CURSOR

NEWSLETTER of the COMMODORE COMPUTER USERS GROUP (QLD) INC.

Registered by Australia Post - Publication No. QBG3958

AUGUST 1986 VOL.3 NO.2

CLUB ROOMS: Bardon Prof. Devel. Cnt. 390 Simpsons Rd. / Carwoola St. Bardon

CONTENTS

Diary For September	2
Regional Meetings	2
Librarian's Report	3
Editorial	5
Notes & Gossip	6
Price List	9
Financial Report 1985 - 1986	10
Financial Planning	11
From the Disk Librarian	12
Language Arts	13
Quo Vadis Commodore	14
The Threat	15
Cartoonist of the Month	15
The 64 000 Byte Question	16
Amiga Basic	17
Amiga Basic Sound and Screen	19
Easy Script and the 1526/802 Printer	21
July Quiz Solution	23
C-64 Trivia	24
Bytes	25
Weather Maps on the C-64	26
Letters to the Editor	27
Review: Mach 128	29
Directory	31

DIARY FOR SEPTEMBER

Group meeting on Tuesday, 2nd September 1986, in our Club Rooms in the Bardon Professional Development Centre, 390 Simpsons Road Bardon. Enter through Car Park in Carwoola Street and follow the lights. Doors open at 7pm (library). Meeting starts at 8pm sharp.

Demonstration of Vizawrite 128 & Vizastar 128

by Lex Hinckley & Greg Perry

--ooo--

Workshop meeting on Sunday, 14th September 1986, from 1pm till 5pm in the Guidance Officers Training Centre, Bayswater Street, Milton.
Bring your programming- or hardware problems, as well as your own computer equipment! Opportunity to copy the group's Public Domain Disks.
Coordinator: Colin Shipley.

PLEASE NOTE: Workshop Meetings are for MEMBERS ONLY!

--ooOoo--

REGIONAL MEETINGS

Cannon Hill meets on the 4th Saturday of the month (7.30pm) in the Cannon Hill State School.
Contact: Ron Jarvis (acting coordinator) - Ph.399 6981 a.h.

Kenmore meets on the 1st Sunday of the month (1pm - 5pm) in the Kenmore State School Library. NO PARKING in the school grounds!
Contacts: Peter Reeve - Ph.378 2265 a.h. / Keith Hadland - Ph.378 6698 a.h.

Pine Rivers meets on the 2nd Sunday of the month (1pm - 5pm) in the Strathpine High School (rear entrance).
Contact: Bruce Wylie - Ph.359 9779 a.h.

Redcliffe Peninsula meets on the 1st and 3rd Friday of the month (7pm) in the Clontarf High School.
Contact: Paul Janek - Ph.283 1663 a.h.

Sherwood meets on the 2nd Friday of the month (7.30pm) in the Graceville State School.
Contact: Leigh Winsor - Ph.379 2405 a.h. / Philip Parkin - Ph.378 5383 a.h.

The Gap meets on the 3rd Wednesday of the month (7.30pm) in The Gap State School.
Contact: Julianne Fallen - Ph.300 2982 a.h.

Wavell Heights meets on the 2nd Tuesday of the month (7.30pm) in the Wavell Heights High School (library), Brae St.
Contact: Robert Adamson - Ph.266 8353 a.h.

Maryborough/Hervey Bay meets on the 4th Monday of the month (7-10pm) in the Sunbury School in Alice St.
Contact: Terry Baade (16 Mouquet Lane, M'borough, 4650) at 21 2271 (w) or 21 5059 a.h.

SPECIAL INTEREST GROUPS

AMIGA Sub-Group meets in the Guidance Officers Training Centre - Bayswater Rd. - Milton on Sunday 31st August (1pm - 5pm)

Peter Wharton will demonstrate Deluxe Paint

Contacts: Steve McNamee - Ph.262 1127 / Darryl Godfrey - Ph.205 1983

Primary Education Sub-Group meets on the 3rd Tuesday of the month (7.30pm) in the Aspley State School.
Contact: Bill Weeks - Ph. 208 8620 (work) or 341 2823, a.h.

Programming Sub-Group meets on the 1st Tuesday of the month, (during main meeting - in our club rooms).
Contact: Jim Vick - Ph. 345 1878, or Tom Kelly - Ph. 277 9900

CP/M Sub-Group meets on the 1st Tuesday of the month, (during main meeting - in our club rooms).
Contact: Regan Russell - Ph. 848 1353 (a.h.)

Superbase Support Group:
Coordinator: Stan Seymour - Ph. 263 7210, a.h.
Programming Advisor: Hank Deucker - Ph. (075) 66 1317, a.h.

If you would like to start a sub-group in your suburb or district contact Terry Steer, Sub-Group coordinator for the C.C.U.G. (Q.) INC. for further details and advice.

--ooo--

LIBRARIAN'S REPORT - 8th August 1986

by Maurie Hawkyard

Library users might have noticed that despite ever increasing membership and consequent library usage, the all too familiar congestion in being served was if anything, somewhat easier at the August meeting.

This was due in no small part to the efforts of Graham Barlow who was there prior to 6.00pm energetically setting up the library area so that we were able to commence operations by around 6.45. Very many thanks Graham for your efforts.

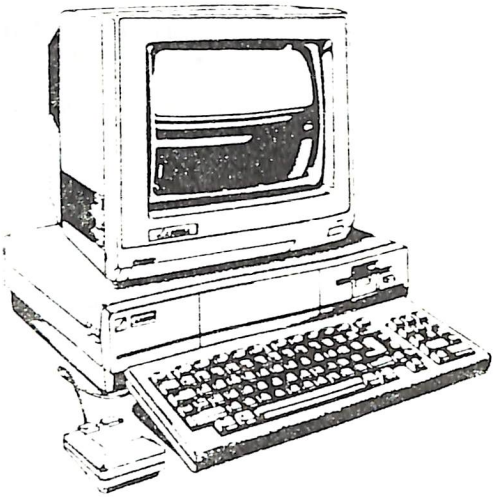
I have also increased the number of operators in each area so that there are upwards of a dozen members providing your library facilities on meeting nights. While we do have sufficient people available at present, I would appreciate hearing from anyone who is prepared to join the team in order to give the regulars a break from time to time.
The recently introduced card system is now working reasonably well and this helps speed up transactions.

There is a possibility that I will be away for the October meeting and might not be back on deck for the November session, so I have co-opted the assistance of Alan Horne to fill my role while I am away. Alan will be supported by his wife Doreen (as keen a computer buff as you might ever meet) and I am confident that members will give them every cooperation and assistance in their temporary role.

My thanks go to those members who donated books and software at the last meeting. Additions to the libraries are always welcome.

SOFTWARE:

GAMES
EDUCATION
BUSINESS



ACCESSORIES:

DISKS
PAPER
FORMS
PRINTERS
FURNITURE
ETC.

AMIGA.

DISK SPECIALS: "Le Floppy" - 5,25" Disks - Packs of 10 - $15.95
"Centech" - 3,5" Multicoloured D/S Disks in plastic case of 10 - $69.95
[Five different colours per Case of 10 - Lifetime Warrantee]
"Mach 128" Cartridge (64 & 128 Mode) - $59.95
"Power" Cartridge for C-64 - $129.00

Sundown Computer Centre

744 GYMPIE ROAD
CHERMSIDE 4032

TELEPHONE (07) 350 3344

Contact Les Van Tovier or Phil Stafford

REMEMBER — ***WE ARE COMPUTER SPECIALISTS!***

EDITORIAL

Three years ago I took over the post as editor of this newsletter. The August 1983 issue consisted of four A4 double sided pages, which were printed on a duplicating machine. To say that it looked rather 'amateurish' is an understatement. During these three years it has been my aim to produce a better newsletter, using an my examples the ICPUG newsletter of the UK and the TPUG newsletter (really a magazine) from Canada. I feel that, at least as far as appearance is concerned, I have partly succeeded.

