5 Clear to Send
- Pin 7 Signal Ground

Note Request to Send and Clear to

Send will sometimes be linked when using the standard to interface peripherals.'

Group 2 consists of Group 1 plus:-

- Pin 6 Data Set Ready
 - Pin 8 Data Carrier Detect
 - Pin 20 Data Terminal Ready
- Group 2 is used mainly in connecting terminals to CPU's and in CPU to CPU transmissions.

Group 3 consists of the other two groups plus:-

- Pin 22 Ring Indicator
- This group is normally only found in fixed station Auto-answer Modems.

Further information of the pin assignment is given in figure 3, also in figure 3 is a diagrammatic representation of the RS232C connector.

Pin 1 Protective Ground. This is a safety ground. When using shielded cable the shield should be connected to this pin at one end.

Pin 2 Transmitted Data This is the pin on which a device will output data. This pin will be held in a marking condition when not in use. Before data transmission can take place Request to Send, Clear to Send, Data Set Ready and Data Terminal Ready must all be on.

Pin 3 Received Data is the pin on which incoming data will be seen. This data is the data from the Transmit Data pin on the transmitting device.

Pin 4 Request to Send this pin is turned on when the device wishes to transmit data.

Pin 5 Clear to Send indicates the data-set or Modem is

Figure 2.

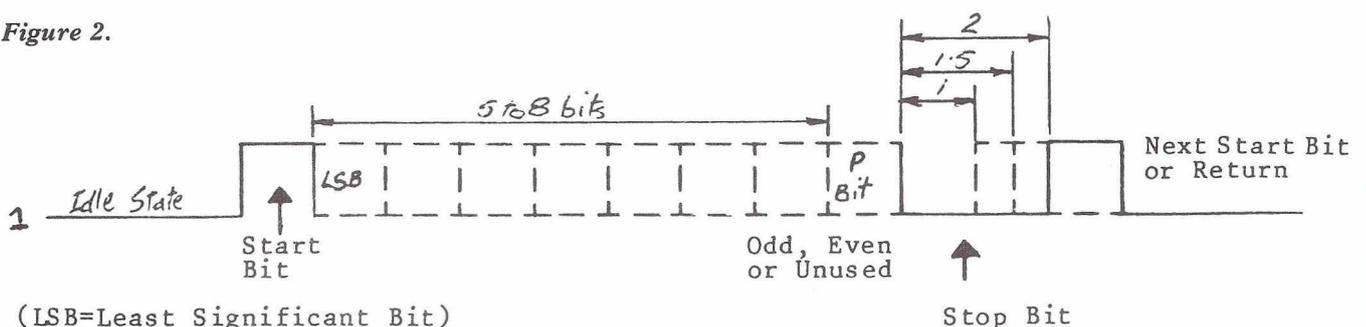
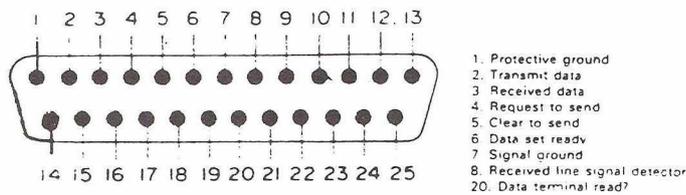


Figure 3.

THE RS-232C INTERFACE



RS-232 PIN ASSIGNMENTS

The RS-232C interface is the set of connections between two pieces of equipment which allows them to communicate by exchanging data

Pin 23 Data Signal Rate Detector. Where a two-speed Modem is installed this line indicates the rate selected.

Pin 24 Transmit Signal Element Timing is a pin which allows the entering of an external timing source for the transmission rate.

Pin 25 Unassigned

Parallel interfacing

Parallel interfacing is used when high speed data transfer is required over short distances between two devices, such as a computer and a peripheral such as a disk drive or printer.

The analogy of parallel data transfer is similar to that of a horse race. All the horses (or data bits) are lined up at the starting line (the sending equipment). The line of horses is straight at the beginning of the race, but by the time the finishing line (the receiving equipment) is reached, the horses are staggered, some arriving much later than the first across the line.

