A Gareth Powell Magazine

The Australian COMMODORE REVIEW

Vol 1 No 3 May $3*

Registered by Australia Post Publications No. NBP 6656
*Recommended retail price

Taking care f bu sine s Joyst icking Lookin ur C m modore E ari thmetic and re 64 sof The Pony not print ng of software your own ad How to write your ow adventure game – Part // Principles of a good game Basic tips View from the hold Taking care of busin ess Joysticking Looking after

If a picture's worth a thousand words...

SPECTRAVIDEO 328

COMMODORE SX64

COMMODORE 64

THE LARGEST RANGE OF 64 SOFTWARE IN AUSTRALIA

STAR GEMINI 10X
C/64 Compat. with VICS Print 64

KOALAPAD
TOUCH TABLET

SPECTRAVIDEO 318
FULL RANGE OF SOFTWARE
AVAILABLE.

You can't say we didn't tell you that the Spot has the largest range of C64 software, plus a wide selection of hardware from the leading manufacturers - at the most competitive prices.
With expert advice and reliable service.
Thinking personal computing? Then there's only one Spot... sorry two!

the Computer Spot

MLC Centre, Martin Place
Sydney. Phone: 235 2971.
Eastgate Centre, Bondi Junction
Phone: 387 5208

IMAGOMAT 2032/1

Contents

Cover Illustration:
Kenny Alder, Imagomat Art Studio
(For new Ozi Soft C64
Entertainment Program)

The Australian
Commodore Review
Top Rear, 4 Carrington Rd,
Randwick, NSW 2031
Phone: 398 5111

Published by Gareth Powell Pty Ltd

Editor: Andrew Farrell

Art Director: Avrille Goldsworthy
Production Manager: Brenda Powell
Photographer: Gary Williams

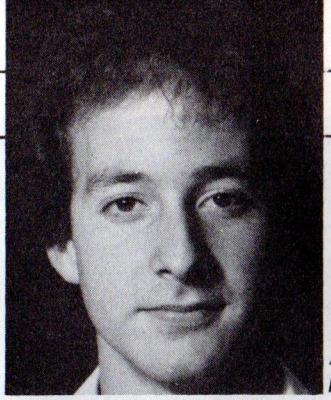
Advertising Enquiries:
Craig Hoskinson,
Top Rear, 4 Carrington Rd,
Randwick NSW 2031.
Phone: 398 5111

Subscription Manager:
Tina Spathos
Phone: 398 5111

Typesetting:
Printaction

Distribution:
GORDON & GOTCH LTD SYDNEY

Printed by: Ian Liddell Commercial Printing



Andrew Farrell

To use a programming phrase, a PEEK inside this issue will show that some things are different. The only major change is the name of the editor. There are a few new names in the rest of the editorial line-up as well – but not to worry, I'm sure we'll all get on fine.

Our objectives will remain the same as in the original issues but there will be some planned changes to fit in with what we have found you, the reader, want.

There will be more Australian programs and and tips appearing, because we have managed to squeeze some more programmers out of the woodwork. This will still be a magazine for beginners. We will try to keep away from any programmer's jargon or the dreaded "computer speak".

"Australian Commodore Review" is a magazine for everyone. For example, several teachers will be joining us, initially to review software, but later on they'll be telling us how Commodores are used in schools. And, press stories to the contrary, they are used a whole lot more than some people would have us think.

Australia has its fair share of Commodore User Groups, one of which Paul Bartolo and I had the fun of starting. To let others know about these User Groups, we invite everyone who is on a committee and wondering what to do next, to send us their name, address and when and where the group meets. With a little cooperation, we should be able to publish a nice big list next month.

New software continues to stream in from overseas, despite the current copyright problems. No doubt it will be sorted out in due time, but meanwhile I'm interested in hearing your points of view on this subject as we attempt to shed some light on the mess.

It's obvious that locking programs will lessen the problem of "pirating" commercial software. However, there are several snags involved with this answer. Many users are finding that having upgraded to a disk drive they must repurchase software on disk, as it is not possible to transfer their old programs from cassette because they are protected.

Must the user pay the author two lots of royalties?

Would it be possible to organise an exchange service from cassettes to disk?

These are questions Australian software houses should look into if they wish to further their case for some form of protection.

US software companies have already implemented an exchange service to a certain extent. Maybe, for a change, we could get into step with the rest of the world.

Finally, this is your magazine. Not mine, not Gareth Powell's. Yours. If there is anything you would like included, anything that you feel deserves a brickbat or a bouquet, write to us. We are waiting to hear from you. □

News . . . Ram rumblings

MULTILINK

Cybex Computing recently released a real networking system for the Commodore 64 which is substantially better than previous systems. A networking system allows you to hang a lot of machines together so that they can all inter-relate with each other and, if necessary, use the same printer, the same disk drive or the same plotter.

Multi-Link provides full communications amongst up to 48 Commodore 64s, making it absolutely ideal for schools. It is rare that a class has more than 48 ankle-biters in it at one time: The satellite computers are set up in groups of six 64s daisy chained together and then connected to the master control unit.

IEEE peripherals – which are most of the peripherals produced for Commodore machines – may be connected to the master unit as well as standard serial devices, which covers almost every device anyone is ever likely to want to connect.

Programs may be downloaded at high speed from the master unit onto any of the satellite computers. At a recent demonstration at Commodore Computers it was shown that even protected software could be used on this system without any problems.

From the master computer – which is where a teacher would be sitting – it is then possible to view the screen of any of the satellites as well as being able to disable the keyboard of that machine or reset that machine.

This enables the teacher to maintain complete control over what the students are doing, as well as monitoring their progress. ("Right, Andrew, I've told you for the last time to stop pulling Tina's pigtails. I'm going to disable your keyboard for half an hour until you learn to behave.")

Messages may be sent from satellite to satellite using a few simple commands.

This may or may not be a good thing depending on the class and whether the teacher has got them under control.

Each satellite can also access any peripherals connected to the master computer.

Security is provided at various levels including password protection and ID prefixes for special files. If it is going to be used in schools you can understand why this would be very necessary.

Cybex expect an entire system complete with all necessary hardware, installation, software and eight satellites to sell for around $2500. Interesting that a comparable commercial set up using other PCs would cost almost precisely ten times as much.

For further information contact Cybex Computing on (08) 333 0711.

Micro Illustrator – Koala Ware

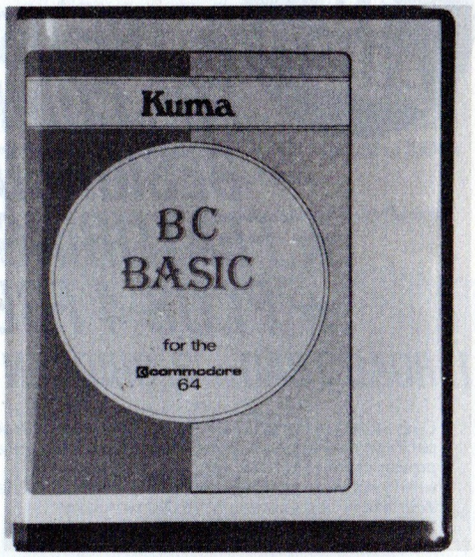
Micro-Illustrator

Famous for its appearance as part of the familiar Koala-Pad package, the Micro-Illustrator will shortly be available from Ozi Soft for use with a light-pen.

The Micro-Illustrator allows you to design your own Hi-Resolution graphics pictures, with options such as draw, fill, box, line circle as well as a variety of brush sizes and colours.

Recently we had a brief look at the light pen used with the Illustrator and found it to be quite reliable. Some background colours proved the pen inoperative, although adjusting the television contrast helped the reception just a little.

Interestingly enough, Commodore recently jumped, hopped and skipped around as they described a fantastic new program to be released soon which lets you draw on the screen.



BC BASIC – full review next month

Could it be the same one?

Ross Bloore, their resident programming guru, was quite sure that it was. He could even be right.

B C Basic

Micro-mail are now importing

several interesting bits and pieces from the UK, the most interesting being a Simon's Basic look alike called B.C Basic. For $80 you get a plug in cartridge and manual that explains the new range of commands in fairly understandable detail.

Extra facilities are provided for Hi-Res graphics, sound, sprites, programmable characters and binary and hex conversion. Watch for a full review next month.

Medibill

Some years ago, John and Jenni Gyffen of Gippsland Business Computers started putting together a program called "Timeline" on the Commodore 8000 series and like so many programs it went to Version 3 before John was happy to see it hit the market place.

It was essentially a database with some unique features enabling the end user to put together a very powerful information gathering program.

Some 12 months ago a couple of local doctors approached John to put together a reasonably priced billing package and it was decided to use many of the ideas gained in programming "Timeline". Working closely with the local doctors John and Jenni put together Medibill and during this past month have released the "base" version.

"Base" version because John is adamant that over the next year or so doctors will be able to get updates to plug straight into their initial program, adding features such as comprehensive patient record keeping and the capability to communicate between surgeries.

Medibill, at present, allows for three terminals, up to 18 practitioners and allows the choice between Medicare and private billings for each patient. It will maintain records on up to 65,000 patients.

On the subject of Medicare – another important feature is the program's ability to adapt to Australia's seemingly everchanging health schemes.

When a doctor buys Medibill he/she receives a copy of Timeline, the database program that started it all, gratis, free of charge, no extra cost.

Medibill starts at around $8,000, depending upon hardware requirements. This price includes both software and hardware.

It's good to see the Gyffens produce a program as upmarket as this and they deserve success. It appears Commodore have faith in the program as they are funding an advertising program to capitalise on the program's potential.

For further information contact Jenni or John Gyffen on 051-52-5939. □

Taking care of business

By Duncan McCann

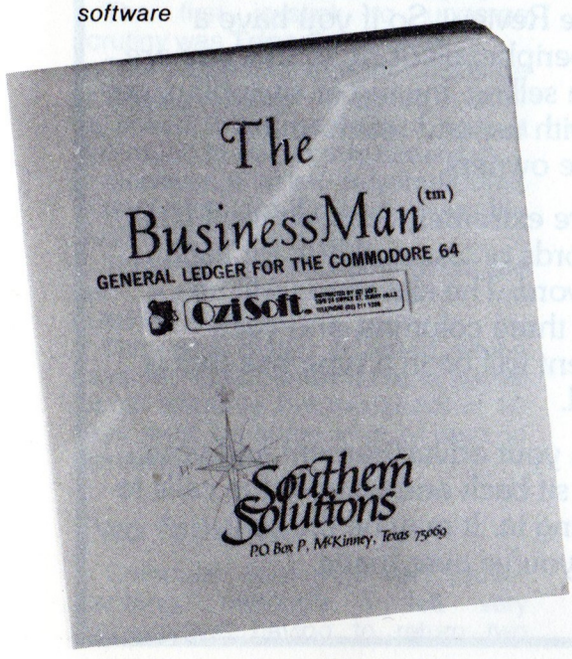
When Commodore entered the market with their 64K computer it would be reasonable to assume that nobody really expected the machine to be anything other than a glorified Vic 20.

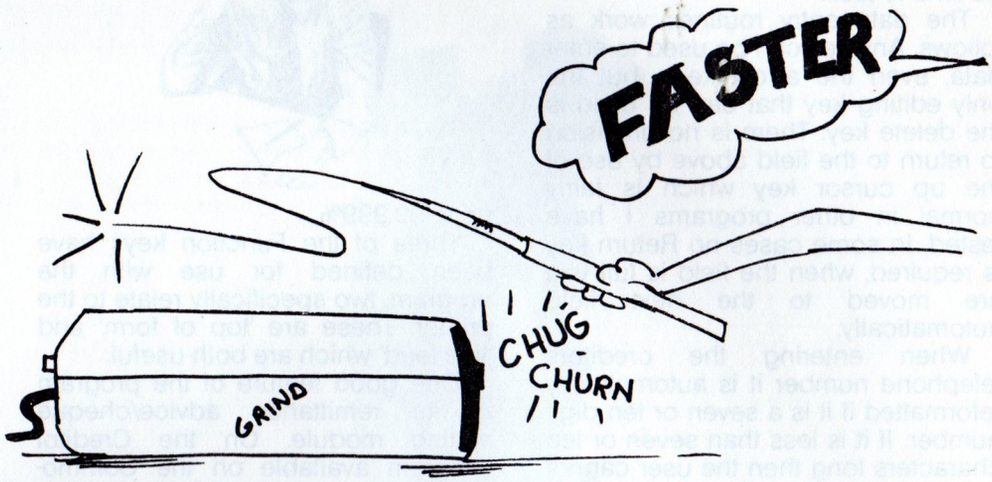
If you think carefully about the speed of the 1541 disk drive you will immediately realise that no designer would have produce a peripheral such as that if it was known that it was eventually intended for business use. It is just too damn slow to work with.

Having said that, it cannot be denied that the Commodore 64 has now been taken seriously as a contender in the bottom and very profitable end of the business market in Australia and in New Zealand. Indeed, in the latter country it is probable that the Commodore 64 is the most popular small business computer on the market.

It is not, it was never intended to be, it will never become, a big business machine. But for carpenters and plumbers and carpet cleaners and all other sorts of small businesses the Comodore 64 is a very good and a very inexpensive way to get into the world of computerised accounting.

The BusinessMan – Ozisoft business software





Even accepting the slowness of the disk drives, there are a lot of people out there saving large amounts of time and effort by having the Commodore 64 do their cash book, keep track of creditors and post their general ledger.

All businesses whatever their size can make use of spreadsheets and databases – and the ones that are available on the Commodore 64 are no less effective than the extremely expensive counterparts on other, larger and more expensive machines.

If a businessperson decides that the Commodore 64 is the ideal machine for a small to medium size business the first task is to select Software that will take care of the business needs.

It is here that the problems begin.

What is good software?

Will help be given after sales to make sure that it runs properly?

Does special stationery need to be printed?

On and on go the questions and hopefully we can supply some answers through the pages of this magazine.

Over the next few issues we will look critically at most of the business packages available for the 64. We will look at the local market and as well software coming in from overseas.