But this brings us to the question of content, and here I feel that it's time to re-define our objectives. The first thing to stress is that a newsletter is not a computer magazine. A commercial magazine has to make a profit, and a large slice of its revenue comes from advertising. Consequently they cannot afford to offend their advertisers by negative reviews or attacking their marketing- or servicing policies etc.

A newsletter however is a non-profit making publication, and its main aim is to inform and offer advice to the members of the organization in question. The information part of the newsletter covers such things as meeting times and venues, goods for sale, libraries etc. The 'advice' component of a newsletter is where we really have an important function to fulfil, namely to assist our members in getting the most out of their hobby. In the specific case of a computer club this means programming, hardware, software, and product information in general. As a computer club like ours has a wide range of members ranging from absolute beginners to expert programmers who also use a fairly wide range of equipment, one has to try to cater for all these interests.

In broad terms most newsletter editors would agree that programming as such cannot be taught through a newsletter - for this purpose we have books and magazines and TAFE courses. However, specific aspects of programming will always have a place in this newsletter. Our real strength will have to lie in the field of reviewing hardware and software. As (hopefully) we are not constrained by advertisers we should be in a position to give fair and unbiassed advice to our members, thus saving them time, money, and lots of frustration.

This is where you as members of this group come into the picture. The reviews and/or advice have to come from within our own ranks. In the past I have made it my business to 'fill in the empty spots' if not sufficient material was received for this newsletter - this however is not the editor's job. An editor's job is to 'edit', which means vetting the incoming material, presenting it in a readable form, and looking after the lay-out of the newsletter.

Although I have just been re-elected as editor of this newsletter I feel that, unless there is a dramatic increase in members' contributions, I am no longer prepared to continue as editor. As at this stage no assistant editor is waiting in the wings to take over the reigns, it could mean the demise of "Cursor". The solution is in your own hands!

Ralph De Vries

--ooOoo--

NOTES & GOSSIP

AUGUST MEETING

The Annual General Meeting. After reading (improvising!) of the Annual Report by the President and Secretary the Financial Report was tabled in the absence of the Treasurer. The Group's honorary auditor was present and expressed his satisfaction with the way the Group's account books are kept. Take a bow Lester Bennett! Elsewhere in this issue you will find a copy of the Group's balance sheet.
Lex Hinckley took over the chair to preside over the Election of the 1986 - 1987 Management Committee. As only one nomination for each position was received no ballot was necessary. Obviously the existing management committee must be doing something right, as they were re-elected en bloc. We hope that we can live up to the expectations of our members.
Due to other commitments our treasurer, Lester Bennett, will be unable to attend our main meetings for quite some time. Our member John Van Staveren has been found willing to act as assistant to the Treasurer and, with Lester, will look after the financial welfare of the Group. Welcome on board John!
After the A.G.M. quite a few members had another look at the GEOS operating system. The meeting closed at 10pm.

AMIGA SUB-GROUP

The July meeting of our group was attended by quite a few new faces. Several more existing members have forsaken their C-64's and C-128's and bought Amigas. We feel that (if and when!) a PAL version of the Amiga makes its debut quite a few more converts will join the ranks of Amiga owners. In the meantime existing Amiga owners were busy adding more of our growing Public Domain library disks to their own collection. Lester Bennett was in hot water for not being able to supply stocks of 3,5 inch disks. This was not Lester's fault as our supplier had let us down. Hopefully this situation will be remedied by our next meeting. We will also make available to our country members Amiga Public Domain disks at $10.00 each, which includes postage.

SUB-GROUP NEWS

A new sub-group has been formed in the Kenmore area. The convenor is Keith Hadland, who has advised us that Peter Reeve, one of our most senior members, is prepared to act as coordinator. This is very welcome news indeed. Further details can be found on page 2 of this newsletter.
Barry Wilson, the coordinator of the Cannon Hill sub-group is looking for volunteers to run Cannon Hill during his absence. They may even run a Saturday afternoon group on the fourth Saturday of the month if a coordinator can be found.
Barry is also keen to see a Sub-Group started in the Capalaba district. If any of our members in the Capalaba area are interested they should contact Barry Wilson on 399 6204 for further details.

COMPETITIONS

In this issue you will find the '64000 Byte Question'; a computer quiz prepared by Phil Guerney. Every month the committee will make available a Public Domain Disk to the most successful entrant.

NOTE: Even if you cannot answer all the questions, send them in! Remember,

the winner will be the most successful entrant, which means that even with a 50% score you could be a winner!
We will also make available a similar price to the most successful entrant in our cartoon competition (see last month's issue).

Forward all entries to P.O. Box 384 - Ashgrove - 4060.

THE 1581 DISK DRIVE?

In the July '86 issue of "RUN" magazine a full page is devoted to the new 3,5" Commodore disk drive, the model 1581, suitable for both the C-64 and C-128. The problem is that Commodore did not release it at the Summer Consumer Electronics Show in the USA, and for all we know it may never be released. We seem to have heard it all before.

AUGIE NORMAN

This is how everybody at the Cannon Hill sub-group knew him - not August Norman, his full name. For several years Augie has been co-coordinator with Barrie Wilson of the Cannon Hill group. Augie has now departed from these shores and returned to his native England. On behalf of the management committee we thank him for the work done in establishing and maintaining the Cannon Hill sub-group and wish him all the best in his new/old home country. As Augie is still a member of this group we obviously will remain in touch and hope to hear from him occasionally with reports of the Commodore scene in the UK.

ATTENTION SUPERBASE USERS

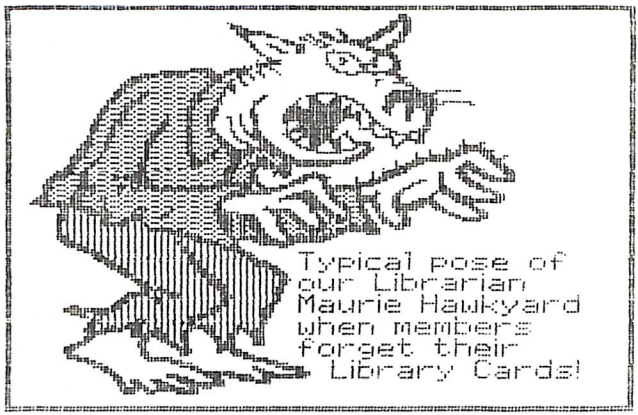
We have just received some disks for our library in the "Stepping Stones to Superbase" series. Currently available are 'Club Membership', 'Cash Book', 'Sales Daybook', and 'Solicitors Time Recording'. These disk can only be used in conjunction with the Superbase 64 program! It is possible to use them with Superbase 128, but a fair bit of conversion has to be done to make them workable. These application disks come without program notes. All information is in the form of 'Help Files' on the disks in question. They can be copied or modified for use in your own applications.

THE WANDERER RETURNS

One of our senior members, Lex Hinckley, has just returned from a seven week stay in the USA. On behalf of the group Lex was sent out on a buying spree to obtain some of the latest software from America. He claimed that it is more difficult to find home computer stores in the US than in Brisbane. And those discount software dealers that we always read about in American magazines aren't so easy to find either. Towards the end of his trip Lex found that the 'Toys-R-Us' chain had one of the widest ranges of software at reasonable prices. Future travellers to the USA might make a mental note of this fact. Amongst the software that Lex brought back were copies of the new fast load C64/C128 version of Multiplan (Epyx) as well as Vizastar 128 and Vizawrite 128. Your newsletter editor has had a quick look at the Vizawrite wordprocessor and has mixed feelings about this program. It has some extremely good features, but the documentation is 'inadequate', to put it politely! Vizastar 128 is a combined spreadsheet, database, with the ability to print out spreadsheet information in the form of graphs. This program seems to be very well documented, and probably will be demonstrated at a future meeting.