This is because each data bit is travelling down a different cable, and the characteristics of each cable will differ.

As each data bit does not arrive together, some method is required of knowing when the data is valid. This procedure is known as handshaking.

Figure 4 gives an example of the handshaking procedure for a Centronics parallel interface. The data is strobed after it has settled on the data lines. When the receiving device recognises the STROBE signal, it asserts a BUSY signal to indicate that no more data should be sent.

When the receiving device has finished with the data, it unasserts the BUSY signal, and sends an ACKnowledge signal to the sending device, and the handshaking is complete.

Parallel data transfer is a very simple method of transferring data, however problems can arise because of the various methods of handshaking. Some devices handshake on the BUSY signal, some on the ACKnowledge, and some have the signals inverted to trigger on a rising or falling edge of the signal. Closer examination is required for each application.

The IEEE-488 Data Bus is one

prepared to transmit data. This signal is generated in response to Request to Send. This line may be connected directly to its own Request to Send Line when connecting peripherals.

Pin 6 Data Set Ready indicates the data-set or Modem is ready to send and/or receive data. This signal may be tied to its own Data Terminal Ready (Pin 20) when interfacing peripherals.

Pin 7 Signal Ground is the ground reference for all the other signals in the group.

Pin 8 Data Carrier Detect indicates that there is a carrier on the line from the other data-set (This usually only applies to modem communication).

Pin 9 Reserved for Data Set Testing.

Pin 10 Reserved for Data Set Testing

Pin 11 Unassigned.

Pin 12 Secondary Receive Line Signal Detector. This line indicates the secondary (or reverse) channel. This Channel is used only on high speed lines (9600 baud) and is used to transmit messages and status in the opposite direction to the main flow of data. The secondary channel cannot transmit in the same direction on both primary and secondary channels at the same time.

Pin 13 Secondary Clear to Send is the same as Clear to Send except on the

secondary channel. None of the pins on the secondary channel are used in systems running at less than 9600 baud.

Pin 14 Secondary Transmitted Data is the same as Transmitted Data except on the secondary channel.

Pin 15 Transmission Signal Element Timing is the transmit clock used in synchronous communications.

Pin 16 Secondary Received Data is the same as Received Data except on the secondary channel.

Pin 17 Receiver Signal Element Timing is the clock received from the other device while receiving data in synchronous transmission.

Pin 18 Unassigned

Pin 19 Secondary Request to Send is the same as Request to Send except on the secondary channel. Note this signal will not turn on while the Request to Send signal is on

Pin 20 Data Terminal Ready is the signal used to indicate to the Data Set that the device is ready to receive and/or (depending on Request to Send) transmit data.

Pin 21 Signal Quality Detect indicates (when off) that the carrier on the transmission line is in such a state that an error is highly likely if transmission is attempted.

Pin 22 Ring Indicator is used on Auto-answer modems to indicate the phone is ringing.

method of parallel data transfer. The method of handshaking will be discussed in future articles, as this is a very important Data Bus for the Commodore range of micro-computers.

A very common parallel data bus that is used on most types of printer, is the Centronics Interface Standard. Figure 5 details the standard, however there may be minor variations between printers.