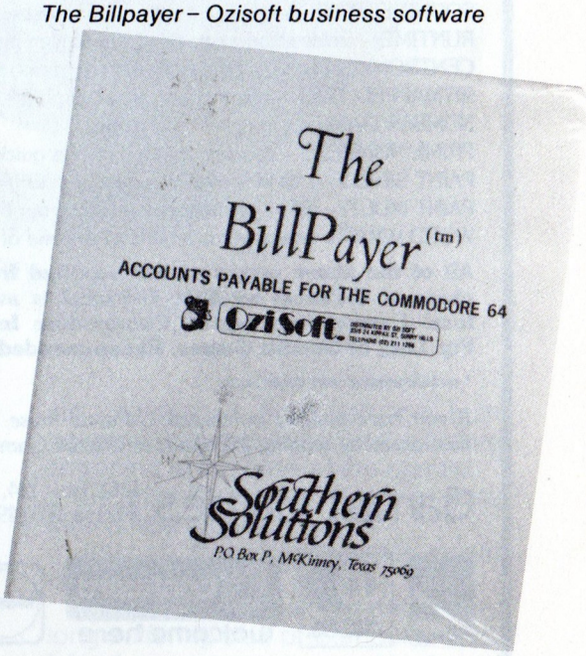
Two business programs that are brought in from the U.S. by Ozi Soft, a company well known for some quite excellent games programs for the 64, are Billpayer and Businessman.

The Billpayer

The Billpayer can be used with any combination of disk drives, from 1 x 1541 to the 9090 hard disk (although I cannot think of a good reason to use the Commodore 64 with a hard disk).

The Master Program disk can be used as a data disk but the Master cannot apparently be backed up. The printers recommended for use are the 1515, 1525 (neither available now), the 8024 CBM model, the 4022 and the 8023P. The latter both require an IEEE Interface.

The Billpayer – Ozisoft business software

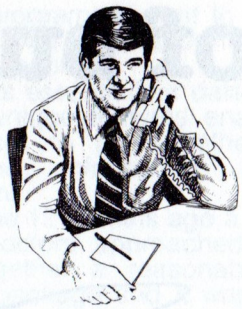


According to the manual the data entered in this program can also be used with Easy Calc although we had no time to test.

The data entry routines work as follows. Any key can be used to enter data, even the arrow keys, but the only editing key that can be used is the delete key. There is no provision to return to the field above by use of the up cursor key which is fairly normal in other programs I have tested. In some cases no Return key is required, when the field is full you are moved to the next field automatically.

When entering the creditors telephone number it is automatically reformatted if it is a seven or ten digit number. If it is less than seven or ten characters long then the user cannot format it into an easily read form as it rejects the (space bar) and (-) as illegal entries.

Discounts for early payment of invoices can be entered as a figure up to 99.999%.



Three of the Function keys have been defined for use with the program, two specifically relate to the printer. These are 'top of form' and 'line feed' which are both useful.

One good feature of the program is its remittance advice/cheque writing module. On the Creditor program available on the Commodore Business the program does not print cheques. Billpayer does.

The reports are not excessive but everything that can be needed is printed out.

The Businessman

The Businessman is partially integrated with the Billpayer.

The information from the Billpayer transfers to the Businessman.

I could not get the program to format a seperate data disk to use with the master program disk instead of writing the data to my copy of the program.

You cannot use the Posting and End of Month options without a printer. This makes a fail-safe method.

In short, the programs are easy to use, although frustrating at times. This is an American package and you simply have to put up with the American spelling if you are going to use them. They are both examples of the sort of integrated package that we will be seeing more and more of in the very near future.

The packages are priced at $139.95 each and can be purchased from Ozi Soft dealers. □

Joysticking

by David Ives

For the last two months or more we have been testing joysticks. At least that is the excuse we've given when we've been caught playing video games.

I can't really say it's been a productive time. We've learned a lot, it's true, but destructive is probably a better adjective for what we've been doing.

There was a time not so long ago when we despaired of finding a stick that would stand up to the rigours of an eight-year-old boy and his friends zooming full tilt into the Lazer Zone.

The major tests we applied to the sticks could hardly be called scientific, but they were practical.

Test One we called the Ladder Challenge and Test Two the Octostruct.

The Ladder Challenge was seeing how well the joystick could deal with the first level of the advanced section of Jumpman – if our hero could cling to the ladder for a complete circuit at speed 3 without hurtling to his doom the joystick was deemed to have passed.

Octostruct was to see how long the instrument could stand up to the ministrations of the aforementioned eight-year-old.

The results are not encouraging.

The first joystick to undergo scrutiny was **Triga Command.**

This passed the Ladder Challenge with flying colours – a bonus of 1200 on speed 2 isn't bad (OK, I know it isn't much to you arcade hot-shots).

However, it lasted only three days of the Octostruct. We were, to say the least, a bit disappointed.

Thinking we had probably got the bad apple from the barrel, we obtained another.

This was better, it lasted seven days. All right, I know it has a twelve month warranty, but two sticks in 10 days must be some kind of pass in the failure stakes.

Two failures

Computer Spot were very understanding. Although I felt very embarrassed having to return two joysticks within a week, they cheerfully and with great courtesy refunded our money and then sold us a **Pointmaster.**

Wico Command Control Joystick

The Pointmaster is of course, more expensive than the Triga Command.

It is just as comfortable to hold as the latter, and it passed the Ladder Challenge easily and with honour. I would have liked to report that we are still using it. I must tell the truth however: it lasted three weeks before snapping at the base of the handle.

By this time I was feeling more than a little piqued.

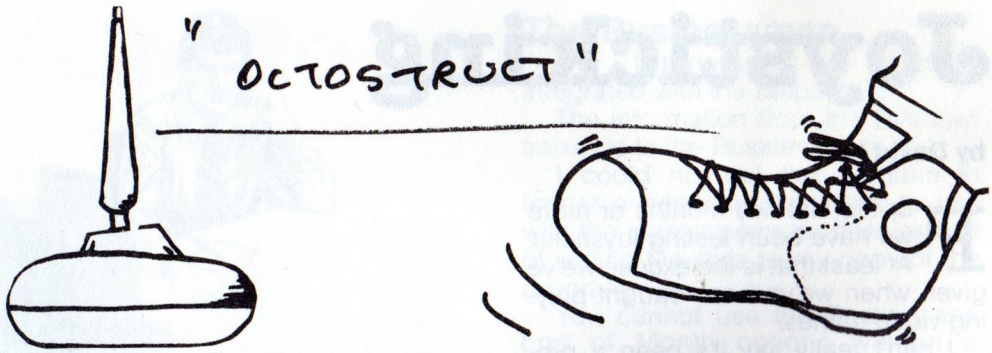
After all, three joysticks per month is obviously a world record.

Not having the gall to confront the long suffering staff of Computer Spot, I tried some running repairs. Armed

with a gallon (4-5 litres) of epoxy I did my best.

It wasn't good enough. Broken in both body and spirit, the poor Pointmaster was consigned to an early grave.

At this point the infinitely trusting but obviously eccentric Computer-Wave came to the aid of the party. They lent me some joysticks for testing.



No failures yet

Let me immediately report that so far none have failed the Octostruct.

This is possibly because the change-over time between sticks has been very short, although at least one was used for over five days and never looked like succumbing.

However, the Ladder Challenge really sorted them out and of the five tested only two managed to pass.

The others caused the poor little superhero to slide around the ladder as if it was a greasy pole, before taking the ultimate plunge. I'd begun to believe that I was to blame and that I'd lost what little skill I'd had.

Of the two that passed, the **Super Champ JC250** only scraped through by the skin of its trigger.

The JC250 is a novel contraption in that its umbilical winds back into its belly. By rotating the handle the 10ft cable is neatly stored in the base of the joystick.

And this, I think, is its problem. The handle of the stick tends to rotate while it is being used, making it somewhat less than accurate. This is a pity: if Champ Electronics (the manufacturer) had put a simple lock into the handle, to stop it rotating when it shouldn't, I am sure it would have passed more easily.

The **Pro 3000** suffered from a similar problem. It has a rubber bush at the base of the handle (which no doubt prevented the Octostruct demolition) that tends to throw the handle off-centre. It also has three fire buttons, two in the handle and one right across the front of the base.

In my not-so-humble opinion this is too many. Because it is designed to be held in the hand, one held it in the hand. And one's fingers kept finding a blasted fire button even when one didn't damn-well want one.

I'd rather not say too much about **Superjoy 18** except that it is very cheap, only a little more than $12, and that to my mind it is better value than the Commodore stick (costing about the same price).

Mind you, the fact that it is better value than the Commodore product is no reccomendation.

Using the Superjoy I couldn't even complete the Ladder Challenge on speed 4.

Finally the **Wico Command Control** and I got together.

It was love at first sight.

It does nearly everything I ask of it (and when it doesn't I know it's my fault). It sailed through the Ladder Challenge and my eight-year-old son has gone into a permanent sulk because one look at the steel shaft told him he'd met his match. The Wico really is a top joystick.

But no doubt you knew that all along.

Commodore Joystick

Looking after your Commodore

by Gareth Powell

Taking care of your Commodore can be summed up in four basic rules: keep it cool, keep it clean, keep magnets away from it and avoid static electricity. This is over-simplifying, of course, but if you follow these rules you will avoid most problems.

Keep it cool: Australia can be a hot place. Temperatures can get tropical. Computers do not like the heat, not even Commodores.

We have never had trouble with our Commodores through overheating but it worth bearing in mind that our offices are air-conditioned. We do know that any disks which are left lying where the sun can get at them have a distressing tendency to buckle and we have a feeling, but no proof to back it up, that the idle working temperature for the Commodore is much the same as a human being finds comfortable to work in. If you have trouble with your machine consider the possibility that it is getting hot and bothered. Try it in a cooler place and see if the trouble doesn't go away.

Keep it clean: Nothing damages disks and machines as fast as cigarette smoke. Don't smoke near your Commodore, and don't let anyone else do so.

Computer disks are platters of thin plastic covered with rust. Your everyday common or garden rust which is ruining your lawn mower.

We suffered from an immense number of disk problems in my office until we totally banned smoking. End of disk problems.

Some members of staff have since been sneaking a fag when I am out. Disk problems have started to come back. You cannot smoke near a computer – not even a Commodore – and expect it to perform properly.

Nor do these machines like dust.

When a computer is not in use it should be covered. In the United States they have a greater respect for computers than we have here, and machines are covered and disks are put into boxes when closing down after every session. Not in Australia. You see disks scattered on desks – and they are not even in their envelopes. From such dirty habits come disk crashes and data losses.

The head that picks up the information from the disk is very similar to a tape recorder head and should, in exactly the same manner, be cleaned. If you are using a cassette recorder for loading and down loading information we presume you are cleaning the recording and playback heads at regular intervals. Taking the Commodore disk drive apart to clean the head is not a recommended practice. Instead, use a cleaning disk of which there are several available.

Rexel sell one sealed in a package so you can be sure that after use you have an ultra clean head. How often you perform this chore depends on how often you use the disk drive on your machine. I do it weekly. You may only need to do it once a month. Same is true of cassette players. How frequently you need to clean the heads depends on the amount of use.

Keep magnets away from it: If you put a magnet near a cassette, cartridge or disk you wipe it clean. Indeed, this is the way many commercial companies recycle their disks after use. The biggest culprits in damaging disks are paperclips. In many cases these are kept in a plastic container that you shake like a salt cellar to get the clips to protrude through the top aperture. When you do this you are sending them up to a magnet and magnetising them. Place one of these clips on the top of a disk and the disk is ruined.

If you are using a computer and one of these paperclip dispensing devices, get rid of one or the other.

Avoid static electricity: Static electricity can bomb a computer out just like that. Being of a nocturnal habit, I do most of my writing at night. I used to wear a blue nylon dressing gown which I discarded when the static electricity it built up shot from me into my Commodore 4016 and zapped two chips. I actually saw the electricity and thought it was a modern version of St Elmo's fire. Quite eerie.

Since then I've become intensely interested in the problems of static electricity. Computer chips get sick and die at the sight of it.

A recent study showed that problems caused by static electricity made more computers break down than mechanical and power failures. Static electrity can be caused in many ways, but with personal computers the problem is, appropriately, people. Walking around you can build up a charge which will typically be around 7,000 volts but can get as high as 10,000 volts. The drier the day the more likely it is you will build up a large static charge. Humidity reduces the problem.

When you learn that anything over 200 volts can damage computer circuitry you realise the extent of the problem.

Static electricity can not only zap your chips so that the computer doesn't work, it can also change the information within a chip so that the problem may not show up for months afterwards. Static electricity can make circuits lose memory, affect data that you have entered and mess around with a printout.

Static electricity is to computers what the demon booze is to the brain. Only faster.

The damage occurs when you pick up a charge and earth yourself through the computer.

The bad effects of static electricity can be minimised during dry spells – the problem is not so bad when it is raining or there is high humidity – by not having your computer in a carpeted room, by making sure that the wiring is properly earthed and by

Continued on page 17

Education – Basic arithmetic and algebra

by Jenny Binstead

Jenny Binstead, BSc, who is a teacher in mathematics and computers from Cromer, NSW High School, takes a close look at a recent release from Computer Tutor.

Tutor is a five part tutorial on the basic skills necessary for the 2-Unit HSC Mathematics Course.

The language and examples used for the explanation are in keeping with the work necessary for the 2-Unit Course in general but some sections are not up to standard.

For example a number of equal signs are placed on the same line – a real no-no at the HSC level and not something to be emulated.

I could not recommend this program to gifted students wishing to brush up on their basics or get ahead of the rest of the class – they would rapidly become bored, even frustrated. Very poor students on the other hand would find this program rather difficult to handle – they need to have everything explained six different ways and there is no way that they can get this.

Students could be forgiven if they thought this was in fact a text book they were reading –only on a computer screen and a not very readable text book at that.

There is very little use made of the capabilities of the Commodore 64 – no graphics or music to capture and hold the interest of the student either as a reward or during the two minute loading period.

Two and you're out

The student is constantly reminded of the fact that two mistakes will send her/him back through the same section in order to relearn.

What is not pointed out is that pressing the shift and clear home keys not only clears the screen but is also regarded as a mathematical error.

Other keys pressed also count as mathematical errors and before too long you suddenly find yourself confronted with repeating the work you have just completed correctly. To any HSC student this is a waste of valuable time and cannot be afforded.

Compensating for errors

The student would very quickly learn to compensate for the kind of errors mentioned above by making sure as I did that they wrote down answers as they went along.

Then it is just a matter of keying in the correct response without even reading the section through again. It seems a rather useless exercise sending the student over the same work. One would have thought a change of examples at least would have been more appropriate.

Another disadvantage I found in one of the sections was the fact that a space left between terms in an answer was also counted as a mathematical error.

It is important that typing errors and mathematical errors are clearly distinguished in a program such as this.

SUMMARY:

The five parts are on two discs.

Part 1: The topics covered are:
(i) Integers
(ii) Rationals
(iii) Irrationals

The examples and the language used are good and the student would probably be quite impressed until making some of the errors mentioned above.