UNITED COMPUTERS

Recently your President and Editor were invited to the official opening of United Computers in East Brisbane (the former Commodore premises). Contrary to a report in a previous issue of "Cursor" this new computer shop is a 'Commodore Only' shop, which means that the full range of Commodore computers and accessories is on show here. The renovated premises are very tastefully furnished, and all computers are displayed on a very nice range of computer furniture which can also be purchased. There is a strong emphasis on the Amiga, but other Commodore computers are by no means neglected. Members who are in the neighbourhood of 991 Stanley Street should certainly pay them a visit, even if only to admire the new showroom!



PRINTER RIBBONS

Cor Geels informs us that Jane's Computer Supplies of Milton can supply 801 Ribbon Cartridges for $9.00 and can re-pack 803 Cartridges for $7.00. Epson LX80/GX80 (=MPS 1000) re-packs are $5.00 each. These prices are for standard black ribbons. Coloured ribbons are also available, but are more expensive.
Jane told your editor that she is now unable to re-ink 801 cartridges and Riteman C+ cartridges.

Barry Wilson has advised us that P.R. Business Machines of Morningside can supply new 803 ribbon cartridges for $8.30. They also have the Brother Tractor Feed (in white - Part No. M-1009) for $40.00, which fits the 803 printer.

IMPORTANT WARNING: When buying conventional nylon based ribbons or ribbons that feed past an ink-pad such as used in the 801 and Riteman printers, ask if their stock is fresh!! Your editor recently bought a new ribbon for a Riteman printer (appr. $12.50) which was completely dried out and consequently useless!
The group is currently investigating the possibility of stocking a selection of the more popular types of ribbons. We will keep you informed.

--ooOoo--

PRICE LIST [Members Only]

PUBLIC DOMAIN DISKS - $ 6.00 ea (Postage Paid)
PUBLIC DOMAIN TAPES - $ 2.00 ea (+ $1.00 Postage Per Order)
PUBLIC DOMAIN DISKS FOR AMIGA [3,5" DISK] - $10.00 (Postage Paid)
3,5" DISKS FOR AMIGA - $50.00 per box of 10 (+ $2.00 Postage)
BLANK DISKS - $20.00 per box of 10 (+ $2.00 Postage)
COLOURED DISKS (Red, Blue, Green, a.o.) - $3.00 ea (+$1.00 Postage)
COLOURED DISKS (Double Sided) - $25.00 per box of 10 (+ $2.00 Postage)
DISK BOXES (hold 90 disks) - $20.00 ea (+ $5.00 Postage)
DISK NOTCHER - $8.00 (+ $1.00 Postage)
"PUBLIC DOMAIN BOOK" - $5.00 ea (+$1.00 Postage)
"STARTING WITH DISK DRIVES" - $2.00 (+$1.00 Postage)
"C-128 MEMORY MAP" - $2.00 (+$1.00 Postage)
TURBO-ROM for C-64 or C-128: Members price - $40.00
Customised version (your choice of screen start-up colours + your name on the start-up screen): $45.00
USER PORT PLUG (with Key Way) - $8.00 (+$1.00 postage)
USER PORT PLUG BACK SHELL - $3.00 (+$1.00 postage)
USER PORT TO CENTRONICS CABLE - $35.00 (+$1.00 postage)
COMPUTER DESKS (2 shelves - steel frame) - $60.00

Address all orders to P.O. Box 274 - Springwood - QLD - 4127
Cheques to be made out to: C.C.U.G. (Q) Inc.

--ooo--

UPGRADE CHARACTER EPROM for 801/1525 Printers.
(Gives Descenders on p,q,g,y, and j. Also requires exchange of a ROM chip)
Price (supplied & fitted) $30.00
UPGRADE EPROM to convert 1526 Printer to 802 Printer - $20.00
For further information on the above contact Lester Bennett on 800 1243 before 8 pm on week days.

--ooo--

AVAILABLE FOR HIRE TO MEMBERS ONLY: 1526 COMMODORE PRINTER
For details contact Roger Haigh on 399 8037 (after hours).

--ooo--

EQUIPMENT MODIFICATIONS performed by Anthony Thyssen during main (Milton) workshop- and Graceville Sub-Group meetings.
For further details contact Anthony Thyssen on 870 1862 (a.h.).

SERVICES OFFERED:

RESET SWITCHES: Plug in $6.00
Built in $6.00
[On some 64's the plug in reset switch does not work. In this case you may return switch for a full refund or swap it for a built in switch.]

RESET RESTORER: Plug In $4.00
Built In $6.00
[Tap reset switch while pushing this button. This will reset any protected memory program.]

DEVICE NUMBER CHANGE: Printer/Plotter 4-6 $6.00 - Disk Drive 8-9 $6.00
TURBO ROM INSTALLED: Computer with Socket $5.00 - Socket required $7.00
WRITE PROTECT SWITCHES (Price to be finalised)
SERIAL SWITCHING BOX (Order Only) $14.00
SERIAL PORT DOUBLER (Order Only) $14.00

COMMODORE COMPUTER USER GROUP (QLD) INC.

BALANCE SHEET AS AT 30 JUNE 1986

1985 $		$	1986 $	$
	MEMBERS EQUITY			
3,509.52	Opening Equity			3,659.80
130.36	Net Profit			5,543.85
19.92	Interest			-
-	BBS Membership Fees 84-85 adj			20.00
$ 3,659.80	TOTAL EQUITY			$ 9,223.65
	REPRESENTED BY CURRENT ASSETS			
170.00	Petty Cash	170.00		
903.59	Bank - CTB Sth Brisbane	192.24		
170.00	Sundry Debtors	372.00		
446.25	Stock on Hand	3,953.56		
1,689.84	TOTAL CURRENT ASSETS		4,687.80	
	FIXED ASSETS			
440.83	Book Library	613.34		
221.54	Magazine Library	209.59		
588.62	Software Library	2,413.48		
513.97	Equipment	2,582.38		
915.00	BBS Equipment	1,398.00		
2,679.96	TOTAL FIXED ASSETS		7,216.79	
4,369.80	TOTAL ASSETS			11,904.59
	LIABILITIES			
-	Membership in Advance		1,295.94	
710.00	Library Bonds		1,385.00	
710.00	TOTAL LIABILITIES			2,680.94
$ 3,659.80	TOTAL NET ASSETS			$ 9,223.65

FINANCIAL PLANNING

BY GRAHAM ROBINSON

WHAT IS A SPREADSHEET?

Until the COMPUTER came along, we did most of our financial calculations on a large sheet of paper which we ruled into columns and rows and then entered our text and figures into this 'sheet'.
An electronic spreadsheet does the same thing on the screen of a Microcomputer. However a spreadsheet has two very big advantages over a manual sheet. With a manual system, when we want to alter an entry, we have to erase the previous one, and this usually makes a mess of our work, and secondly we have to mentally recalculate the whole thing again. This is where the 'electronic spreadsheet' really shines, as it can recalculate the whole of a big sheet of calculations in a matter of seconds.