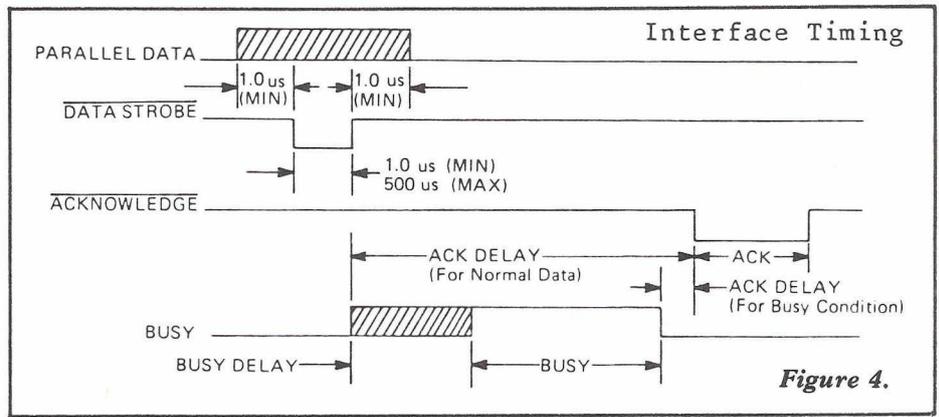
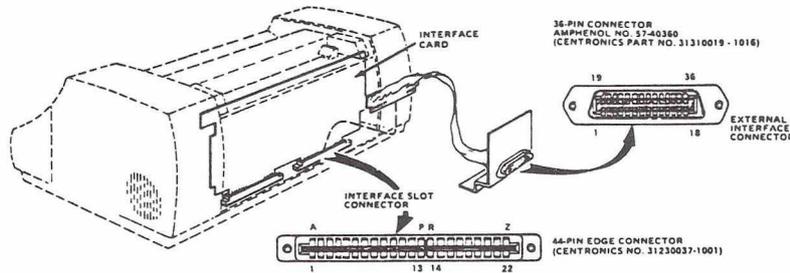


Figure 4.

Figure 5.

STANDARD INTERFACE SIGNALS

The following table describes the standard interface signals available at both the interface slot connector and the external interface connector of all Centronics Printers.



SIGNAL NAME	INTER-FACE CONN.	INTER-FACE SLOT	SOURCE	DESCRIPTION
(DATA STROBE)	Pin 1, 19	Pins 21, Y	Input Device	A 1.0 usec pulse (min.) used to clock data from the processor to the printer logic.
DATA 1	2, 20	18, V	Input Device	Input data levels. A high represents a binary ONE, a low represents a ZERO. All printable characters (i.e., codes having a ONE in DATA 6 or DATA 7) are stored in the printer buffer. Control characters (i.e., codes having a ZERO in both DATA 6 and DATA 7), are used to specify special control functions. These codes are not stored in the buffer except when they specify a print command and are preceded by at least one printable character in that line.
DATA 2	3, 21	16, T	Input Device	
DATA 3	4, 22	17, U	Input Device	
DATA 4	5, 23	20, X	Input Device	
DATA 5	6, 24	15, S	Input Device	
DATA 6	7, 25	11, N	Input Device	
DATA 7	8, 26	19, W	Input Device	
DATA 8	9, 27	12, P	Input Device	
ACKNLG	10, 28	22, T	Printer	Acknowledge pulse indicates the input of a character into memory or the end of a functional operation.
BUSY	11, 29	3, C	Printer	A level indicating that the printer cannot receive data. For conditions causing BUSY, refer to Busy Condition Timing Table.
PE	12	9	Printer	A level indicating that the printer is out of paper.
SLCT	13	F	Printer	A level indicating that the printer is selected.
± OV	14	7	Printer	Signal ground (Formerly SS signal, older version)
OSCXT	15	H	Printer	A 100 KHz signal (Models 101, 101A, 102A, 101S) or 100-200 KHz signal (All other models).
± OV	16	A		Signal ground
Chassis Gnd	17	—	Printer	Frame ground
+5V	18	13	Printer	+5 Volt power buss
(INPUT PRIME)	31, 30	L, 10	Input Device	A level which clears the printer buffer and initializes the logic. (Not in 101).
FAULT	32	M	Printer	A level that indicates a printer fault condition such as paper empty, light detect, or a deselect condition. (Not in 101).
Line Count Pulse	34, 35	2, D		Both sides of the line count switch appear at the interface connector. This switch is opened and closed during each line feed operation. A level delivered to the switch would be pulsed off and on each time a line feed operation is performed.
Not Used	36			(Series 300 and 500 except 306SC, 503).

NOTES:

1. Second pin number indicates twisted pair return (± OV).



ADD POWER TO YOUR COMMODORE COMPUTER

£49
+ VAT

POWER produces a dramatic improvement in the ease of programming BASIC on Commodore computers. POWER is a programmer's utility package (in a 4K ROM) that contains a series of new commands and utilities which are added to the Screen Editor and the BASIC Interpreter. Designed for the CBM BASIC user, POWER contains special editing, programming, and software debugging tools not found in any other microcomputer BASIC. POWER is easy to use and is sold complete with a full operator's manual written by Jim Butterfield.