Part 2: Topics here are:
(i) Multiplication and division of fractions
(ii) Decimals
(iii) Related Topics-
*percentages
*repeating decimals
*approximations
*scientific notation

Again examples and language used are generally acceptable. There is a need to concentrate on some of the wording to get the meaning – this is a distraction to slow students who really need clear explanations of the concepts of the mathematics involved.

Part 3: In this section the following are covered:
(i) Surds
(ii) Algebra I – simplification
(iii) Algebra II – factorisation

Surds and Factorisation are quite well done but great care has to be taken to avoid the non-mathematical error of leaving spaces in answers to simplification problems.

Part 4: Topics covered:
(i) Algebra III – further factorisation
(ii) Algebra IV – algebraic fractions
(iii) Algebra v

In Algebra III the use of general terms with slow learners is not a good idea. They find the work far too difficult to understand if there are no real numbers in the examples.

Some provision should be made here for reference to a teacher when in difficulty. Doing the same examples several times over will only make them concerned and disgruntled.

Part 5: This part concludes with the following:
(i) Linear Equations
(ii) Quadratic Equations
(iii) Simultaneous Equations
(iv) Linear Inequalities

Topics (i), (ii), and (iii) were quite well done.

I was not very impressed with the actual teaching procedure used in the fourth topic – Inequalities. There were many rules given both for solving inequalities and absolute value. A large number of rules serve to confuse rather than help, particularly the weaker students.

This program, although it has many good points, would not be a worthwhile investment for a school.

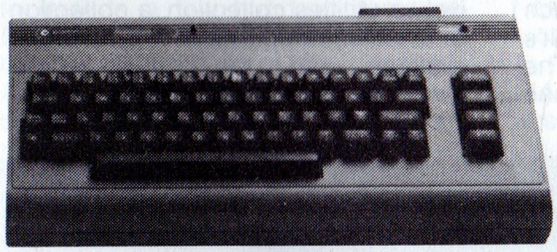
I feel that if it were a Public Domain program it might interest some teachers with sufficient programming experience to take the challenge and modify it to suit a particular school's or student's needs. □

Who's keeping up with Commodore?

The Commodore 64 is designed and priced so that you can pile on the peripherals. Like a disc drive, printer and even a printer plotter. You can own the Commodore 64 AND disc drive and printer for little more than you'd pay for many other computers alone.

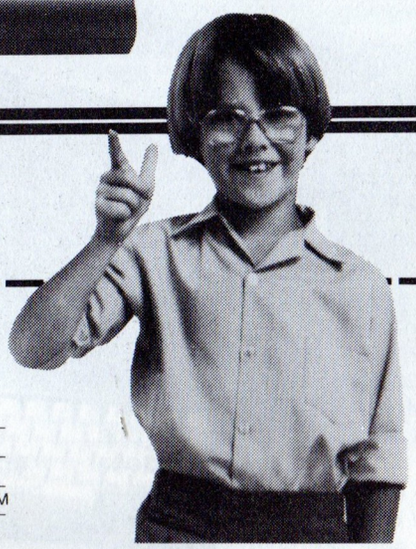
With 64K memory, outstanding colour graphics, a music synthesiser to rival the professionals, a fantastic range of games cartridges and full upper and lower case keyboard.

It's outstanding!



C= commodore COMPUTER

Keeping up with you.



Commodore Business Machines Pty. Ltd.
5 Orion Road, Lane Cove NSW. 2066. (02) 427 4888.
Please send me more information on the Commodore 64™

Name _____

Address _____

_____

BMS/CC157 Postcode _____ Phone _____ CM

Commodore 64 Software Development Kit

by M. Zolin

I only wanted an Assembler to play with as an alternative to BASIC and G-Pascal. I bought a cheapie but the documentation was so poor that it was next to useless. So I went shopping – by telephone.

The man said he had two packages that might interest me: the normal Commodore Assembler for three hours' pay and "a something about an inch thick that you might to look at" for four hours' pay. The "something an inch thick" comprised a blue, three-ring binder with a sticker saying:

COMMODORE
Public Domain Software
64 Software Development Kit
120100
DOS 2a Date 01/05/83

plus something like 25mm of photocopied paper and three diskettes. Being the sort of person who buys boxes of "sundries" at auctions, I bought the "inch thick" thing.

Six months later I'm still glad I bought it. It's taught me as much about CBM as about Assembler. It's taught me a lot about Assembler. The contents of the diskettes (see directory listings) will give an idea of the scope of the Kit.

What are the software and documentation like? In typical Commodore mode they are a combination of brilliant to excellent stuff marred by as much rubbish, false documentation and just plain non-existent documentation. Oh, I almost forgot, many of the programs don't work as supplied – as usual.

If CBM had decided to promote my Assembler education by making me take apart their programs to try and make them work, they could be accused of being formidable educational psychologists. Unfortunately it it more likely that we have a case of sadly typical CBM-SNAFU. Just as an example, two programs called up in the writing were missing from the diskettes. I got one of them later via the dealer but Commodore refuse to acknowledge that the other one is missing. What can you do?

The software

The three diskettes are respectively a utilities collection, a collection of ROM-independent Assembler routines and the official C-64 demonstration diskette.

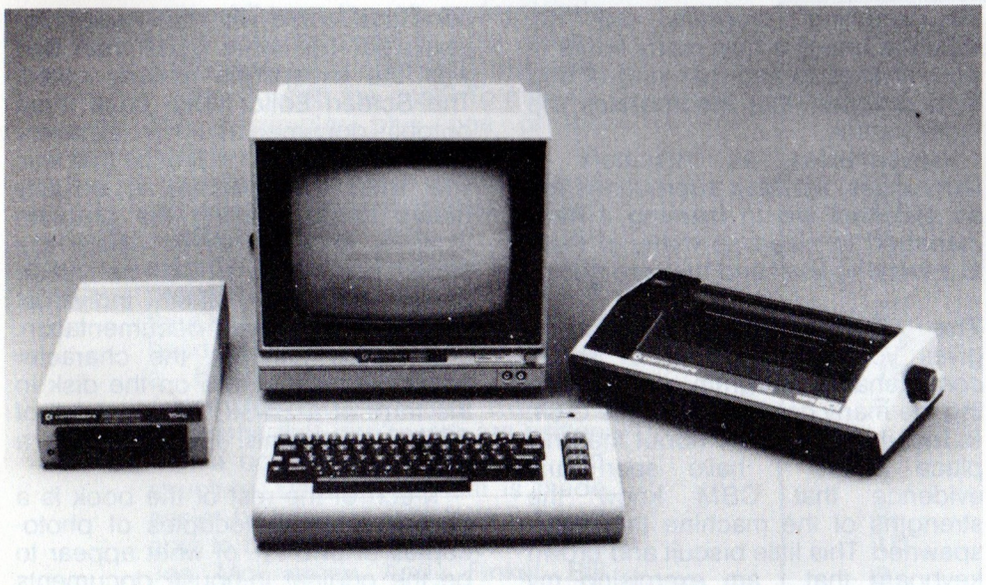
Commodore SX-64

The Utilities Diskette:- contains some six independent packages and a handful of unrelated bits.

The first ten programs are the standard Assembler package. The documentation is a photocopy of the writing that comes in the pretty covered pack that sells for much more. But is it any good? It is. The assembler has MACRO facilities and is quite forgiving of formatting variations. The editor has an antiquated 6-character upper limit on labels and operands. It has one of the older syntaxes but that's no real drawback. Only two of the claimed facilities fail to work. One is the fairly important "append" facility; the other is the possibility of saving a specified portion of a file on disk.

You can get around the second by deleting what you don't want to save and saving the rest. For the first, Commodore's Software Support Manager informed me that one should use the concatenation facilities. It makes for a messy source file and I for one have written my own appending program. But then I know the difference between appending and concatenation. As if to make up for not providing the promised facilities, the system errs the other way. It fails to document some available ones. For example, the ability to use "<" and ">" to extract the low byte and high byte respectively from a two byte parameter is not mentioned. The editor works, with the exceptions quoted. The two loaders work, the two monitors work. The monitors, though, would have the worst debugging facilities I have encountered in a commercial monitor. I seldom use them.

The next two programs, simply identified as "editor64" and its booter, I have never seen available separately and yet the documentation and software both indicate that they must be available in their own disk. In fact, as presented on the utilities disk they do not work and have to be moved about or reworked. It's worth the trouble, though. The editor is a very



Commodore 64, 1541 disk drive, MPS801 printer, 1701 colour monitor

powerful data base generator and display facility which can be used with BASIC programming. With the facilities of the former and the string handling ability of the latter one could do much worse.

The PET emulator would be a fun program on which to try out old, pre-C64 programs. It boots up well but on the one program of PET origin that I tried it, it sort of, well, shall we say, crapped out? The program was a BASIC loader for a machine language program so perhaps that's why. As yet I haven't had enough interest in it to follow it up.

Then we have a triplet of DOS programs. The "1541 backup" will copy a disk with a single drive. It's a little slow but it's brilliant. The others work, what more do you want?

An excellent collection of 10 programs comprising character and sprite editors follows. Both editors are very good and written in BASIC which allows a peep into the techniques used by the clever people. I was quite fascinated, for instance, at the ability to open a sequential file and then tell the DOS to call it a PRG file – quite successfully, mind you. One minor fault exists in the original Character Editor. When first "RUN", the characters have the same colour as the background and one has to fiddle with colour controls to get a sensible screen. The program being in BASIC, it was little trouble modifying it.

Documentation for these editors is adequate to good. I had no trouble driving them.

Unfortunately, as usual, not as much support is given to the SID chip and its facilities as is given the other bits of the C-64. Two keyboard music playing programs and a "monitor" comprise the support. One unexpected bonus was to find that the "PIANO KEYBOARD" program in the original C-64 User's Manual was available in the diskette. It is called "sound 11". A quick look at the listing showed why it didn't work back then when I laboriously keyed it in from the manual listing. Some idiot forgot to include a few hundred items of DATA in a couple of dozen more lines at the bottom of the program. About par for CBM: the day their demonstration programs work they may give even IBM a run for their money. Back to the sound programs, though. The SIDMON program I hated when I first saw it. It embodies all that is worst in display design in its dingy screen. Ergonomics was not invented when that program was written. With time and use, though, I have come to appreciate its strengths. Of a bunch of poor sound editors in my keep, it has turned out to have the most facilities. I still hate that screen though.

The rest of the diskette comprises a few ragtag booters and "pretty" programs and the terminal emulator (modem driver) "64 term".

The Max Subroutines Diskette:- was a bit of a puzzle at first. I had never heard of the MAX machines (more formally known as the ULTIMAX) which I presume is CBM's answer to either the dedicated home video games machine or the arcade monster crusher. Reading heaps of blurred print, I have concluded that the Max and the C-64 share some features: the former being a sort of empty skeleton of the latter. It's what the 64 is in GAME configuration. The significance of this to the Assembler programmer is that the Max machine has no Kernal and no Charrom and, of course, no BASIC translator but has the VIC and SID chips and some of the I/O gear. Therefore, the Max programs are all written to be self sufficient and not dependent on system ROM routines. That opens up the whole 64K of RAM to play with and allows programs of truly heroic scope and extent.

What one pays for this freedom, of course, is that one has to take care of all sorts of irksome little things like

```
0  "c64 utility       " 56 2a
1     "dos wedge64"          prg
4     "dos 5.1"              prg
7     "ass.editor64"         prg
39    "assembler64"          prg
6     "crossref64"           prg
4     "loloader64"           prg
4     "hiloader64"           prg
17    "monitor$8000"         prg
17    "monitor$c000"         prg
1     "ass.boot all"         prg
11    "Editor64 Demo"        prg
16    "editor64"             prg
1     "PET emu boot"         prg
17    "emulator"             prg
28    "1541 backup"          prg
3     "change 1541"          prg
11    "copy/64.v1"           prg
3     "sprite boot"          prg
42    "sprite editor"        prg
2     "scroll.data"          prg
3     "sample sprites"       prg
4     "char boot"            prg
32    "char editor"          prg
32    "char instr."          prg
1     "rotate.data"          prg
9     "standard.set"         prg
9     "computer.set 5"       prg
1     "colour test"          prg
11    "64 term"              prg
1     "64 print"             prg
9     "sound11"              prg
10    "sound/ring mod"       prg
18    "sidmon"               prg
1     "dos boot"             prg
2     "smooth scrolling"     prg
287  blocks free.
```

initialization on power up and nasty little mathematical routines. The Max Subroutines diskette has all those programs already written. Very good, you say. Here's the catch, though, we've forgotten the CBM habit of fouling up good things: many of the routines don't work. Some do, mind you. They are there to hook you into making the others work. This is where CBM educational psychology would shine if we didn't know better. Let's face it, I program for my daily bread, butter and occasional bikkie. I have no great desire to spend hours locating the instruction that some nice person left off.

With most of the Max routines there are notes in the paperwork giving a rundown on program capabilities, requirements etc. In most cases the notes carry a listing of the Assembler language programs. In many cases, the listing actually corresponds to what is on the disk. There is a good mathematics routines section which, by the copyright date, would appear to be the original MOS Technology routine for the 65xx family. After a while you realise that it won't assemble because the names by which files are called are not the names they bear in the directory. See what I mean? Stupid, frustrating CBM muddled thinking.

Oh, I almost forgot, most of the example programs don't work. When you get them to work there are some excellent sprite control programs, sprite modification programs, collision detection programs of great depth and excellence, screen scrolling programs, interrupt handling programs to enable 16 or 32 sprites to be displayed AT ONE TIME on the screen (at one time, of course, means in one frame) and so on. I must quote one example which has given me no end of fun. Anyone who has played Jeff Minter's "Attack of the Mutant Camels" will be familiar with the centre track of a screen scrolling while top and bottom remain stationary. The routine for doing that is in this diskette. And it works IMMEDIATELY AFTER FIRST ASSEMBLY but not subsequently. Magic being precluded from serious – ie non adventure – programs, one can only conclude that the fossil footsteps of the defunct Assembler have conditioned some critical memory location which the author of the scrolling program did not initialize. Imagine how many hours of work it takes to find that kind of bug in a program that incorporates the faulty routine.

Nevertheless, as indicators of techniques, the Max subroutines are an excellent aid to learning. I have managed to clean up many of them to a working level and they are good.

The Demonstration Diskette:- gives you half an hour or so of concatenated program demonstrating the many capabilities of the C-64. In fact it seems to be about the only place where I have seen any evidence that CBM know the strengths of the machine they have spawned. This little biscuit and brown keyboard that I am exercising my fingertips on hides more power than even its purveyors seem to be aware of. I am led to suppose that the demo disk is both an occasion to show off and an opportunity for other programmers to see some of the better techniques available on the machine. Certainly I have used it as such. It's a goldmine of techniques of which I have hardly scratched the surface.