In an 'Electronic Spreadsheet' the points at which each column and row meet are known as 'fields' or 'cells'. As the screen of a Micro cannot display many fields, because of its small size, the screen acts as a window on the Spreadsheet, and this window can be moved over the whole area of the spreadsheet as you work on it.
To save time and space, fields are usually identified by a co-ordinate system, much like a chess board. The top left field is usually identified as A1, i.e. column A and row 1. (MULTIPLAN uses a co-ordinate system of R1 for Row 1 and C1 for column 1 etc.).
Into each field you can type a "label" (such as the word "costs"), "values" (such as 23.5 etc.), or a "formula" (which is a calculation we wish to do, using data from other fields. These calculations can include *,/,+,and -, and just about any combination of these you require).
The formulae we would mostly use for Business or home calculations and projections are addition, subtraction, multiplication and/or division. However, much more complicated calculations can be undertaken with a spreadsheet, such as "if then else", which I find very useful in calculating my projected taxation bill.
The Electronic Calculator was a boon when it first became available because it could perform complex functions very quickly. However, business calculations, rarely require very complex calculations, but they may involve very many variables which would affect the end result.
The Electronic Spreadsheet uses the power of a Micro Computer to provide a fast and easy calculator for business use.

"VISI CALC" was the first of the Spreadsheet programs, and it was largely responsible for the Micro Computer Boom.
Since it came on the market, the whole idea has been copied and refined, and there is now a very wide choice of "Calc" packages from which to choose.
The big advantage of a spreadsheet program over a manual system is that when you change any value on the sheet, all other values which depend on it, either directly or indirectly, change automatically if you correctly set it up in the first place.

WHERE CAN YOU USE A SPREADSHEET ?

The answer is, of course,for any sort of calculations which use rows and columns.
Monthly budgets, yearly budgets, expenditure analysis, and costing, are only some of the things we can use it for in our day to day work.

The two main benefits, apart from the time saved, are guaranteed accuracy in the calculations performed (provided the formulae are correct), and the neat presentation obtainable from the printer.
Once you use a spreadsheet for your budgeting, you will wonder how you ever got along all these years without one.
Spreadsheets are fairly easy to learn, and the learning process should not take any longer than a couple of hours. The time and effort you will eventually save yourself will be well worth it, and you will never want to go back to a manual system again.

--ooOoo--

FROM THE DISK LIBRARIAN

Our recent Public Domain Disk 022 'UH' incorporated a program called SPEEDCALC. It was brought to our attention that this program was corrupted. We have now rectified this fault, however several disks were sold to members before the error was pointed out to us.
If you have purchased such a faulty disk, we will exchange it free of charge. Please bring your faulty disk to either the main meeting, the monthly workshop or post it to the disk librarian P.O. Box 274 , Springwood. We will mail a new copy to you within a few days. We have also included a SPEEDSCRIPT file of instructions on SPEEDCALC on the new disk.

In the mean time, here is the latest club disk:

Disk 031 C.C.U.G.(Q)INC. ,UQ

MENU	Run this first
DIRECTORY PLUS	Prints not only the directory, but also track, sector and starting address.
DATABASE	A simple database. Ideal for card file
FINANCE PACKAGE	Works out loans,amortization,devaluation etc.
ATLANTIS ADVENT'R	Adventure Game
MOVING SIGN	Displaye lage messages to screen
HOME BUDGET	Manage your finances (Tape based)
TREKSTAR	Space Game
CONVOY	Game
DISK INSPECTOR	Track and Sector Utility
SOLARPIX	Follow the stars. From Compute's Gazette June86
SWITCHEROO	Another Game from Compute's Gazette June 1986
MONOPOLY 4	Latest version of Monopoly for 4 Players.
ZONEBOOT	Game with very smart graphics.(Loader)
TITLE2	Loaded by above
THE ZONE	" " "
ZONEINSTRUC	" " "
STORM WARNING	Game [Game for two players
S/W 01	Loaded by above [
S/W 02	" " " [
S/W 03	" " " [Try to control the
S/W 04	" " " [weather. At the same
S/W 05	" " " [time learn to guess
S/W 06	" " " [angles.
S/W 07	" " " [
S/W 08	" " " [
	[

--ooOoo--

Language arts

by Jim Butterfield

A language is more than just a method of communicating information: it also involves transforming the information in some way. ASCII is a code, not a language: it doesn't modify the data in any way. When we express a thought in English, French, or Chinese, we adapt that thought to the 'mindset' of the language involved. The idea often comes through slightly modified by the language: translators often have difficulty carrying a thought from one language to another.

The same is true of computer languages. Your choice of Fortran, BASIC, COBOL, LOGO, machine languge, Pascal, C, Lisp, Modula 2, COMAL, APL, or Snobol (to name just a few) will influence the way you approach a problem. And most languages have dialects: for example, there are many BASICs to choose from.

From a human standpoint, there are two general quality levels of person/processor communications: poor and lousy. A simple question such as 'Got the time?' might produce:

?syntax error

Poor communication both ways. Or you might get:

command 'got' not in vocabulary

Better, but not much.

file 'the time' not found

At least it's trying.

the time at noon today was 12:00

This isn't too useful.

explain 'got'

This is going to take a while.

We have to reshape our thinking to communicate with a computer. The manner in which you type commands is a language, and so are the responses you receive (oddly enough, these input and output languages are different).

When a program runs, the information we type in is often a kind of language. This is especially true if we have a large number of options. Thus, your word processor's commands consitute a language, and some WP's have better languages than others. Spreadsheet programs, too, have their own languages. You need to become proficient in the use of these languages in order to use the programs efficiently.

Interpreters and compilers

The 'formal' computer languages — Fortran, BASIC, and so on — are generally grouped into two types: compiled and interpreted. For those unfamiliar with the terms: a compiled language (such as Fortran) must be completely written and then translated ('compiled') before it can start to run; whereas an interpreted language is translated as it runs. Compiled programs run faster; interpreted programs start quicker. If a compiled program needs a change, it must be recompiled — a lengthy process; an interpreted program can be fixed and restarted in a few moments.

Early computers were expensive and could do only one task at a time. All programs were compiled for fast run time. Fortran and COBOL dominated the high-level language world.

In the mid-1960s, computers were still expensive, but multitasking became practical. One slow job wouldn't hang up the whole computer. An array of 'computer ports' allowed many users to be connected. A single 'interpreter' program could run many users' jobs at virtually the same time. The 'time-sharing' computer concept started in the field of education and spread to computer service companies. The first two interpreted languages were JOSS (later adapted to FOCAL, and now almost forgotten) and BASIC.

The BASIC languages we see today are greatly changed from the original BASIC, but the style is still recognizable. Good old, sloppy old BASIC... it's so universal in small computers that it seems unlikely that it will ever be superseded. There's both joy and pain in 'spaghetti code', and BASIC lets you write any way you want.

With the advent of personal computers in the late 1970s, interpreted languages — BASIC especially — became the standard. Interpreters took up less space in these small computers (there was even a 2K tiny BASIC!), and they were more friendly to beginners who were just discovering computers. The interpreter could be placed in ROM — read only memory, another technological innovation — so that the language would be in place the moment the computer was turned on.

Now we're going through a period of reflection and consolidation. There are now compilers for languages such as BASIC, which were designed to be interpreted. These can be only partially successful. For example, the design of BASIC leads to clumsy handling of some variables — especially small integers — and a compiler must preserve this awkward logic or risk being 'incompatible'.

On another front, 'dual' languages have been designed, such as Pascal, which are capable of being either interpreted or compiled. Even so, in practice it seems that commercial packages seldom do both jobs well... they always categorize themselves into one camp or the other.