POWER's special keyboard 'instant action' features and additional commands make up for, and go beyond the limitations of CBM BASIC. The added features include auto line numbering, complete tracing functions, single stepping through programs, line renumbering, and definition of keys as BASIC keywords. POWER's 'WHY' command enhances debugging by listing the appropriate program line and highlighting where BASIC stopped executing. The cursor movement keys are enhanced by the addition of auto-repeat, and text search and replace functions are added to help ease program modification. POWER can even execute a sequential tape or disk file as though

it were typed on the keyboard, allowing the user to merge two BASIC programs together. Cursor UP and Cursor DOWN produce **previous** and next lines of source code. COMPLETE BASIC PROGRAM listings in memory can be displayed on the screen and scrolled in either direction. You can even add your own commands to BASIC. Like our very successful Word Processing Programs (the "WordPro" series), POWER even includes convenient "stick-on" keycap labels which define new functions on the keyboard. POWER is a must for every dedicated CBM user.

Call us today, for the name of the Professional Software dealer nearest you.

Professional Software, Ltd.

153 High Street

Potters Bar

Hertfordshire EN6 5BB

Tel: (STD 0707) 42184 / (STD London 77)

MICROCOMPUTING TRAINING COURSES FROM COMMODORE

Now you have the chance to come away for a few day's uninterrupted study, free from all distractions, to learn more about whatever aspect of microcomputing interests you, under the personal tuition of Mike Gross-Niklaus, Commodore's Training Manager and his expert assistants.

More confidence, more practical skills and greater knowledge of all aspects of microcomputing, from BASIC for beginners to Advanced Assembler Programming for really fast applications, is what we promise all participants. There's more too - though the study is intensive, our 'hands on your shoulder' teaching methods make you feel comfortable, relaxed and the learning easy.

Each course includes a minimum of 60% practical work, numerous step-by-step exercises to practice and consolidate new-found skills plus individual advice and personal tuition which keeps the pace geared to your own needs.

The atmosphere is friendly and informal with plenty of time to get together for discussions, serious or not so serious.

We give outline details of the training courses below, but SEND OFF TODAY FOR THE FULL BROCHURE GIVING FULL INFORMATION ON THE NEW MICRO COMPUTER TRAINING COURSES FROM COMMODORE.

● Ond day work shops & seminars

The subject of these seminars range from installing microcomputers in small businesses to interfacing a microcomputer to industrial equipment. (Please see you local dealer for schedule.)

● BASIC for Beginners

3 days Mar 23 £210 ex vat

Start from scratch and check out knowing how to write useful programs in BASIC. The highly-practical course, with each student allocated to a Commodore PET and Disk system, extends into the first two evenings. You'll come away tired, but certainly programming and undoubtedly very pleased with your newly acquired knowledge and skills.

● Disk File Programming

3 days Mar 9, Mar 16, Apr 13 £210 ex vat

Learn how to use the 'tools' available in the PET and Disk operating systems, how to apply them effectively to create usefully structured, secure and backed-up data files, both sequential and direct access. Each student has access to a Pet and Disk system and gains the knowledge and confidence to tackle other applications involving disk files.

● Program Planning Course

3 days Mar 2, Mar 16, May 4, June 15 £210 ex vat
For users who want to do their own programming,

this course shows you how to turn an idea for a micro computer application into a well designed, accurate program, using BASIC. By the end of the course you will have designed, coded, tested and documented a sizeable, general purpose, utility program, structured the program into a series of modules, created program menus, created and amended tables of Data, etc. You'll leave knowing how to tackle other applications, with the designs and coding skills to do them well.