The documentation

Much of what needs to be said about the documentation has already been said. It's a mixture of good, adequate and bad. The good and adequate are acceptable, the bad should not be there. The majority of the computer industry seems to be plagued with the problem of bad documentation and this is understandable. A gun programmer is not necessarily a reasonable writer and seldom interested in being one. It's fairly obvious that CBM and others have attempted a solution to this by separating the two functions. What that has got us is someone programming who doesn't write the instructions and someone writing the instructions who doesn't know programming. The solution to the problem was found long ago by the engineering industry. One wonders how long the computer industry will take to find the answer. Let's hope it's before they have fouled their next so much that time passes them by and the great popular interest is lost.

As has been mentioned, the Assembler System documentation is a photocopy of the official instructions that come with the Assembler package. It is more good than bad with the exceptions already noted. The Screen Editor (data base input output) documentation is excellent once you decode the first bit that tells you what the program is actually meant to do. Making the program work is another matter. Much the same comments could be made about all the other individual programs and their documentation. Just to be different, the character editor instructions are on the disk in the form of a BASIC program full of PRINT statements. I suppose it's a change.

Much of the rest of the book is a collection of photocopies of photocopies of pho . . . of what appear to be the original in-house documents on the development of the various chips that now are the C-64. As such they probably have archaeological significance but their place in a manual of this nature escapes me. They are also bloody hard to read.

A bunch of pages would be of particular interest if they were accurate which they do not appear to be. They are titled "Commodore 64 Memory Maps" with no less illustrious name than Jim Butterfield claimed as their compiler. They purport to give the memory location in ROM of the various routines used by the operating system. A disassembly of the routines would be nice – even useful – but is not there. The more disturbing factor is that routines used in known published programs do not correspond in address to these memory maps. A "sys" command to the addresses given often results in a hung-up processor.

Conclusion

So that's the package, a combination of the superb and the unforgivable. The latter, of course, is what one remembers most. It would reflect handsomely in Commodore's (and others') standing if they came to realise this. One comment needs to be made. The Kit is obviously directed towards the professional user rather than the dilettante. As such, an amount of knowledge and experience could be assumed of their consumers by Commodore. One could, then, forgive some of the documentation drawbacks. By the

same criterion, though, professionals don't have the time to play around debugging bad commercial software.

Was it worth the cost? To me, yes. The Assembler plus one of the editors (character or sprite, for instance) would make up the price of the whole kit. The rest comes for free.

What will it do? If you are intending to write machine language programs involving the capabilities of the C-64 system you will save a lot of head work with this kit. I can't think of a better kit to teach yourself Assembler programming. Even the programs that don't work are written with a commendable elegance and structure. Debugging them is made easier by this and, let's face it, being able to better a program authored by Joe McEnerney, Andy Finkel, Bill Hindorf and the other big guns is good for the soul. □

Our Kit from Maxwell Office Equipment,
162 Nicholson St, Abbotsford, Vic.

```
0  "c64 demo          " 03 2a
2      "boot"              prg
45     "demo fin"          prg
2      "boot2"             prg
23     "sprites"           prg
7      "hufo"              prg
1      "scrol"             prg
2      "boot3"             prg
80     "demo13"            prg
17     "demo.c000"         prg
57     "demo.guts1"        prg
3      "sprite.data"bin    prg
3      "bounce"m/c         prg
2      "boot4"             prg
57     "music prg"         prg
2      "music player"      prg
29     "boot6"             prg
4      "music"             prg
27     "c64/rev3"          prg
18     "Key"               prg
2      "boot5"             prg
28     "bt"                prg
80     "13"                prg
45     "boot fin"          prg
69     "nuclear demo"      prg
47     "byts and bites"    prg
5      "bytsprites"        prg
7  blocks free.
```

```
0  "max subroutines "  99 2a
26     "flips.src"1003     seq
17     "utopr"1002         seq
3      "max-emu.p3"        seq
17     "irqsvc.src"1005    seq
21     "irqsvc.exa"1005    seq
5      "numbers.bin"       prg
25     "scox"              seq
24     "dproc"             seq
13     "chrset1.src"       seq
12     "rchrset1.src"      seq
7      "setptr.src"2001    seq
1      "c64-math"          seq
3      "c64m-def"          seq
8      "addsub"            seq
10     "muldiv"            seq
2      "sqrt"              seq
2      "tanx"              seq
14     "utilit"            seq
13     "poly"              seq
5      "Konst"             seq
3      "log10x"            seq
3      "mvsx.src"1001      seq
13     "unimvx.src"1001    seq
15     "unimvx2.src"1001   seq
4      "sponx.src"1001     seq
4      "sonxx.src"1001     seq
3      "mvsy.src"1001      seq
17     "app1001ex1.src"    seq
18     "app1001ex2.src"    seq
7      "random.src"3005    seq
10     "app3005ex.src"     seq
10     "Kbd.src"4001       seq
12     "app4001ex.src"     seq
6      "paddle.src"4002    seq
8      "app4002ex.basic"   prg
6      "joy.src"4003       seq
4      "app4003ex.basic"   prg
8      "clock.src"4007     seq
36     "app4007ex.src"     seq
19     "poc.src"4009       seq
11     "bang.s"1004        seq
11     "cbang.s"1004       seq
7      "cord.s"1004        seq
14     "ckback.s"1004      seq
1      "elib"1004          seq
8      "adec"1004          seq
41     "example"1004       seq
8      "ebang"1004         seq
5      "ecord"1004         seq
2      "sdata"1004         seq
122  blocks free.
```

Learning your two times table – educational maths packages from DLM

by Scott Wilcox

From all the talk amongst the major software house, it appears DLM has set a new standard for educational software. Personally, I think they still have a few things to learn, although they stand alone when it comes to graphics and packaging.

The DLM series is comprised of six separate software packages, each selling for around $39.95. These include: four covering the main mathematical functions (+,-,x,/) and two which contain a mixture of either multiplication and division or addition and subtraction.

Viewed from a distance, the DLM packages virtually jump off the shelf from the drab surroundings of other less originally packaged programs. Each box is well decorated with one of those animated plastic pictures which moves depending on what angle you're looking at it.

Catchy stuff in a world where software houses are spending more and more on packaging in a bid to capture part of the growing software market. However, perhaps it's whizz-bang stuff like this that contributes to the rather high price of these programs.

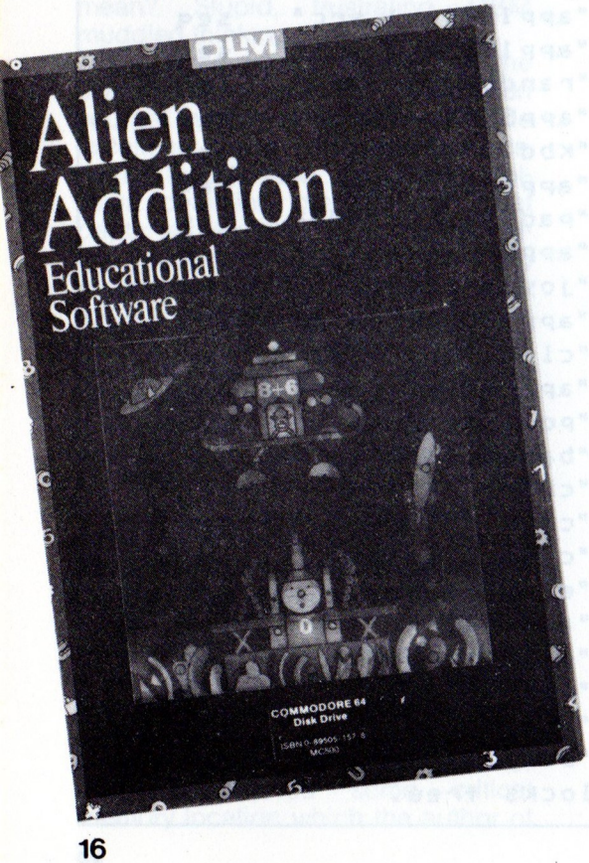
Loading is a fairly simple matter, as explained by the accompanying manual. My disk drive made a fair number of bangs and bops before things finally started happening. Once past the title page the graphics really went to work.

Making a game out of learning is nothing new, but this time someone really has done the game part right. Marauding crocodiles, spaceships, asteroids and huge dragons traverse the screen to entertain and teach the fundamentals of maths.

Any child, or adult for that matter, would be instantly rapt in most of what appears on the screen. Plenty of colour and detail make these very entertaining programs (although they are a little jerky in the animation area). This means that the child's interest will be sustained for a whole lot longer than if there was just a string of numbers floating around the place.

One area which appeared a little confusing was the number of controls available at any one time. Considering these programs are aimed at younger age groups, this feature may be more of a minus than plus. On average around 24 different keys could be pressed at any time. Perhaps they could sell it as a typing tutor as well?

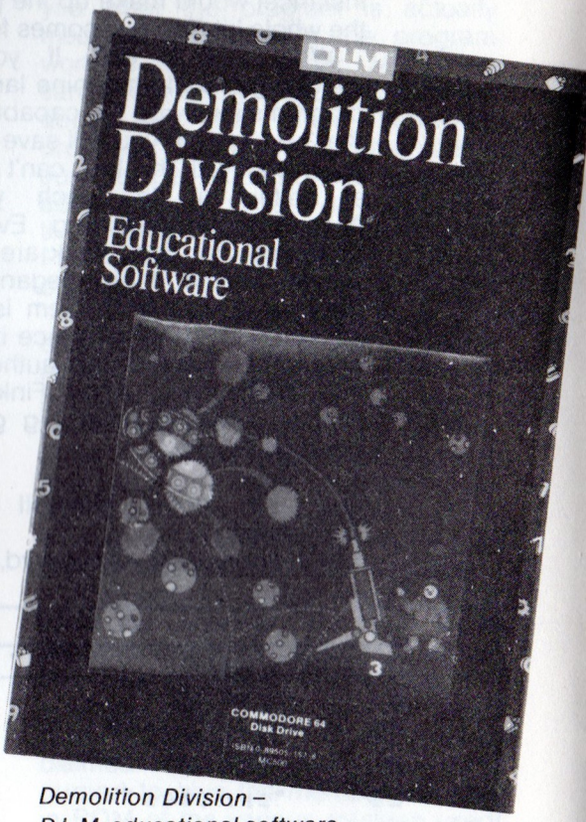
Alien Addition – D.L.M. educational software



Alien Addition and Minus Mission

Space Invaders, eat your heart out . . . well, almost. In Alien Addition you are armed with an answer to one of several deadly maths questions, descending from above in alien spaceships. Line up the right questions, fire and presto, the alien returns to the top of the screen with a new question. The going gets tough as the aliens get close enough to destroy your ship.

Minus Mission is almost identical to Alien Addition, except you have to shoot the green slime that descends, and the questions are all subtraction. The graphics in this program are very colourful and will no doubt keep a child's attention for as long as the slime holds together.



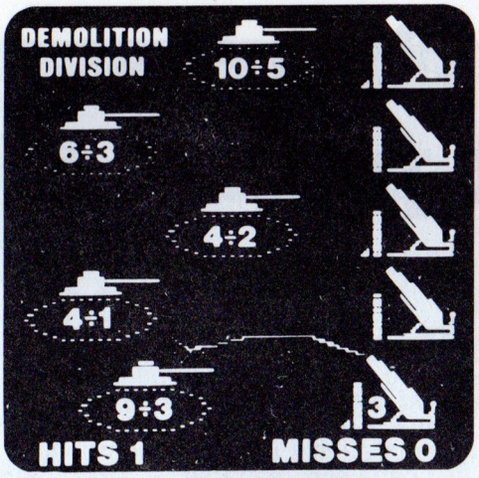
Demolition Division – D.L.M. educational software

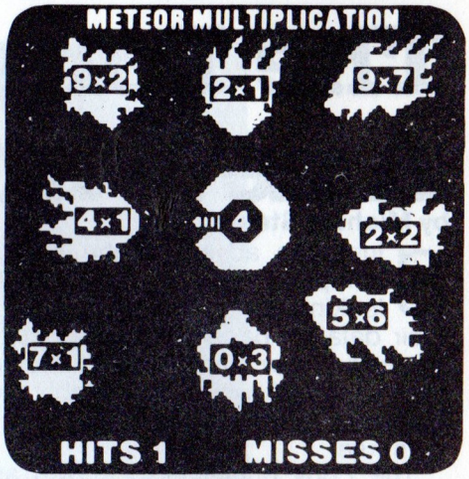
Meteors and tanks

Meteor Multiplication is the most disappointing of the DLM series, as the graphics lack the professional touch of all the others. Controlling the gun turret also proved to be very difficult.

Once more you are pitted against a barrage of questions, this time in the form of meteors closing in on you at the centre of everything. Associating the right answer with the right question fires a deadly blast, destroying the offending meteor.

In the division program you must destroy the advancing tanks as they move from left to right across the screen. You have five guns in line with the tanks to fire your answers,





D.L.M.'s screen display

only this time you get some indication of whether you're too high or too low.

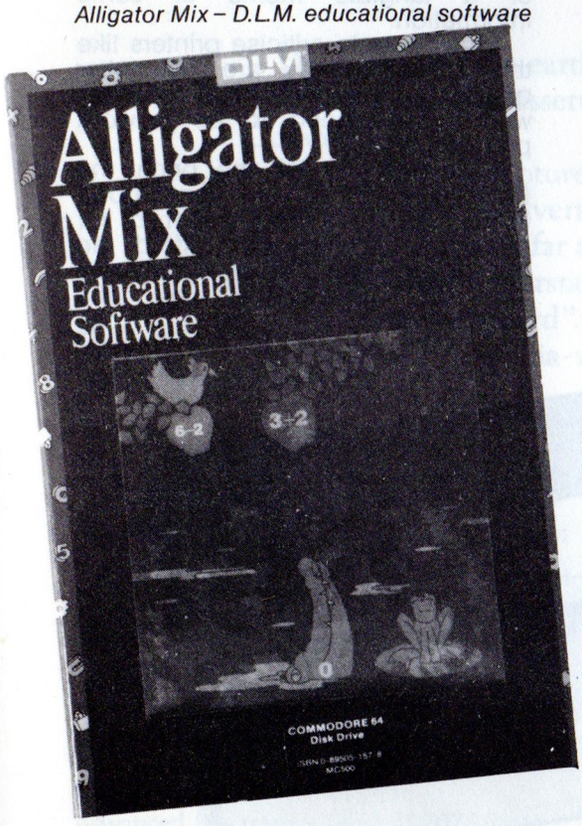
An answer above the correct result will send the gunshot firing past the tank, whilst a low answer will fall short. If you are wrong, the tanks advance a few spaces until they are close enough to blow your gun to bits after demolishing your protecting wall.