I can recall asking one of the COMAL principals why they had no compiler for their language. He emphasized that COMAL was designed as an interpreted language, and that specific language decisions had been taken with this in mind. He was correct... but I still pine for the best of both worlds.

Which, when and why

I think that most people will choose a language based upon the kind of programs they want to write. For payrolls and billing statements, a language like COBOL is wonderful. For advanced engineering, Fortran has major advantages; artifical intelligence — Lisp or Prolog; text editing — Snobol; mathematically-oriented material — APL. The list goes on, and different programmers will have their own favourites. Some languages seem to be in the eye of a storm: for example, C is an object of both love and hatred.

But most of us can't go out and learn a new language every time we have a new program to write. We must develop skills with a few selected ones. And in most cases, the most prevalent language will turn out to be... BASIC.

This leads to the next question: which BASIC? But that's a question for another time... □

QUO VADIS COMMODORE

by Ralph De Vries

After last month's pessimistic editorial about Commodore's future it would be nice to be a bit more positive on this occasion. However, after having read the editorial in the August issue of "Compute!", I am less positive than ever.
This editorial covered the Summer Consumer Electronics Show, where traditionally companies like Commodore and Atari show off their wares. Atari (Jack Tramiel and family) had a huge stand with as centre piece the ST range of computers. Commodore however had a very small stand on which they only displayed the 64 and 128. There wasn't an Amiga in sight, as Commodore apparently has decided that the Amiga is a 'Business Machine', and does not belong at a Consumer Electronics Show! Robert Lock claims in this editorial that Commodore's Amiga sales have stagnated and that "we find the ST with an installed base of roughly ten times that of the Amiga". He goes on to suggest that Commodore should target the Amiga towards the consumer market with some price cutting, rather than concentrate on the so-called business market. How right he is! When all is said and done the Amiga is a far better computer than the Atari ST, but convincing the market place is another story. My own feelings are that if the Amiga fails in the USA, Commodore will probably fail as well.

On a more positive note the 8-bit machines are still going well. Officially released in the USA were the 64C, in a 128 style case, but no changes internally, as well as the 1541C drive, again re-packaged in a cream case. And, as we all know by now, the 64C is sold in the States with the GEOS operating system.
The surprise however is the C-128. Commodore's Nigel Shepherd claims that over 600 000 units have been sold worldwide. These are very impressive figures, particularly when you realise that in the USA a 128 package of computer, drive and colour monitor costs just as much as a similar Atari 520 ST package! The memory expansion units for the 128 are available now in the States, costing about $200.00 for the 128K unit and $300.00 for the 512K unit. However, to utilise this extra memory, you will initially have to write your own routines (mostly machine language at that!), because to our knowledge no commercial program currently supports these memory expansion units. **This in turn means that almost all the 128 applications software will have to be re-written to be able to make use of these memory expansion units!**

This brings me to the wretched question of C-128 software. 128 Owners are getting sick and tired of seeing software advertised as C-64 / C-128 software. They KNOW that their machines can run C-64 software! **What they want is software specifically written for the C-128, making use of the extra memory, the 'Fast' mode, the 1571 double-sided drives, the better Basic etc.** Our own president, Greg Perry, showed us with his membership program how to do it. Commercial software developers are very short-sighted if they don't start to support all the available features that the 128 has to offer.

So, where to now, Commodore? As a long-time Commodore user I sincerely hope that the danger signs are heeded by the top management. I, for one, want Commodore to succeed in the market place, as no doubt do the other five million odd users!

--ooOoo--

THE THREAT.

Don't just sit there looking bored.
The mechanical monster,
Seemed to say to it's lord.
Don't think of the money.
Don't think of the cost.
Go see the Pirates,
Tell them others to get lost.

I'm needing a feed,
and its got to be regular.
Not this day to day stuff,
That's so boring and secular.

If the lolly's run out,
You'll have to work over.
I'm not sitting here,
Like a darned piece of sculpture.

And just to make sure, that
You've got the right message.
I'm screening a warning.
You've better take notice.

If the input is wanting,
And not up to par.
The output I'll give you,
Will flame you afar.

I'll scramble your addles,
And girate your genes,
Till your family comes running
At the sound of your screams.

Maureen Barrow.

--ooOoo--

CARTOONIST OF THE MONTH: LINDSAY WHIPP

The train that Norm never had as a kid!

The 64 000 Byte Question!

by Phil Guerney

It's time our club members have a chance to show how much they know about their 64 and get fame and fortune in return (well perhaps their name in a following CURSOR and a free PD disk if the committee feel especially generous!). Just be the one to submit the best set of answers to the club before 2nd of Sept. I'll try to have ten questions ready for each month. Don't worry if you don't know the answers, it's the experimentation to work them out which is fun and may turn you into another Raeto Colin West or whoever else I crib the questions from!

1. Why does the following line in a program result in over 1K of memory being used up ?

```
FOR I=1 TO 45: A$=A$+" ": NEXT
```

2. What does WAIT 56320,16,16 do in a program?

3. What is/was/never was the Commodore MAX?

4. What does the following structure in BASIC do that could be useful?

```
 90 A=1: B=2
100 FOR J=0 TO -1 STEP 0
..(
..( block of code)
..(
180 J=(A=B)
190 NEXT
```

5. Normally you cannot INPUT a string containing commas or colons because the comma or colon is taken as the end of the input. But you can allow commas and colons to be INPUT by preceding the string with a particular character. What is it? For a bonus, two POKEs before the INPUT in the program will do this automatically, what are they?

6. There are four graphic characters that do not appear on the front of any of the keys. What are they & how do you print them on the screen?

7. Print a medium grey stripe on the normal text screen (eg. [rvs on][<commodore 5>][10 spaces][return]) then change the screen to multicolour with POKE 53270,216. What colour does the stripe become and why? (to turn off multicolour POKE 53270,200)

8. A game cartridge maker could include the machine language instruction INC $8300 as part of copy protection code against disk versions of the program working. What would come next in the code and why would this work?

9. What do you get if you read the disk error channel immediately after turning on the system?

10. When is the other time a message appears in the disk error channel but the red light does not flash?

Well some of those are hard and some aren't easy. But it wouldn't be any fun otherwise! Good luck! [See 'Notes & Gossip' for more details]

Amiga BASIC

by Dick Barnes

*The **SuperPET Gazette**, which Dick Barnes edits, is the official newsletter of the International SuperPET Users Group, and is an excellent source of information on the SuperPET and much else besides.*

The buggy ABasiC for the Amiga has been replaced by a new Amiga BASIC from Microsoft. Commodore began distributing this new dialect with V1.1 of Amiga's operating software. I was considerably surprised to find that Microsoft has not only entered the 20th century with many structured features, but has also maintained compatibility with older versions of the language — no mean feat.

A flavour of the language

At last you may indent code. **While...wend** is built into the language. You may construct any structure wanted. Labels are allowed — and must be followed with a colon. Line numbers are optional (so is free speech in the USSR, and on the same terms, as we later see). The language converts all keywords to capitals, whether you like it or not. Variable names may be up to 40 characters long. **If...else if...else end if** is now allowed, in the form shown, with no limit on the number of **else if**s. The **then** statement is required; **else** is optional.

There is a price for upward compatibility. Variable names aren't sensitive to case: **alpha**, **Alpha** and ALPHA are the same. You may use periods in variable names: **file name** fails; **file.name** is okay. There are no modern **mat** statements; all work on arrays must still be done in loops.

For compatibility, Microsoft hauled its archaic and clumsy method of handling REL files into Amiga. You have a bushel of separate intrinsic commands specific to random-access files, of which only two (record length and record number) are necessary for such work in a well-designed language. **Line input**, however, did see the light of day, although the simpler **linput** was lost along the way.