● Assembler Programming

3 days Mar 9, May 4, July 20 £210 ex vat

If you've tried to learn Assembler Code before and have failed because of the vast amount of information that seems to be needed you probably won't believe that you'll write your first successful Assembler Code program within an hour of starting the course - but you will, with our unique teaching approach. Go on to learn how to use the Editor, Assembler, Loader and Extramon and the whole range of 6502 facilities effectively.



TO: Training Department
Commodore Business Machines Ltd.,
675 Ajax Avenue Slough. Tel: Slough 74111

Please send my copy of your brochure MICRO COMPUTER TRAINING COURSES FROM COMMODORE with schedule & booking form.

I wish to make a provisional reservation/s for the following course/s (please give preferred dates.)

.....

.....

.....

Name

Business/Profession

Tel(home)

(work)

Address

.....

County.....Post Code.....

Venues include London, Manchester & Worcester.

Please complete the coupon for further details.

BUY IT WRITE AWAY!

PAGEWRITER is a machine code word processor crammed into a single 2k chip! Obviously in a mere 2k we couldn't fit all of the facilities of WORDPRO, WORDCRAFT, or our own MICROSCRIPT, but you'll be pleasantly surprised to find how powerful and easy-to-use it is!

PAGEWRITER doesn't limit you to a 40 or 80 character line length, but scrolls the screen left or right, up or down as the cursor nears the edge. In fact, the electronic 'page' that you type onto can be up to 240 columns wide and up to 191 lines long (subject to memory size).

PAGEWRITER prints out your text exactly as you see it on the screen! There are no margin or tab settings to worry about. When writing or editing a document all the normal cursor controls may be used - and in control mode PAGEWRITER has more sophisticated functions enabling you to DELETE or INSERT a LINE, or MOVE a BLOCK of text. If you use the CBM 3022 or 4022 printer then PAGEWRITER gives you full control over the programmable character - as many as 26 characters can be defined at any time (a pre-defined set is included in the chip).

When you've finished writing you can SAVE text to cassette or disk. The whole thing is really so amazingly simple that you'll wonder why nobody thought of it before! And remember, because PAGEWRITER is written entirely in 6502 machine code it's FAST!

PAGEWRITER is available to fit in any spare ROM socket of an 8, 16 or 32k PET with New Roms or Basic 4 (please state socket & model when ordering). And the best thing of all is the price, just £39 plus VAT!

P.S. PAGEWRITER is also available in a 4k chip with ARROW, the chip that can LOAD, SAVE, VERIFY and APPEND at 6 to 7 times normal speed. ARROW on its own is £30 plus VAT, the two together cost £69 plus VAT.

ALSO IN OUR NEW CATALOGUE....

SUPERSORT (£40) heads the list of new utilities in our latest catalogue. Whereas MULTISORT (£25) will sort a string array and move a number of other string arrays around in parallel, SUPERSORT handles numeric arrays too, and will sort on one field within another all the way down the line.

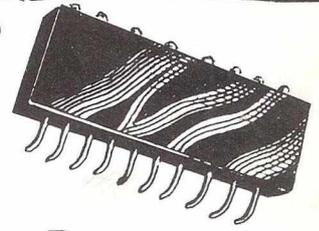
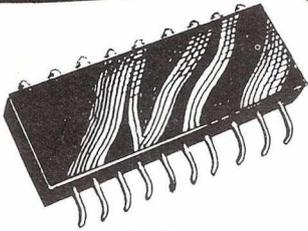
DISK SEARCH 2 (£40) is an improved version of another powerful utility. If you need to search a RELATIVE FILE all you have to do is put the strings you are looking for into an array - then call up DISK SEARCH 2. You can request records that contain ANY ONE of the strings sought - or just those that contain ALL of the them. Why buy a database program when with DISK SEARCH 2 you can design your own?

MAKRO DISASSEMBLER (£25) is a true disassembler - it will take a machine code program apart and turn it into ASSEMBLER SOURCE CODE complete with labels! If you own MAKRO ASSEMBLER (£50) you can then make changes and re-assemble the code - just as if it was one of your own programs! And we've developed a special version of MAKRO with a find-and-replace facility that you'll find particularly useful (put in your own labels etc). MAKRO-XR is available as an upgrade to the standard MAKRO at a cost of just £10.