Dragon Mix

In this program, the questions are either division or multiplication. You are a large green dragon and on your belly you have the answer to one of the questions on the advancing space ships. As your tongue moves up and down you select the appropriate question and spurt a destructive ball of fire.

The graphics are certainly of better standard than in the single function programs, but there are only three space ships trying to destroy your city. Not very hard to lose because there are not enough questions to choose from. You only need use one key in this game, which will make it very easy to use, just right for small children.

Alligator Mix – D.L.M. educational software



Alligator Mix

By far the best of all the DLM software, a terrific concept with superb graphics. As in Dragon Mix, you only need to use one key to control the game. Apples containing addition and subtraction questions move fairly quickly towards you, the alligator.

On your side is an answer. You must decide whether to eat the apple or to let it bounce away. As you get the questions correct, another alligator surfaces above you. When he completely surfaces the alligator that was eating goes to sleep and the new alligator eats the apples instead.

Each new alligator makes the program a little harder as they are always a little bit closer to where the apples enter, meaning you must make up your mind faster.

Conclusions

I love the concept of the DLM series, although some of the programs seem to be lacking in areas. DLM may have made a mistake in having a different program for each maths function, as this does not represent value for money at $39.95 each.

But most importantly, the series fills a gap which has existed for some time and fills it quite successfully at that. □

looking after your Commodore
continued from page 9

not wearing clothes made from artificial fabrics.

Far better is to use an anti-static mat.

Rexel import one which costs something under a hundred dollars. This is certainly a good investment in an office where a computer breakdown can leave you floundering at considerable cost in time and money. You touch the mat before starting and tap it every now and again to get rid of any static build up. The mat has an earth wire which has to have a good earth connection, not always an easy task in Australian buildings. Rexel (Wilson Jones) also have an anti-static kit which includes a solution you spray on to discharge static build-up and a cloth which can be used for wiping surfaces clean without building up a charge.

Dirt, magnetism and static electricity are a computer's worst friends. Keep away from them. Remember that your mother told you never to mix with strangers. □

The Pony Express
A new hot printer

by Michael Stead

Since the advent of the microcomputer, users have been looking for viable ways to put their data on paper. One product of relatively new technology is the thermal printer, which is probably the cheapest printer for a small computer.

The Pony printer (also known as the Alphacom 42) is made by Alphacom, and is distributed in Australia by Ocswell International. It operates using thermal technology, rather than dot matrix, to give a quiet, good quality print at a fairly fast 80 cps.

It is a forty column printer, and is capable of handling Commodore graphics, as well as standard ASCII, and other modes of print. The whole unit is only 27cm wide, 19 cm deep and 10 cm high, with a separate power supply.

The Pony comes complete with a manual, a roll of paper, and a power supply. Together with the interface, this is all you need to get started. The interface also has its own manual, which contains the relevant information for the particular computer.

Separate interface

The Pony and the interface are sold separately, so that many types of computers can run the Pony.

The interface clips firmly into the back of the printer. A cable from the back of the interface goes to the serial port at the back of the computer where the disk drive plugs in, (or into the back of the disk drive if you are using the Pony and the drive at the same time).

The Pony will operate with both the Commodore 64 and the Vic 20 with complete compatibility, due to the complex design of the interface.

The Pony can also be used with many other types of computers (Atari, Apple etc) with the purchase of another interface for around $30.

Some printers seem to make enough noise to wake the dead, and this is where the Pony really shines.

When operating, it makes a quiet buzz, and even then, most of the noise is in the motors moving the paper and the carriage return at the end of each line.

Running my dot matrix next to the Pony proved it. I couldn't even hear the Pony for the racket of my dot matrix.

Reasonable quality

The quality of the print is very good. Admittedly, it isn't as good as some of the dot matrix printers, and isn't of a letter quality standard, but it is certainly readable – good for program listings, or any other purpose that doesn't require a high quality print.

One of my major criticisms of the Pony is the width of the paper. It is only 11 centimetres wide (or a bit over 4 inches for those not bitten by the metric bug). Also, because it is a thermal printer, it requires special paper as well. To buy the paper costs about $5 a roll, which may prove a bit expensive in the long run.

(Publisher's note. We didn't supply other versions of the printer for review. Worth knowing – it is available in other sizes up to 132 columns which should cover almost any eventuality.)

The Pony also has other features, such as condensed height, upper/lower case, ASCII character set, bit mapped graphics, reset etc. To keep in with the Commodore way of things, these features are accessed by the secondary address when a file is opened (eg OPEN 1,4,7 puts you into lower case). These features can be combined to produce various combinations.

The printer can also print in reverse, upper and lower case, with pagination (printing a "page" of text, and leaving 6 lines at the bottom), does line feeds, form feed and other types of printing features through control codes.

The paper can easily be torn off, as the plastic housing of the printer is saw-edged. The housing colour is a mixture between the colour of a 64 and a VIC, with a transparent plastic cover over the paper, and two buttons for "power" and "paper advance".

Lack of light

There is no light to indicate that the printer is operational. Possibly this could be overcome by a small light to indicate power on. Test print can be activated by pressing both the power and paper advance buttons.

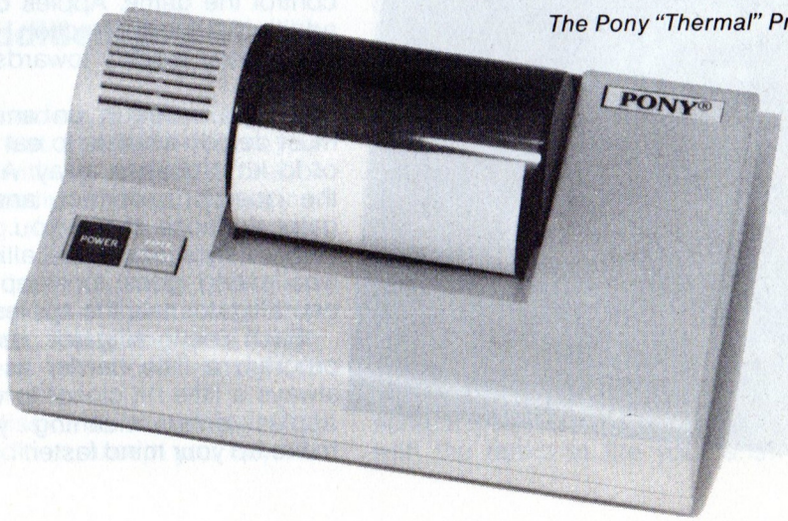
What uses would the average person have for the Pony?

For programmers to list out a program, or to list out something similar – recipes, charts, notes, homework: the list is endless.

It is a great help to programming to have a hard copy of your program, or a tangible memo of some information.

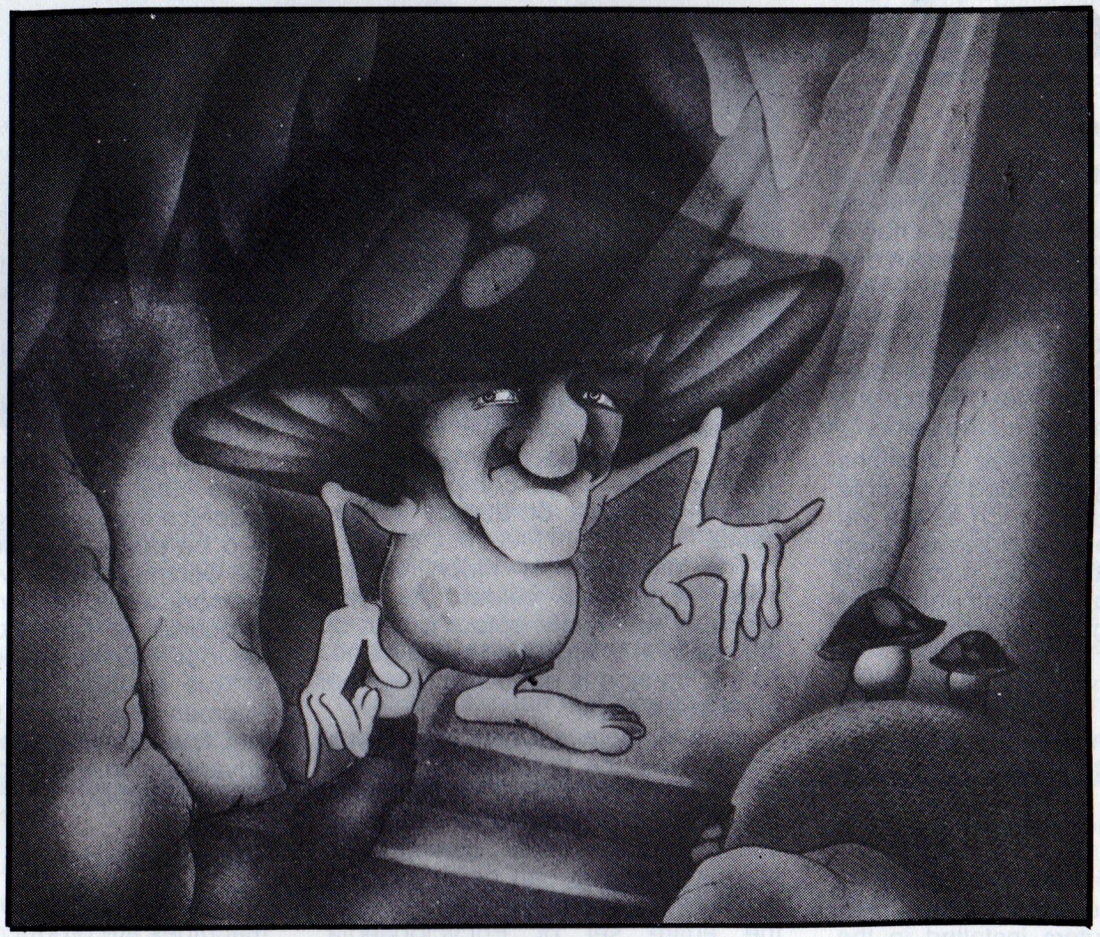
Many people criticise printers like this because they are not letter quality, but they forget that the Pony was not designed to be a word processor.

I believe that for the price, the Pony printer is a very good investment for a programmer, or for someone that cannot afford an expensive printer. The disadvantages are far outweighed by the advantages. □



The Pony "Thermal" Printer

Underworld of Kyn©



Deep beneath the earth's surface lies another world, the **Underworld of Kyn.** The first complete cassette or disk text adventure written specifically for the Commodore 64.

You have been captured by the evil wizard of Kyn, who has left you lost deep within the maze of caverns and passages. Can you evade his immense powers and escape to the surface far above?

Underworld understands complete intelligible sentences, such as "Kill the Bunyip using the sword".

Distributed Australia-wide by Ozisoft for $24.95 cassette and $29.95 disk.

Written and designed by Andrew Farrell

Available from the following dealers:

Toyworld, selected Grace Bros and Retravision stores,
Angus and Robertson, Chandlers Electronics.
High Technology (Vic) 596 7130
Computer Spot (NSW) 235 2971
C.W. Electronics (Qld) 397 0888
Abraxas Computers (SA) 223 5133
Vic West (WA) 445 2152
Computerworld (NT) 81 7438
Advanced Electronics (Tas) 31 7075

PRIME ARTIFAX

Video games

by Andrew Farrell

Undoubtedly most of us have, at one time or another, ventured into the dark depths of a noise-filled video arcade somewhere in one of Australia's fair cities. Today, a few are taking on a new image, with flashy colourful furnishings and plenty of lights to expose undercover agents working for the CIA, but the basic idea has remained the same.

On the surface these machines may seem totally unrelated to humble personal computers, such as the Vic 20 or Commodore 64. However, upon closer examination, you may notice remotely similar attachments and fittings adorning the innards of each machine.

At heart, they are computers, dedicated to the single job of producing stunning graphics and sound along with the occasional lost twenty cents.

What is the future of these monstrosities?

Why can't we all have graphics and sound like that on our own personal computers installed in the comfort of our own homes?

The answer is, you almost can.

There are a few obvious differences between what is available now on a home computer and what is in the arcades, including the cost.

However, if we look at what both machines can do, you'll notice some very close similarities.

Similarities

For a start, both use some form of controller such as a joystick or paddle to let the machine know what you are doing. Home computers have a keyboard, but there is no reason to have one on a games machine being used in an arcade.

Let there be light . . . sound . . . and then, there was television. Every good video machine also has something attached to it that looks remarkably like the common television. Technically speaking, it is virtually a television without the "tele".

Closer to home, it is more like the Commodore monitor you may have sitting in front of you. Once again, a dedicated piece of equipment, capable of producing pictures of a little higher quality than your average telly.

Joining the dots

Some even work differently from your telly. Instead of drawing what dots have got to be drawn, it may literally join the dots. This type of monitor uses a technique known as vector graphics. In recent times the Vectrex home video computer introduced this type of display to the home.

The difference is much clearer lines, with less colour. As you probably well know, colour is part of the essence of any good video game, so this method is not being used too often these days.

Instead, there is a new technology which rivals the graphics of virtually any computer. Video disks. Vast depositories of information. Memory that you have never dreamed of. Memory that you can never fill. One single shiny platter that can contain the entire set of Encyclopaedia Brittanica.

But does it make for better games? Possibly, just possibly. At present, groups of fans are falling all over each other to play this new breed of games.

I have played these "amazing" cartoon-like games too and I found it similar to driving a tank through a marble factory. The machine simply did not respond in the way I wanted when I wanted.

Having played it for several days and mastered the huge selection of two or three moves available on each screen, I've decided that I would prefer to watch Tom and Jerry.

If these aren't the next generation of games, then what are?

The clue lies in the amount of memory available on a video disk.

Unlimited memory

Imagine an adventure game with an almost unlimited amount of information accessible from disk at any time. The mind boggles at the range of possibilities.

The other area where video machines have the edge on home computers lies in the video chip.

Memory is a costly commodity when it's used 256K at a time. Some video games use several 256K blocks to store data for the massive landscape visible at various parts of the game.

Anyone who has ever played Xevious will know all about the scrolling background which seems to last into eternity.

Most personal computers use little more than 8K to store Hi-Res graphics thus allowing far less detail – to be precise, 32 times less. Even the Commodore 64, with a full 64K of memory, would be hard pressed to fit a quarter of one video game like Xevious, into the memory.