Amiga BASIC has two types of subroutines, one of which is a 'subprogram' with local variables and parameter-passing. Ordinary subroutines may be entered with a **gosub** to either a label or a line number. Subprograms may be called, as shown below. You have two options: in the first, you pass arguments by reference. This changes the values of **a**, **b**, and **c** in the main program by what is done to **x**, **y** and **z**.

```
CALL sendscreen(A,B,C) 'cre
     ate A,B,C values
SUB sendscreen(x,y,z) STATI
    C
if <condition> THEN EXIT 'm
   anipulate x,y,z
...statements
END SUB
```

In the second option, you may enclose the arguments in parentheses, as in:

```
CALL sendscreen((A),(B),(C)
     )
```

The values of **a**, **b** and **c** in the main program remain unaffected. In short, you have full control over whether a subprogram's variables are local or global. In addition, you may define any variable in a subprogram as a **shared** variable, which may be used in any other subroutine or in a main program, and which passes its value there. So far, so good.

Unfortunately, recursive subprograms aren't allowed. Do you see the **static** that follows the subprogram name above? It's required. Nobody in his right mind demands that subprograms be designated **static** if that's the only flavour you can define. And it is! **Static** (like other features of this BASIC) is derived from C. Static variables retain their value in a subprogram between calls. The other C option of automatic variables, which do not retain their value between calls, is not found in Amiga BASIC, but the presence of **static** indicates that Microsoft plans to use it in later versions. Either that or **auto** variables were taken out at the last minute. Anyway, because all subroutines are **static** and all variables retain their values, recursion becomes deadly; the manual warns users not to try. Too bad.

The manual comes with an errata sheet that asks you to remove several references to **reset** (which closes all open files). It seems they planned to have it and jerked it out at the last moment — along with the **auto** variables. We'll later see why.

Line numbers vs. labels

You are indeed free to use line numbers, which I prefer for reference when I list or edit. But there is utterly no way in Amiga BASIC to generate those line numbers automatically, or to renumber them thereafter. The language doesn't even check to see if line numbers conflict or are in proper order. Microsoft obviously doesn't want you to use line numbers.

How do you list a part of a program? Well, er... uh... you'd better stuff in named labels every 20 lines or so, or add line numbers on the same basis — or you must list your program from beginning to end to find the part you want. When we tried to list the subroutine below (without the label **test:**), Amiga BASIC refused. You must add either a line number or a label (as with **test:**) despite the fact that the subroutine is already named. And there is no search or search/replace in Amiga BASIC's editor, either.

```
SUB test STATIC
...
END SUB
```

Suppose you want to add a library routine from disk to a program on screen. You **merge** it at the end, then **copy** and **paste** into the program. Line numbers have no effect on a **merge**.

Directories

You're supposed to be able to get directories by saying, for example: **files "df1:basic.programs"**. Do you get the Amiga's typical two-column listings? Nah! You get a list in one column, and it scrolls right off the screen unless you are fast enough to hit **right-amiga s** to stop that listing. Golly, when will those folks in Belleview learn to pause listings at the end of each screen page? Sorry to say that half the files on our Workbench disk are never listed at all. It's a bug. Well, click over to DOS for directories...

Immediate mode

You may not cursor up and amend. We're used to working out difficult algorithms in immediate mode, by defining our constants and variables and testing variations in often-complex code. You won't do that in this BASIC: you must retype all the code each time you test, which is so time-consuming that you are far better off to make a program of the test. And — a **run** always clears the screen (you

can't control that), which wipes out all previous results. Most of the utility of immediate mode is thus destroyed. We retreat to making pencil copies of data we've been handling on screen for years. Back to 1920.

You have two separate windows: one for program output, one for listings; you must interminably click your way back and forth between them. It seems this was done so that you can see your code execute in one window and trace the executing line in the other. We tried it for a couple of hours and conclude that the folks at Belleview are mad. The output window is always obscured by the list window; a little red border skips from one executing statement to another so swiftly that you can't follow it. If you try tracing one line at a time, you find that every comment is considered an executing line. We'd much prefer a single window with the executing statement printed at top or bottom of the screen. Programmers often get carried away by enthusiasm for fancy and faddish new approaches. So it is with two windows in this BASIC. They're a useless pain in the behind.

Error messages

Despite a large error window ("Syntax Error!"), and a red border around the line of error, Microsoft's error messages are no more useful now than they were five years ago. You aren't told where on the line the error is, nor why it's an error. Example: what's wrong with the statements below? You get an error message 'ELSE/ELSEIF/ENDIF WITHOUT an IF". Is that the real problem? Those accustomed to systems with intelligent error handling and marking will weep. The problem? Shucks, we left out a **then**.

```
IF x OR y
PRINT "Whoa!"
END IF
```

Editing

The editor in Amiga BASIC is both primitive and clumsy. It is also very slow. The screen scrolls very slowly when you page up or down by screen page. It takes one full second to move the cursor from the top line to the bottom of a 19-line page unless you shift the cursor with the mouse. The interpreter of Amiga BASIC may be in assembler; the editor is done in molasses.

Editing within Amiga BASIC otherwise is fairly handy. You delete and copy lines or sections of code by highlighting the material with the mouse. You then copy the material to a buffer (called the 'clipboard'). You may copy erased code to a new position, shown by the cursor, with **Amiga P** (for **paste**). If you choose to copy, the original code remains, but a copy is put into the clipboard; you may insert that copy anywhere else in text. This works well on small segments. We tried to delete the second half of a larger program by highlighting with the mouse. By stopwatch it took over four minutes to highlight most of the material; then we crashed (we suspect that the clipboard buffer overflowed). So you must delete in small sections. In sum, you'll probably use this editor to debug code; it is far too slow and clumsy for long editing sessions.

Fortunately, Amiga's multitasking allowed me to use **Ed**, Amiga's screen editor, to write programs. And then, with **Ed** running concurrently, I crashed Amiga BASIC six times in two hours and forty minutes, with nary a program in memory longer than 80 lines. I kept getting 'Heap Full' errors.

von Lundsdorff

There's a text-to-speech demo on the Amiga BASIC disk. I cranked it up and found it hilarious. The default voice settings sound precisely like a Nazi villain, so I tried: "Let me introdusse myzelsf; I am Count Erich von Lundssdorff, und diss is Frankenstein." The extra sibilants were necessary for proper effect. The range of voices possible is limited: I tried for hours to get rid of the Teutonic flavour, without success. But those who want trolls, ogres, wizards and witches will have great fun. The built-in **translate$** function converts any Engish sentence into proper phonemes for the voice. If you want to bypass **translate$** and control the voice directly, you may do so — but with phonemes, glottal stops, and such.

Windows, mice and menus

Amiga BASIC gives the programmer control over all the features of the Amiga, from pull-down menus to windows and the use of the mouse and joysticks — if you care to use them. A very large part of the lanuage is devoted to these things; they are powerful and easy to use.

ML interfacing

Amiga BASIC provides both **varptr** (the address of numeric variables) and **sadd** (the address of a string), so that you may access any variable by its address. You may therefore pass these named variables to ML routines. You may also call ML library routines that you write yourself or those already present in Amiga.

A few nice touches

The index function to find a substring within a larger string (called **instr**) allows a secondary index, as in: **INSTR(7,X$,"Brown")**. In this case, you look for "Brown" within a larger X$, but you start searching at the 7th character in X$. The secondary index is optional. Very handy.

More importantly, you may identify variables either by defining their data type (as in Pascal or C), or by using suffixes, as in other BASICs. The identifiers shown below (% for a short integer, & for a long integer, and so on) take precedence over any stated definitions.