SUPERSOFT

First Floor, 10-14 Canning Road, Wealdstone,
Harrow, Middlesex, HA3 7SJ, England
Telephone: 01-861 1166

audiogenic LTD CHIP SHOP



EDEX 2.0 & 4.1

adds commands to BASIC for use within your Program

**IF THEN ELSE ● PLOT ● BEEP ● PRINT USING ● SWAP
MERGE ● HARD COPY ● PLUS A RANGE OF TOOLKIT
TYPE FUNCTIONS AND A FAST EDITING SYSTEM**

EDEX is an extension to BASIC which considerably enhances the potentialities of the Commodore PET/CBM. It consists in a 4K-BYTE ROM which installs inside the PET/CBM.

EDEX is compatible with Commodore disk devices as well as with the DOS Support Program.

EDEX operation is fully transparent towards the Microsoft Basic Interpreter

EDEX is fully compatible with prior programs written without EDEX.

AUTO

Activates automatic line numbering.

APPEND *

Allows the creation of a program with a subroutine library

BEEP

Gives a sound of programable pitch and duration

CALL

Calls a machine language subroutine with transmission of up to 16 arguments

DELETE

Allows multiple line suppression

DUMP

Lists all variables in a program, together with their values

EDITING *

e.g. @ M prints MIDS

ERROR

Shows where an error has occurred

FIND

Lists all lines where a given character string is present

EDEX 2.0 for use with BASIC 2 40 Column Pets **£39.50**

HARD COPY

Dumps screen to printer

IF THEN ELSE

With up to 16 nested conditions

MERGE

Merge two programs files

PLOT

Plots curves of 50 x 80 or 160 resolution

PRINT USING

Formats printing on screen or any printer

RENU

Program renumbering

RESET

Suppresses a dot (contrary of PLOT)

SWAP

Swap one program for another keeping variables

TRACE

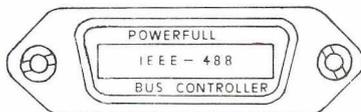
Single line execution (displayed at top of PET)

* **EDEX 2.0 only** **EDEX 4.1 only**

EDEX 4.1 for use with 80 Column Pets **£49.50**

Available shortly for BASIC 4 40 Column PETS

IEEE-488 PACK



The end of instrumentation's problems. It resolves all kind of troubles:

- Time-out
- Special characters ("null", and so on...)

IEEE-PACK allows the use of IEEE-488 Universal Commands:

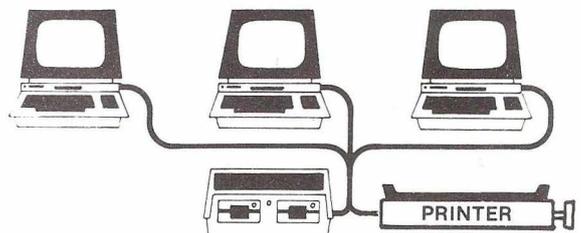
- | | |
|---------------------------------|------------------------------------|
| - DCL (Device clear) | - SDC (Selective device clear) |
| - SPE (Serial poll enable) | - SPD (Serial poll disable) |
| - LLO (Local lockout) | - GTL (Goto local) |
| - PPL (Parallel poll configure) | - PPU (Parallel poll unconfigured) |

IEEE-PACK also allows BASIC interrupt with functions:

- ONKEY "x", line number
- ONSRQ line number (On Service Request)

IEEE-PACK comes complete with two ROMs. **£89.50**

MULTEX



MULTEX allows several CBM 8032 to work together on the same peripherals.

MULTEX is a ROM which replaces a ROM of the CBM 8032.

Except the substitution of this ROM no other modification is required on the CBM 8032.

MULTEX is much cheaper than any other system.

MULTEX £69.50



ALL PRICES INCLUDE V.A.T. & P.P.

AVAILABLE FROM ALL GOOD DEALERS OR DIRECT FROM



AUDIOGENIC, P.O. Box 88, 34-36 Crown Street, Reading, Berks. Tel: Reading (0734) 595269