True enough, old favourites like Pacman and Space Invaders will fit just fine. However, there is still one advantage in the original machines. The video chip.

Video chips

All the graphics on these arcade machines are produced by one if not several video chips. These chips are quite similar to the Vic II chip inside the Commodore 64, but a lot more powerful.

What does a video chip do?

On Galaxions, have you ever counted the number of aliens on the screen at any one time?

There's more than a handful. These aliens are usually sprites, produced by the video chip.

A sprite is a small object capable of being moved around the screen by simply designing the shape and giving it an X and Y coordinate. Sprites pass under or over each other, they let the computer know when they are colliding and who they are colliding with.

Great to work with, especially if you are a game programmer like me.

If you can imagine you've got 64 of these little MOBS (movable objects) flying around the screen you can understand that it certainly helps to make the game a little more interesting.

Some personal computers are appearing with this feature and, as a result, the quality of computer games

Video Village – Bondi Junction

has improved.

The Commodore 64 is capable of moving eight of these MOBS around at any one time and with a bit of stretching it is possible to manage a few more.

As you can see, video machines have a major advantage over the 64 with around 50 extra objects to play with.

Cut down versions

When it comes to games, personal computers are just cut down versions of full blown arcade machines. If the costs of vital parts drop, features previously unheard of on small computers will start becoming the standard.

Which will mean that eventually, in theory, the home computer will equal arcade games in quality, graphics and general design. Except that it is unlikely that this will ever happen. Because as computer games catch up with the arcade games of today, so the designers are moving forward with the arcade games of tomorrow with video interfaces, eyeball sensor controls – look at a command and it will be implemented – and a full range of realistic voices.

For personal computers it is a no-win situation, as it is unlikely they will ever catch up.

On the other hand, arcade games make lousy word processors.

In many ways, I see the 64 as one of the few computers for under $500 which has the chance of catching up part of the way. The quality of games now available on it are a fine example of its potential in this area. Check out International Football if you want to see what really can be done. □

Speaking of software

"Say it again, Sam . . ."

by Andrew Farrell

"Hello, I am SAM, the Software Automatic Mouth for the Commodore 64 Computer." I'm beginning to hear voices. From my computer. But hang on a minute, there's nothing sticking out the back. There must be, otherwise I am hearing things. The job is beginning to get to me.

Where's the cartridge?

To put it bluntly, there ain't one.

SAM is a fully software controlled voice synthesiser which speaks through the speaker in your television set. It works in much the same way that a music program for the Commodore 64 works, producing sounds which are parts of speech instead of individual notes. These sounds are run together to make speech in the same way that the notes are run together to make music.

The difference is that SAM is a whole lot more complex when it comes down to the nitty gritty and has the ability to produce sounds almost equivalent to the human voice.

How near is "equivalent" is open to a lot of debate. Bob Drew – not a man noted for his classical musical tastes – is extremely scornful on the subject.

I consider it is just as acceptable as the hardware voice synthesiser. It is a matter which only you can decide because it is so subjective.

The producers of SAM offer a very comprehensive package. Each package contains several programs for use with SAM and an excellent 36 page manual which explains the entire operation of SAM from BASIC and Machine Code.

Two inputs

Once loaded, SAM asks if you require the reciter, which adds the ability to SAY words, and after loading it, if required, enters BASIC. You now have an additional ten commands to control various areas of SAM's voice.

SAY is the first of these, and commands SAM to "say" the following text or variable.

Other commands include PITCH, SPEED, LIGHT, RECITER, and KNOBS.

SAM accepts two types of input, complete words and phonetics. In this way, simple sentences can easily be constructed or, for increased quality, each word can be individually defined.

Using various combinations of the commands, you can design your own voice and, in phonetic mode, the actual inflection on each word.

The demonstration program included with SAM gives some good examples of the variations possible, with everything from a tiny elf to a big giant or, in a parallel mode, from a junior editor to a senior publisher.

Versatile

SAM is very versatile and I think this is its best feature. Understandably, the quality is not quite up to that of the Commodore Voice Synthesiser.

Each word may sound a little scratchy or jagged, but then again each word could be any one of several hundred spoken variations.

"Though I speak with the voice of electronic angels . . ."

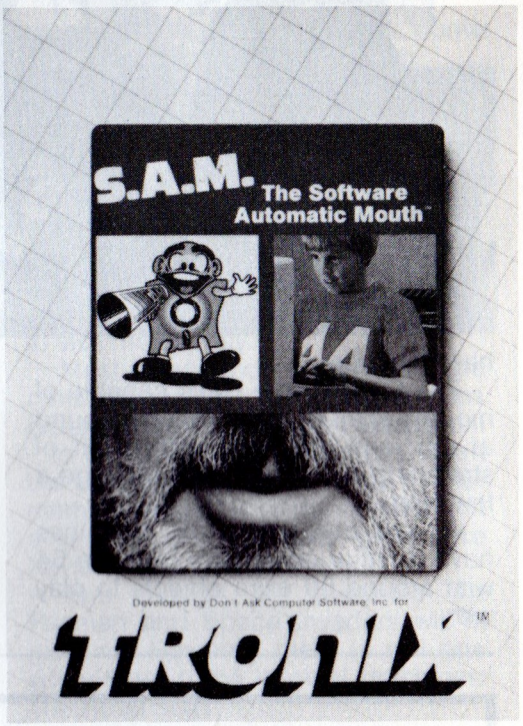
Hardware synthesisers provide better enunciation but allow far less room for modifying the inflection or pitch. It is a case of "What you lose on the swings you gain on the roundabouts".

If you are not into programming, and to my continued amazement not everyone is, then the hardware synthesiser is the way to go.

At the moment it has its limitations but it will undoubtedly improve and expand as time goes on. Certainly, in the near future – a year or so – we can expect to see a hardware speech synthesiser built into every Commodore 64 and, eventually, into every personal computer.

SAM's vocabulary, unlike that of a hardware voice synthesiser, is virtually unlimited.

With the RECITER loaded, complete English words may be entered, which are then converted by SAM to phonetics and spoken.



The Software Automatic Mouth – S.A.M.

In most instances the results are fairly good. However, the odd word may cause some confusion.

In most cases it sounds like a ventriloquist who has got past the first few lessons of the "Happy Home Ventriloquist Learning Kit" but is not yet ready for a public performace. He keeps saying "gottle of gear" instead of "bottle of beer" which is understandable, just, but nowhere near perfection.

Using phonetic input to make SAM say "Hello, I am Sam" we enter:

SAY"/HEH3LOW, AY$ AEM SAE$M." (return)

The result is a lot more understandable, although rather clumsy in the making.

Now it sounds like a ventriloquist who is good enough for a country RSL prawn night but not for an appearance on the Bert Newton Show.

You may have noticed a few extra symbols hidden amongst the phonetics in the above example, but first, what's a phonetic?

I've heard several renditions of

what this strange mouthful really is and none of them make too much sense. Well, here's my version. Pretend for just a moment that you can't spell. (In my case I won't have to pretend.)

Now try writing some familiar words down, keeping in mind we are being very simplistic and enunciating each word beyond normal speech.

For example the word "Hello" might come out as "Heylow" depending on where we come from. Another example – the word "Computer" could be written as "Kumpyuwter".

English is a wonderfully varied and complex language and there is no such thing as a standard pronunciation, despite the little men in the back rooms of the ABC laying down the law for annmouncers. Even the publisher's Welsh accent is acceptable. But only just.

Go back to the SAY command mentioned earlier for "Hello". By removing the special numbers and slashes you'll find it is made up of simple sounds spelt how they sound. Put together they make up a bunch of letters that a computer can understand much better than normal text.

By adding a few numbers along the line we can control areas such as stress and pitch. A single phrase may have a different meaning when various words are stressed more than others.

For example, when Shirley Bassey sang, "What now my love?" the whole meaning of the song could be changed by emphasising the second word.

Rude words

Now that hopefully it's all making a little more sense, you're probably asking the question "What can I use it for?"

Well, you'll probably just have a lot of fun writing programs to say rude things to friends as they enter your room.

Educational programs will find many uses for it as the authors of SAM soon discovered, and who promptly produced a spelling program which says each word for you to spell.

Sam won't be compatible with existing software although "Don't Ask", the American company behind it all, have promised to release several programs which use its features.

Overall, SAM is one of the best speech synthesisers I have heard. In software it manages to achieve almost what a hardware synthesiser does without restricting you to 250 words. With Sam, if you really want to, you can program it to say "Llanfairpwyllgwyngyllgolgerygwyn drobwlchllantysilioggogogoch". But only if you really want to.

At $69.95 it's good value for money. SAM is distributed in Australia by Imagineering and is available from most Commodore dealers. □

Vic 20 Hints
Make your own space creature

by Sean Mcsullea

Many Vic-20 owners are unaware that their machine is capable of producing BIT ADDRESSABLE CHARACTER GRAPHICS. (A fancy term to describe the ability to change any existing character to any shape that the user may choose). For example, the letter [A] may be changed to an alien for the purpose of a space game.

This is done by changing the memory locations that tell the computer where to look to find the values that define the characters. This location is called the character set pointer.

Characters are defined by use of bit patterns. As you may or may not know, 255 is the maximum value that may be poked into any memory location. In each memory location is a possible combination of eight bits. These 8-bits all combine to produce a number between 0 and 255. The value of the first bit in the bit pattern is 1, the second 2, the third 4, the fourth 8 and the fifth 16 and so on till the eighth and last bit. Thus the binary number 10101001 would be 1+0+0+8+32+0+128, 169.

For example the letter [A] would look like this.

Image	Binary	Value
...**...	00011000	24
..*..*..	00100100	36
.*....*.	01000010	66
.******.	01111110	126
.*....*.	01000010	66
.*....*.	01000010	66
.*....*.	01000010	66
.....	00000000	0

The built in character memory in the Vic lies in memory location 32768. In order to tell the Vic that you want your own character set you must first change the character set pointers to the location of your characters. Putting a value of 255 in location 36869 will tell the computer that you now want the character memory to lie between locations 7168 and 7680.

Let's try a little example that may make things clearer.

```
10 rem space creature
20 poke 52,28:rem Tell basic not to use the area your characters will be in.
30 fort=7168to 7679:poket,peek (t+25600):NEXTT
35 rem copy all characters into new location (32768-33279 to 7168-7679)
40 poke 36869, 255:rem change character pointer to point to location 7168
50 fort=7167to7167+8:readd:poket, d:next:?"[clr/hme]":poke 7900,0
55 rem read data into memory and place an @ sign in middle of screen
60 Poke 38620,0:END
70 data 14,60,90,126,189,24,36,66
```

The character created is a space creature. Notice now that whenever you press the [A] key the creature will appear.

This is how the creature was constructed.

Image	Binary	Value
...**...	00011000	14
..****..	00111100	60
.*.**.*.	01011010	90
.******.	01111110	126
*.****.*	10111101	189
...**...	00011000	24
..*..*..	00100100	36
.*....*.	01000010	66

Experiment a little, you may be able to make animation by showing your characters in a sequence.

Using the joystick

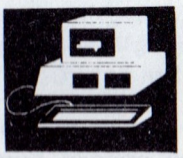
It is possible to read the joystick from within your own=programs and perhaps make those games a bit easier.

Type:

```
10 DIM JS(2,2):POKE 37139,0:DD=36154:PA=37137:PB=37152:REM SET UP VARIABLES
20 FORI=0TO2:FORJ=0TO2:READJS,(J,I):NEXTJ,I:READ DATA INTO ARRAY
25 DATA -23,-22,-21,-1,0,1,21,22,23
30 GOSUB 9000:PRINTJS(X+1,Y+1):GOTO 30 :REM READ AND PRINT JOYSTICK DIRECTIONS
9000 POKE DD,127:S3=-((PEEK(PB)AND 128)=0):POKEDD,255 :REM JOYSTICK READ ROUTINE
9010 P=PEEK (PA):S1=-((PAND8)=0):S2=((PAND16)=0):S0=((PAND4)=0)
9020 FR=-((PAND32)=0):X=S2+S3:Y=S0+S1:RETURN
```

A joystick has only five switches inside it – up, down, left, right and fire. A diagonal reading is when two switches are depressed. Eg, pressing the joystick to the upper right corner will light the right and the top switch and return a diagonal reading.

To read the joystick we use memory locations 37137 and 37152. But it is not that easy. To use the joystick we must first make a little routine which you can add to your program.



Lines 10 to 25 need only be executed at the start of your program and gosubing line 9000 at any stage will return the joystick values. The variables s0, s1, s2, s3, and fr will normally be set to 0 but will return values of 1 or –1 if joystick is in their direction. S1 stands for switch 1 and the switches are in the following order.

```
          Switch 0
           (S0)
            .
            .
Switch 2 . . . . Switch 3
(S2)        .      (S3)
            .
          Switch 1
           (S1)
```

Remember, I would be interested in answering your questions next month and I trust I have left you with enough to keep you busy until then.

PS. Try writing a program that moves your alien around the screen using the joystick. □

How to write your own adventure game – Part II

by Andrew Farrell

Adventure games are proving to be big sellers world wide as more and more people discover the fun of being tortured by their own curiosity, discovering the undiscovered and going where no man has gone before. This month we put our knowledge about text files to work with a short Editor for use in designing our own adventure game.

Text files, ring a bell? If it doesn't you'd better grab last month's issue for a quick refresh. Meanwhile, the rest of us will get on with putting them to work.

As I mentioned early in the piece, for this series, we will be using text files to store our room descriptions and other important information. As long as we don't get too carried away, this should prove to be a fairly adequate means of doing things.

Anyone with ideas about having more than 30 rooms should reconsider. Likewise for anyone thinking of writing more than two sentences on each room. Unless of course you don't mind waiting for your datasette or 1541 to read the files.