Definition	Meaning	Suffix form
DEFINT	var Short integer (16 bits)	var%
DEFLNG	var Long integer (32 bits)	var&
DEFSNG	var Real (7 digits, 4 bytes)	var!
DEFDBL	var Real (16 digits, 8 bytes)	var#
DEFSTR	var String	var$

In subprograms, the definitions may be purely local, or they may be **shared** if so declared within the subprogram. If they are shared, they become global variables in the main program, and may be used in other subprograms if declared there as **shared**.

Tests on variations

I ran a number of tests, and was surprised to find no difference in execution time with or without line numbers. Even more surprising: a program using defined variables (as above) runs faster than one where variables are suffixed with their type (**defint y** is faster than y%). Last, and most astonishing, multiple statements per line are slower to execute than single statements per line! In this language, there is utterly no excuse for cramming code together in the fewest possible lines. All these things are positive advantages: they reward simple, readable code.

Summary

This BASIC is much improved over ABasiC but just doesn't go far enough. The code is far easier to read and maintain than older, unindented, and unlabelled versions. It was hastily and drastically shortened at the last minute to let it fit in available memory. As time goes on and code is boiled down, or as more memory becomes available in the Amiga, I wouldn't be at all surprised to see **auto** variables, recursion, **reset**, additional optional data types, and some fully structured options. It is far superior to any Microsoft Basic I've seen before, and a large step in the right direction. □

Amiga BASIC sound and screen

by Chris Johnson

Amiga BASIC makes the advanced sound and graphics features of the Amiga easily accessible to the BASIC programmer. The use of 'bobs' (blitter objects) and sprites is fully supported in Amiga BASIC; screen images can be captured in an array and placed elsewhere on the screen or saved to disk; and the **sound** command makes music easy to implement. Unfortunately, some aspects of screen manipulation are not supported as fully as in ABasiC.

Using colour

Colours for the various graphic commands are defined using the **palette** command. Up to 32 colours can be defined by the amount of red, green and blue in the final colour. For example, **palette 1, .8, .6, .53** assigns a brown to colour-id 1. The number of colours accessible at any time is determined by the depth (number of bitplanes) of the screen (set by the **screen** command). If the **palette** command is omitted, Amiga BASIC uses the colours defined with the **Preferences** tool. **color <pen>,<background>** sets the background and foreground colours using the colour-id as defined with **palette.**

Graphics commands are executed using the pen (or other, specified) colour. These commands include:

- **Line**, which can also be used to draw a rectangle by specifying two opposite corners and adding ,b after the co-ordinates or ,bf to fill the rectangle with the pen colour;
- **Circle** can be used to draw arcs and any variety of ellipse;
- **Areafill** fills an area defined by **area** statements;
- **Paint** fills an enclosed area with a specified colour: and,
- **Pset** and **Preset** set a point on the screen. **Preset** differs from **Pset** in that, if you do not specify a colour, it uses the background colour.

Commands that draw lines or fill areas can be modified with **pattern**, which sets a 16-bit mask. **Pattern &hffff** would turn on all the bits when a line is drawn; **Pattern &hf0f0: line (0,0)-(100,100)** draws a dotted line. (The **&h** in these commands specifies that the following number is to be interpreted as hexadecimal.)

Bobs and sprites

Bobs and sprites are controlled with a series of **object** commands, each of which has a different suffix:

- **.Shape** assigns an object-id number to an object definition contained in a string expression;
- **.On** and **.off** make an object visible or invisible;
- **.X** and **.y** place the object at a specified point in the x or y axes or, used as a function, return its position;
- **.Start** and **.stop** set an object in motion or freeze it;
- **.Vx** and **.vy** set the velocity of the object in X and Y coordinates;
- **.Ax** and **.ay** set the acceleration in pixels per second per second;
- **.Priority** sets a bob's priority. A bob will appear in front of an object with a lower priority and behind one with a higher priority;
- **.Clip** defines an area of the screen beyond which an object will not be drawn;
- **.Hit** allows selection of which other objects will cause a collision;
- **.Close** releases memory assigned to an object.

Collisions between objects are detected by event trapping, which is enabled, disabled and suspended by **collision on**, **collision off** and **collision stop** respectively. When enabled, up to 16 collisions can be queued and read by **collision(<object-id>)**, which returns either the number of another object or a negative value indicating top, left, bottom or right border.

Object Editor

The creation of a bob or sprite can be done with a utility program called **ObjEdit**, which is one of the demo programs on the Amiga Extras disk. Written in Amiga BASIC, this program allows you to draw your bob or sprite on the screen with the mouse. It includes a 'zoom' command that enlarges the object for detailed work. The objects can be saved to disk and read into memory when necessary. Concise examples are given both in the manual (with a few errors) and on the disk.

Screen Get and Put

Areas of the screen can be stored in an array with **get**, then **put** to the screen. An array of sufficient size must first be dimensioned, then **get** is used with the pixel coordinates that define a rectangle containing the image. These can be **put** in one of several ways: **pset** or **preset**, or the images can be **and**ed, **or**ed or **xor**ed with the image already on the screen.

Printing to the screen

Printing to the screen can be done with **print**, **print using** and **write**. **Print** behaves in the normal fashion, as does **print using** for those who have used it before.

Write, on the other hand, prints an expression list, the items of which must be separated with commas. Numbers are printed without leading spaces, strings are printed in quotes, and the delimiting commas are printed.

Width defines the maximum number of characters that will print on a line: Amiga BASIC will not print a carriage return when it reaches the edge of the screen if **width** is set to the default of 255.

There is no **print at** command in Amiga BASIC (there is in ABasiC) and the **locate** command positions the cursor only by character, not by pixel. Positioning text at a precise location on the screen in combination with graphics is thus made somewhat difficult. The **ptab** command moves the cursor horizontally by pixel. There is no direct method for positioning text vertically on anything other than the normal printing lines.

There are two awkward ways around this limitation. The first is to print the text to the nearest line, then **scroll** a portion of the screen until it is where you want it. The second is to print the text somewhere where it will not interfere and store it in an array with a **get**. This can then be **put** anywhere. Neither of these is entirely satisfactory, and one misses the ease with which this operation could be performed in ABasiC.

For some uses, however, **scroll** is a handy command. Two corners of a rectangle are defined, followed by the number of pixels to scroll the screen in both the x and y axes. This allows diagonal as well as left, right, up and down scrolling.

There are also no commands for cursor movement, inserting characters, clear to

end of display or clear to end of line. The ANSI standard commands which the system supports and which work in ABasiC (see Amiga Screen Magic, *TPUG Magazine* #22) do nothing in Amiga BASIC. Homing the cursor must be done either with **cls** or **locate 1,1**. Relative cursor movement (to the previous line, for example) has to be done by reading the current line with **csrlin**:

```
y = csrlin
x = pos(0)
locate y-1, x
```

Sound and music

The easiest method of getting a sound with Amiga BASIC is by using the standard **beep** (or **print chr$(7)**).

But you probably want to do more than this, and in Amiga BASIC it is not all that hard. After the convoluted method of producing sound in ABasiC, this is child's play. Unfortunately, there is no command for setting the ADSR envelope, as there is in ABasiC. ABasiC also contains functions that can tell you whether any voice is still playing. New sounds can then be sent to an inactive channel. Amiga BASIC has none of this.

There are really only two commands you need to know to create music: **sound** and **wave**. Wave forms are set up in integer arrays of at least 256 elements; each element in the array must be in the range of -128 to 127. The array is then assigned to **wave** 0 to 3, corresponding to the four voices.