What we really need is a word processor, to simply edit our de-

```
ready.

1 rem description editor
2 rem
3 rem by andrew farrell
4 rem
10 poke53280,0:poke53281,0:poke53272,23:print"
{CLR}{LT BLU}":goto1000
12 poke198,0
14 getr$:ifr$=""then14
16 return
20 ifed=0then1$="":l=1:
25 ifed=1thenprintl$;:l=len(l$)+1
30 poke204,0:poke207,0:gosub14
35 poke204,255:poke207,255:r=asc(r$):
40 ifr=20andl>1thenprintr$;:l=l-1:l$=left$(l$,len(l$)-1):goto30
50 ifr<31orr>96then60
55 printr$;:l$=l$+r$:l=l+1:ifl<255then30
58 goto80
60 ifr<193orr>218then70
65 goto55
70 ifr=13thenprint" ":return
79 goto30
90 rem display rooms
92 forg=1torm-1:print"
{CLR}Room "g"{CUR DN}":prints$(g):print:printr$(g):gosub12:print:next
94 print"
{CUR DN}{CUR DN} [ Press Any Key ]":gosub12
96 goto100
100 print"
{CLR}EDIT Mode : Press F1 to list rooms."
110 print"
{CUR DN} Room "rm " File is "n" bytes long.{CUR DN}"
125 print" Press return to enter this room."
130 print"
{CUR DN} s - Save  l- Load  e - Edit a room"
135 gosub12:ifr$=chr$(13)orr$="n"then160
```

scriptions and save them. If you've already got one, great. Most users won't, so I thought we should slip something in to do the job.

The following program will allow you to enter your descriptions and save them to disk or cassette for use later on. A short menu will be displayed after the opening title page and from there you can go ahead and use the various options.

To start entering your descriptions press any key except those used for the other options. The Editor will expect you to make two entries. A brief description of the room, perhaps two or three words long and a full description a couple of lines long.

During entry, you may use the delete key to erase any mistakes you might make. None of the other editing keys will have any effect. Once a room has been entered the main menu will reappear.

Pressing "e" will allow you to edit a previous entry as if you had just typed it in and were about to press return. Function 1 will display all room descriptions which have been entered. "s" will select save and "l" will select load.

When saving your file the file name defaults to descriptions. If you are a little choosy or you would like to be able to enter your own file name the variables are set in lines 320, 405, 410, 520 and 610.

One interesting feature is that the program automatically decides if you are using disk or cassette. The command PEEK (186) returns the current device number, which will usually be 1 or 8.

A few notes about the notations used in the listing. All the cursor commands and screen controls appear in brackets as the corresponding key abbreviation. For example:

(CLR) means press shift and CLR home.

(CUR DN) means press the cursor down key.

(LT BLU) means press the commodore key and BLU.

Well, I'll leave you to it. Next month we can read our descriptions back, combined with our map and get the show on the road. By the way, be sure to type your descriptions into the editor in the right order. Room "A" would be room 1 on the editor and room "J" would be room 10. You may have up to 40 rooms . . . but don't get lost.

```
140 ifr$="e"thenprint"
{CLR} Room to edit (1-40) ";:gosub20:goto200
150 ifr$="§"thengosub90:n9=0:goto100
152 ifr$="s"then300
154 ifr$="l"then500
160 rem
170 print:ifed=1thenl$=s$(rm)
172 gosub20:s$(rm)=l$
175 print:ifed=1thenl$=r$(rm)
177 gosub20:r$(rm)=l$:ifed=1thened=0:rm=y
180 n=n+len(s$(rm))+len(r$(rm))
190 ifrm<40thenrm=rm+1:goto100
199 print"file full":goto100
200 z=val(l$):ifz=0orz>40thenbp=1:goto100
210 y=rm-1:rm=z:ed=1:goto170
300 rem save a file
305 print"
{CLR} Saving{CUR DN}"
310 dv=peek(186):ifdv=8then400
320 open1,1,1,"descriptions"
330 print#1,rm
340 fori=1torm:print#1,s$(i)
350 print#1,r$(i)
360 nexti:close1:goto100
400 rem save file to disk
405 open 15,8,15,"s:descriptions"
407 close15
410 open 1,8,2,"descriptions,s,w"
420 goto330
500 rem read a file
505 print"
{CLR} Loading{CUR DN}":n=0
510 dv=peek(186):ifdv=8then600
520 open 1,1,0,"descriptions"
530 input#1,rm:fori=1torm-1
540 input#1,s$(i):n=n+len(s$(i))
550 input#1,r$(i):n=n+len(r$(i))
560 nexti:close1:goto100
600 rem read from disk
610 open 1,8,2,"descriptions,s,r"
620 goto530
1000 print"
{CLR} Adventure Game Description Editor"
1010 print"
{CUR DN} (c)1984  Andrew Farrell"    ew issue 3"
1020 print"
{CUR DN} This Version 5/4/84"
1030 print"
{CUR DN} [ press any key ]":gosub12
1035 rm=1:n=0:dims$(40),r$(40)
1040 goto100

ready.
```

Principles of a good game

by Scott the Wilcox

Scott has had his 64 for some time now and his favourite occupation is pulling other people's games to bits to see how they work. Here, he discusses some of what he's learned . . . and maybe one day we'll even see a brilliant game from him.

By now you'll have seen many games on your 64 and will have undoubtedly passed judgement on all of them. Unfortunately many look as though they were written by headless earthworms, yet, there is a handful of brilliant games that make their authors a whole lot of money.

Why?

Why do the same people keep on churning out games that are unique?

Some hints can be found in their graphics, scrolling, sound, speed, controls, the amount of screens or levels and the way the game catches the player's interest and sustains it.

Graphics

To start with, there are three basic forms of graphics on the 64.

Sprites, redefined or standard graphics characters (PCGs) and bit mapped or high resolution graphics, none of which are supported to any great extent from BASIC.

In all the better games for the 64, the sprites don't look like sprites.

They are so well designed that you cannot tell that they are, in fact, just a plain old sprite. Sadly, sprites in most programs are easily spotted.

A good example of this is the difference between the sprites in "Cops and Robbers" and the sprites in "Revenge of the Mutant Camels" (I might incriminate myself if I tell you which is the bad one.)

(At the time of writing "Revenge of the Mutant Camels" was not yet commercially available. It is an excellent game and as soon as copies are in the shops a full review will ensue. Ed.)

When animating sprites, it is easier to change the sprite data location pointer, than to turn on sprite one, turn off sprite one, turn on sprite two. It may appear trivial but there is a big difference.

Of course, other small things can throw the effect of a sprite right out the window. These things include its colour, size, (expanded or unexpanded) and position in relation to other objects on the screen. An example of good sprite positioning is in "Robin to the Rescue".

PCGs

For those of you who don't know, programmable characters are simply characters such as 'A' or '@' which have been remodelled, much the same way as sprites.

For example you can turn the letter 'G' into a little man, and have him walk across the screen as easy as moving the letter 'G' across the screen.

PCGs, in good games, are usually used for things such as ladders, bricks and anything that doesn't have to move.

To give any program a professional look, the character set is usually changed from the usual boring old set, to an original design which gives a polished appearance.

In all the better programs on the market for the 64, most do use their own character set. For example in all Jeff Minter's games, he uses his own character set, which enhances all his programs greatly.

If you are desperate you might only use the graphics symbols on the keyboard of the 64. For some things, some characters are quite useful.

Sound

For a game to be truly polished there must be sound effects and/or music. What would it be like to run across the screen without the well known pitter patter that features in most games?

Where would we be without a long and drawn out death tune? I cannot imagine such an unrealistic game.

Using the 64 to its full capability, we would presumably have sound effects in voice one, a tune in voice two and the accompaniment to the tune in voice three.

That type of set up is used in "Revenge of the Mutant Camels". Very catchy music it is, too.

Scrolling

Many games are too big for the

ATTACK of the Mutant Camels

size of your monitor and combat this with scrolling. By scrolling I mean when everything moves up, down, left or right and some of the screen disappears, but new information is brought on.

The screen can be made to scroll very smoothly, as in "Skrambler", but in some games the scrolling is jerky and not very realistic at all.

This is because the information on the screen is probably made up of PCGs and as we know (small piece of intellectual snobbery here as I pretend every one has read the Commodore manual), all characters are eight dots wide by eight dots high. Therefore, when scrolling a character or many characters, they must move in lots of eight. This gives a jerky look to the movement.

In the games that scroll smoothly, the whole screen is scrolled a dot at a time, so that everything on the screen moves, even PCGs producing a more realistic effect.

Interest rates

It is no good having great graphics and sound if there is no interest in the idea of the game itself.

Too many games are frustrating to play and lack meaning and purpose.

The game must keep you wanting to play it by making the events of the game dependent on what the player does, not what the computer wants to do.

In other words make the movement of everything apparently random except for the symbol representing the player.

Another way of sustaining interest is to provide many different levels of play.

For example, the greatest number of levels I have seen is in "Revenge of the Mutant Camels". It packs 42 different screens, each with their own name, and each entirely different from the others.

(Yes, yes, Scott. You've got your point across. You like the game. Perhaps readers may care to telephone him at home late in the evening to enquire where he obtained his pre-release copy. Editor.)

The secret is to fool the player into thinking that each frame is separately set up.

The trick is to give the player the same frame several times, without him/her realising it. In some games the background is the same in each frame or it is made up of the same graphics shapes. (A la Jumpman.)

Sometimes, objects are randomly placed in the background and then scrolled across the screen whilst in the foreground all the action takes place.

Human interface

There is no mistaking that the keyboard or the game controllers are the human interface. (Interface between humans and computers.) Ideally, the way in which these are used in a game should be easy to operate. Reaction time is critical, so joysticks or paddles should be read frequently to see that the player isn't falling asleep on the job.

And the game must react quickly to movements of the paddles or joystick. This is of prime importance to the player as he may have to kill cannibals, stab dragons or dodge the odd bullet or two. How is this possible with a ship or man that moves so slowly it's nearly in reverse?

The most reactive game I have seen is "Jumpman". (Unfortunately for owners of the cassette players, Jumpman is at this moment only available on disk.) The game lets you select your "Run Speed" and this determines how fast you will go when you move the joystick. On run speed "One" the joystick becomes too reactive.

It seems obvious that a game without the right sensitivity on the controls will undoubtedly fail miserably.

Bad language

No matter what you think you'll never see bad language in a video arcade. They don't write Space Invaders in Basic. Instead, the speed of Machine language, which is not an easy language to grasp, is used. Its speed quickly makes up for the effort involved in learning it.

A comparison was made by a few of my friends on a Vic-20 a long time ago. They wrote a basic program to fill the screen with the letters of the alphabet, one after another. They also did the same in machine language.

In the time it took the basic program to fill the screen with the letter 'A' the machine language program had finished.

That's why all the best games are written in machine language, it provides the speed that you see in the arcades.

Ideas

One common way to get ideas is to go to a video arcade and see the latest games direct from the US and perhaps copy the idea, and produce the same game on the 64. Be careful about using the arcade names as they are usually copyrighted.

Get hopping

I hope these points will help turn you into a full time programmer. There is something that might help you along the way. A Graphics tool kit. Next month, I'll be having a look at what's available in this area, as well as pulling a few more games to pieces for you to look at. □

Basic tips

by Andrew Farrell

If you ever type any sort of program into your computer you will sooner or later get that terrible sinking feeling that comes when you make some damn fool mistake and the whole of your work goes down the gurgle. You can lose the result of several hours typing through some inadvertent POKE or SYS command. I make no claim to holding the world's record in losing typed-in programs, but I would certainly be up there with the top-ranking few.

Now, you and I can retrieve a program intact by simply loading and SYSing the following program. Having been told very severely by the publisher that there will be no "computer speak" in this magazine, let me explain what SYSing means.

You load the program using the command 'LOAD "Filename",1,1' if you are using a cassette and 'LOAD "Filename",8,1' if you are using a disk drive. Then type in the command SYS and then the start address of the relevant program which in this case happens to be 49152. Press Return.

Once you have entered the program and run it, use the BSAVE program, which appears elsewhere in this issue, to SAVE it as a machine code file. I suggest you save the program under the name "OLD" as it is virtually the opposite to the "NEW" command.

In the event of your system "hanging up" or "crashing" there is one small job to carry out first.

Some of us are lucky enough to own a reset switch which will do the job just fine.

For the rest of the nation, reach for the nearest paper clip and bend it as you will see fit in just a moment.

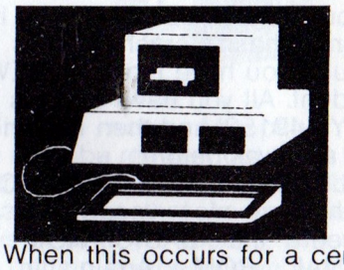
Now, stand up and lean over your computer so that you can see the back panel in the general vicinity of the USER port. (The one directly below the Commodore logo, next to the cassette port).

Still leaning over the machine but looking towards the front, starting at the right end of the port mentally number the first three pins from 1 to 3. Now the tricky bit. Using your trusty paper clip, short out pins one and three. In other words, connect pin 1 to pin 3 for about one second.

Simple stuff, no need to fret. Chances are highly remote that you could damage anything apart from your nerves.

If all else fails, drop around to your local computer store and have them either sell you a reset switch or demonstrate the many uses of a paper clip.

What you have actually done is grounded or earthed the RESET



line. When this occurs for a certain period of time (shorter than you'd care to imagine) the 64 goes off and does the next best thing to a cold start.

Cold start?

That's what happens when you switch your computer on.

There is one major difference between forcing a reset and turning your computer off and on.

Using the first method, the contents of memory are not lost, only rearranged slightly. However, the pointers which tell the computer where our program starts and ends are gone and that's where the OLD program comes into the picture.

OLD restores these pointers and relinks the BASIC line numbers so that everything is back to how it was those few brief moments before all was lost.

Now to the final step:

LOAD "OLD",8,1.

Type in SYS 49152 and bingo!

Your program is now ready to continue editing.

If the same nasty gremlin should

```
ready.


1 rem old.. basic loader
2 rem
3 rem by andrew farrell
4 rem
5 rem program sits at $c000..(49152)
6 rem
10 s=49152:
20 data 169 ,1,141,2,8,32,51,165,24,165,34,105,2,133,45,165,35,105,0,133,47,96
30 reada:ifa=999then50
40 pokes,a:s=s+1:q=q+a:goto30
50 ifq<>1658thenprint"data error":stop
60 data 999

ready.
```

cause a similar catastrophe later on, there's no need to reload "OLD" as long as you have not erased any part of it.

You can also use OLD even if the program hasn't crashed. It is very useful if you have typed in NEW by accident. All you have to do is type in 'SYS 49152' and then Return and all is as it was before.

OLD sits in the memory at $C000, which is a nice vacant 4K area between the BASIC ROM and KERNAL ROM. Again, in our constant efforts to demystify the whole process, we'll look at exactly what that means. In exactly the same way that a Street Directory says "63 G7" to show you where a street is in a suburb, so the memory has a map and $COOO is a place on that map. For simplicity's sake let us say that in the Memory is installed a language, BASIC and a series of instructions called the KERNAL. OLD is positioned between these in the same way as a computer shop is placed between the pub and the town hall.

I hope you lose less hair after typing in this program.

To save a byte or two

One of the smaller problems of owning a Commodore 64 is the lack of a built-in monitor.

What is a monitor?

Monitor is a word that can be used to mean a Visual Display Unit on which you view the results of your work, or someone else's if you are using a commercially produced program.

The monitor I am talking about is something altogether different. This monitor is actually a monitor of what goes on at the heart of your computer.

I'm talking about a machine code monitor.

This tool provides us with the ability to see what's where in memory as well as being able to modify things a little.

Hesware distributes a very nice monitor on cartridge which, for those of you interested, also has a one line Assembler. This program gives you the ability to SAVE any given block of memory, which is perhaps the most useful facility of this cartridge and others like it.

How, you ask, can I possibly use that?

For a start, you can save the OLD routine included elsewhere in this issue, as well being able to SAVE any other machine code program of which you know the start and end address.

Having typed in the program below and run it, you will see one of two things. The familiar READY sign will indicate all is well. If the words ENTRY ERROR appear, check your data statements and try again.

Then SAVE it, before you LOSE it. (Although the LOSE command is not part of Commodore BASIC, it is often used by programmers world-wide to cause much grinding of teeth and loss of skin.)

To use the BSAVE routine type in the following command:

SYS 52992, start address, end address, "filename", device (r)

What a mess. It almost looks like an instruction out of an Easy Script manual.

In human terms it means you must know the start address and end address as a decimal number (1, 2, 3, 4 . . . 65536), the filename and the device to which you wish to SAVE. For disk the device is 8 and for cassette it is 1.

Disk users may also include DOS commands as part of the filename such as "O:FRED" to save the file FRED to drive O.

BSAVE has a small drawback. You may not use a variable as the filename. For example you must not say SAVE N$. The filename must be a constant.

```
ready.


5000 rem block save routine
5005 rem
5006 rem by andrew farrell
5007 rem
5010 rem start 52992 - end 53066
5020 rem
5100 i=52992:
5110 reada:z=z+a:ifa=999then5130
5120 pokei,a:i=i+1:goto5110
5130 ifz<>9334thenprint"entry error"
5150 end
6000 data 165,43,133,87,165,44,133,88,165,45,133,89,165,46,133,90,32,253
6010 data 174,32,138,173,32,247,183,165,20,133,43,165,21,133,44,32,253
6020 data 174,32,138,173,32,247,183,234,165,20,133,45,165,21,133,46,32,25
6030 data 3,174,32,86,225,165,87,133,43,165,88,133,44,165,89,133,45,165
6040 data 90,133,46,96
6099 data 999

ready.
```

Getting a directory . . . without getting a directory.

Occasionally we all wish there was a second 64 nearby to reach across and load up a directory of our disk. It can be quite frustrating, until you realise that there is a program to get around it.

To use CATALOG simply type in the following program, SAVE it and run. Now type SYS 49152. Magic. Up pops the list of disk files and you didn't lose your program. If you wish to save the program as a machine code block to LOAD,8,1, later on, use the BSAVE program included elsewhere in this issue.

CATALOG starts at 49152 and continues for a further 190 bytes, give or take a few. (I always save a few bytes more just as a habit.) When you try it out, you'll also notice that colons appear before the file type. This allows you to just type LOAD before the file name and a ',8' after it and press return. There's no need to delete the file type which appears on the far right.

(Publisher's note: Accepting that there must be readers out there who understand what all this is about, and accepting that Andrew Farrell knows of what he speaks, why do I keep getting bogged down half way through?) □

```
ready.


1 rem : non-destructive directory listing * written by paul regan 5/5/84
2 rem : program resides in 177 bytes starting at 49152 ( $c000 )
3 rem : once loaded and run , sys49152 will display disk directory
4 rem : on the screen.
5 rem : any basic program in memory will not be disturbed
6 rem : holding <shift> down pauses the listing
7 rem : <run/stop> aborts the listing
8 rem :
10 for j= 0 to 176
20 read a :s=s+a
30 poke49152+j,a
40 next j
50 if s<> 24545 then print "error"
60 end
80 data 169,128,32,144,255,169,15,32,195
81 data 255,169,15,162,8,160,15,32,186
82 data 255,169,2,162,174,160,192,32,189
83 data 255,32,192,255,176,127,169,1,162
84 data 8,160,0,32,186,255,169,1,162
85 data 175,160,192,32,189,255,32,192,255
86 data 176,104,162,1,32,198,255,32,207
87 data 255,32,207,255,32,204,255,162,1
88 data 32,198,255,32,207,255,32,207,255
89 data 240,77,32,207,255,72,32,207,255
90 data 133,98,32,204,255,104,133,99,32
91 data 209,189,169,32,32,210,255,162,1
92 data 32,198,255,32,207,255,72,32,204
93 data 255,104,240,16,32,210,255,32,225
94 data 255,240,31,173,141,2,208,251,76
95 data 106,192,56,32,240,255,160,23,24
96 data 32,240,255,169,58,32,210,255,169
97 data 13,32,210,255,76,70,192,32,204
98 data 255,169,1,32,195,255,169,15,32
99 data 195,255,96,73,36,73

ready.
```

As a rat one hears rumours that do not sully the ears of other members of the animal world – like humans. One of the better rumours that has been making the rounds lately is that Commodore are going to launch a lap computer "any day now". Where that rumour started bewilders me. I was told that this machine was on display at the Hanover Show where every computer manufacturer was showing his wares. No, it wasn't – I checked with my German cousin, Herr Rattmeister, who said, nein, there was nicht lap computer at the show.

The Managing Director of Commodore in Australia swears in a Scots accent that he has never heard of the machine. And surely he would be the first to know.

Obviously stories about this machine are but an idle rummour spread around by rats working for other companies. Yet my Hong Kong cousin, Wan-chai Rat, says he has seen the machine. And that it works with a 3 inch disk drive. And that it has 132K of RAM memory. And that there is a word processor and a dictionary and a spread sheet built into to the ROM.

Plainly he has been drinking mao tai again, that alcoholic beverage which did so much for President Nixon. If the machine does not exist, then how can he have seen it?

A record?

In the current edition of the Guinness Book of Records there is a singular omission. Although I know it is not the role of a humble rat to correct a publishing company with the magnificent name of "Guinness Superlatives", yet I feel I must.

I have carefully checked the latest edition under the section man-made structures. Nowhere do I see any reference to the pile of machines awaiting service at Commodore Australia. Surely this is the largest man-made edifice in the world. It is undoubtedly bigger than the Aswan Dam, it dwarfs the Great Pyramid of Giza, it towers over the Great Colossus of Rhodes, it covers a wider area than the Rockefeller Centre.

Then why is it not listed as the greatest man-made edifice in the world? I am puzzled at the omission.

NZ game

I have a cousin in New Zealand – doesn't everyone? – who tells me that the biggest selling business machine in New Zealand is the Commodore 64. Can this be true, I ask myself? Apparently yes. Because the exchange rates in that country are such that even the Commodore 64 is something more than cheap, less than expensive.

What my Cousin Rat has yet to explain to me is whether the machines are used for games after the office is closed. If they are not it is sad to think of all that computer power going to waste. Perhaps we could think of a program that would be suitable. "Count the Sheep" or even "Escape to Australia". Both would, I am sure, satisfy different segments of the audience.

In a corner

It is with a sense of shame and sadness in my heart that I have to report that, even viewed from the lowly stance of a Rat, our publisher Gareth Powell has turned out to be a man of straw.

His excuses for ducking out of the editorship and handing it over to the young Andrew Farrell are, I suppose, just acceptable. But whatever happened to the post bag full of abusive letters that he received after the first issue. Why were they not published? Why have they not seen the light of day?

True, some of the remarks they contained would not have been suitable for a family magazine such as this. Anyway, I happen to know that his parents were married some weeks before he was born.

However, he might have published a selection instead of sulking off in a corner trying to pretend none of them had ever arrived. Surely, a computer publisher must have the courage of his convictions. And the publisher of this magazine has had enough convictions. And nearly always for the same crime.

As a lowly Rat I can merely hope that Andrew Farrell will bite the bullet and accept the abuse that flows his way as part of the job. And publish it to cheer up the readers.

SX 64

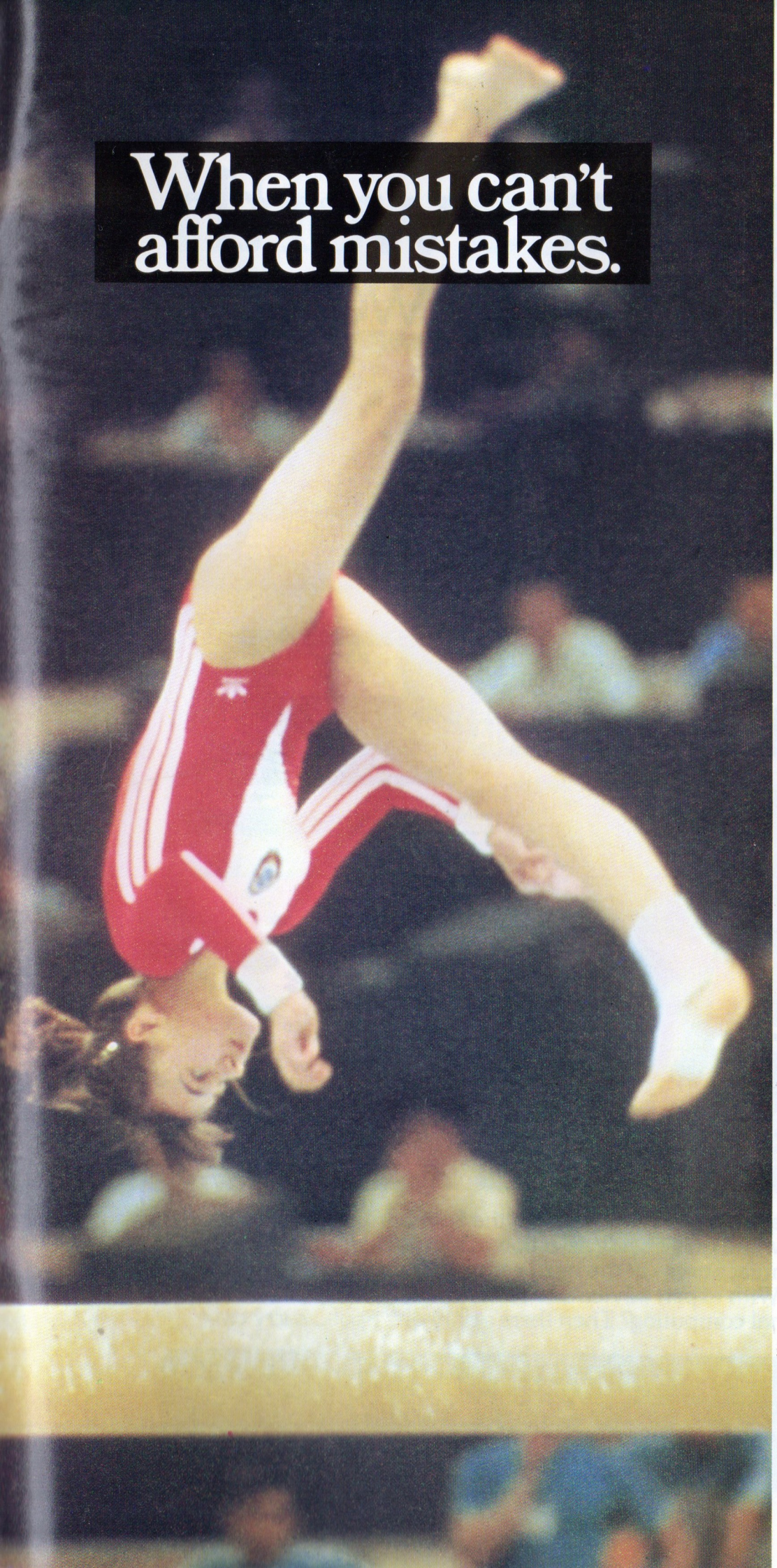
One hears there was pain and perturbation in Commodore Australia when the young and lovely Sue Robinson wrote a review of the SX 64 in the computer section of "The Australian" and went pooh-poohs all over that lovely machine. Word on the rat newtwork says that Scottish cries of anguish were heard from Commodore, which means that either the managing director was having trouble with his sporran again or he did not approve of the article.

And, in truth, she does appear to have got it all wrong. A fellow journalist Rat (true, nearly all journalists are rats, but not necessarily with a capital letter) says that young and lovely Sue didn't quite understand what the machine was for and kept asking plaintively why it wasn't more like the Microbee.

Perhaps someone should explain to the poor bewildered thing that these two machines are made by two different manufacturers in two different countries for two different purposes at two different prices. Or on second thoughts, perhaps not.

Endgame

I have an acquaintance. Not a Rat. Just an acquaintance. Who works for another computer company which shall be nameless – Apple. And I went to his home for a drink. (He is not a proud man and entertains Rats occasionally.) And his children were playing with a Vic 20. I asked him whether this was not a temporary aberration and he said – and this is an accurate and direct quote – "No, I'm a closet Commodore queen." □



When you can't afford mistakes.

There really is a difference in diskette brands.

It's in the way they are made.

Because the computer industry cannot afford variation in the quality of diskettes, at Nashua we looked for, and found a way to ensure absolute consistency.

Here's how we do it.

Quality Circles.

At Nashua we've found the best way to attain this 'consistency' is to ensure that at each stage of production our diskettes

are statistically checked to make sure the quality is 'built-in' every step of the way.

Rather than long production lines, we have 'Quality Circles' – small groups of people whose job it is to make sure that each Nashua diskette is right in the first place.

The result is a diskette with such consistency, that it is chosen by those people who can't afford mistakes.

Phone Sydney 958 2044, Melbourne 428 0501, Brisbane 369 4244, Adelaide 42 0021, Perth 328 1888, Hobart 34 3761, Darwin 81 6204.

Nashua

There really is a difference.

Ruwald & Skinner 400NA

Who's keeping up with Commodore?

The Commodore EXECUTIVE 64. A personal, portable computer with outstanding graphics, colour, music and astonishing computing capability, all in an easy-to-carry case.

The Commodore EXECUTIVE 64 is designed for the movers of this world. Designed to give you the power. Power at your fingertips. The power of 64K memory. The power to keep up. In the office. At home. Or in your home away from home.

commodore COMPUTER

Keeping up with you.

Commodore Business Machines Pty. Ltd.
5 Orion Road, Lane Cove NSW. 2066. (02) 427 4888.
Please send me more information on the Commodore Executive 64™

Name _____

Address _____

_____

BMS/CC 152 Postcode _____ Phone _____ CM