Notes are played with: **sound <frequency>, <duration>, <volume>, <voice>**; the last two parameters are optional. **Sound wait** suspends execution of **sound** statements until a **sound resume** is encountered, allowing synchronization of all four voices.

The music demonstration provided with Amiga BASIC, J.S. Bach's *Jesu Joy of Man's Desiring*, is a magnificent example of programming that will repay a close examination. It is not immediately apparent how the program works, but with a little effort it can be used as a driver for your own music simply by replacing the **data** statements.

Summary

Though Amiga BASIC is, on the whole, a great improvement over ABasiC, many features have been left out. The manual makes references to continued developments; let us hope that these are forthcoming.

To our local Amiga users:

The two articles on Amiga Basic were taken over from the April 1986 issue of the Canadian TPUG magazine. This type of critical article on one of several aspects of the Amiga is very much overdue and we are looking forward to getting some comments from our own Amiga members. This would be of particular value to our country member Amiga owners, as at this stage they are absolutely starved for information.

Editor

==========

--ooOoo--

****

*

* IMPORTANT NOTICE TO ALL PRESENT & FUTURE COMMERCIAL ADVERTISERS

*

* Recently we have sent to all commercial advertisers an updated schedule
* of advertising charges for this newsletter. We have kept our charges
* the same now for some years and feel that, with our current membership
* strength, advertising in "CURSOR" is better value than ever.
* In the past I have actively canvassed for adverts and in some cases
* prepared layouts etc, and even have had to collect the adverts from
* our clients.
* As ours is a voluntary organization without financial remuneration I
* regret to advise all advertisers that I am no longer able to perform
* the above mentioned services.
* We therefore request the support of all our present and future
* advertisers by providing us with their layouts by the First Tuesday
* of the Month to ensure publication in that month's issue of "CURSOR".
* Any material received after this date will, regrettably, have to be
* held over till the following month.
* We thank you in anticipation for your co-operation and patronage.

*

* Ralph De Vries - Newsletter Editor

*

****

EASY SCRIPT AND THE 1526/802 PRINTER

by Rev. W. Peter Gadsby

Recent issues of CURSOR have contained articles about "Controlling the Printer with Easy Script" (Nov. 1985) and "The 1526/MPS 802 Printer" (July 1986).

As an owner of an MPS 802 printer, I had wondered for some time whether all the features of this machine could be accessed from within the EASY SCRIPT word processor. Was it possible to define programmable characters? Could I print more than one on each line?

Well, the answer to both questions is 'Yes' although the process is tedious to say the least!

Consider the following short piece of text:

This is a test to determine whether it may be
possible to insert Hebrew יהוה אלהינו in a line of
English text using EASY SCRIPT and the MPS 802 printer.

(The 'programmable characters' are the Hebrew for "The LORD is our God")

If you look at the next page, you will see how much has to be done from inside EASY SCRIPT to produce this result! What did Jesus say about swallowing gnats and straining at camels!!?

Yet, the example illustrates one or two interesting principles:

1. The article in CURSOR July 86 said that you can print multiple programmable characters on one line by using a CHR$(141) (= carriage return with no line feed). However, you cannot do this from inside EASY SCRIPT because the only way to send a CHR$(141) to the printer is to define a special character, say Ø=141 as I have done in the listing (note 1). But if you then try to send a second group of data to secondary address 5 (to redefine the programmable character), EASY SCRIPT gives you an OUTPUT ERROR N. The program does not know that a reversed "1" is a kind of carriage return, and so reminds you that your format command (.) must be at the beginning of a line. But if you end a line with a <return>, the printer does a line feed and so your prog. characters won't be on the same line!

2. Not to be beaten, I thought, "Well, you can send data to secondary printer addresses from inside EASY SCRIPT, so why not set the line spacing to zero? Then I can use as many carriage returns as I like and the paper won't move a bit!" It worked! You just send a Ø to sec. address 6 (see listing note 2), and hey presto!

3. Having frozen the paper in place, it was now just a matter of defining the special characters one after another, and printing them in the right place. The Hebrew letters were drawn onto an 8 by 8 grid, the data for each worked out, and the extra zero added to avoid terminator errors (see July CURSOR). The EASY SCRIPT special character 1 was defined equal to 254 (note 3), so that every time the word processor sent a CHR$(254) the 802 would print the current programmable character. The left margin was reset to the appropriate place (see listing note 4), and the characters had to be moved along the line so they would not overprint. After the prog. characters, a few word of English were also put on the same line. Once the

```
■lm5:rm65:0=141:jul <ret>   ( NOTE 1 )
This is a test to determine whether it may be■■sa6,0 <ret>
possible to insert Hebrew <ret>
<ret>
■lm31 <ret> ( NOTE 4 )
■nb"0:YHWH ELoHeNU" <ret>
<ret>
■sa6,0 <ret>  ( NOTE 2 )
<ret>
■sa9 <ret>
<ret>
■l=254 <ret>  ( NOTE 3 )
■nb yhwh elhnw (The LORD our God) <ret>
■nb waw <ret>
■sa5,0,0,224,96,127,0,0,0,0 <ret>
■ <ret>
■nb nun <ret>
■sa5,0,0,1,227,99,126,0,0,0 <ret>
 ■ <ret>
■nb he <ret>
■sa5,0,223,96,96,96,127,64,0,0 <ret>
  ■ <ret>
■nb lamedh <ret>
■sa5,0,0,224,97,98,100,120,0,0 <ret>
   ■ <ret>
■nb aleph <ret>
■sa5,0,237,123,56,92,254,103,0,0 <ret>
    ■ <ret>
■nb he <ret>
■sa5,0,223,96,96,96,127,64,0,0 <ret>
      ■ <ret>
■nb waw <ret>
■sa5,0,0,224,96,127,0,0,0,0 <ret>
       ■ <ret>
■nb he <ret>
■sa5,0,223,96,96,96,127,64,0,0 <ret>
        ■ <ret>
■nb yodh <ret>
■sa5,0,0,224,96,112,0,0,0,0 <ret>
         ■ <ret>
■lm42 <ret>
in a line of <ret>
<ret>
■sa6,36:lm5:rm65 <ret>  (NOTE 5)
<ret>
English text using EASY SCRIPT and the MPS 802 printer.
```

process was over, a "36" was sent to secondary address 6 to reset the line spacing to 6 lines per inch (note 5), the left margin reset, and printing continued normally.

As I said, "Tedious." However, the principles are interesting and there are more practical applications than printing Hebrew! For example, you may now do underlining on the 802 by defining an underline character. However, unless you do a small linefeed, the underline overprints the descenders on letters g,j,p,q, and y.

I had other problems just getting this article printed up, but won't bore you with details; the main point is that from inside EASY SCRIPT, it is possible to print multiple programmable characters on one line using the 1526/MPS 802 printer. Shalom!

--ooOoo--

July Quiz Solution

by Greg Perry

Since we received no replies to Hank Deucker's short quiz in last month's newsletter, I guess I have to make some attempt myself!

1) GOTO's and GOSUB's

Both commands work the same way. In brief, if the required line is less than the current line then the interpreter goes back to the first line of the program and begins its search. However, if the required line is greater than the current line then the search begins from the next line.
Readers should think about this when writing programs. With some thought it is possible to make the program run more efficiently by carefully structuring the GOTO's and particularly the GOSUB's.

2) DIM Statements

As most programmers know, the task of the DIM statement is to inform the program that memory space will be required for a set of variables. For example, DIM A(200) allocates approximately 1004 bytes of memory in the array area of RAM for 200 floating point numbers. (5 Bytes per number.)
One of the problems in a large program occurs when a new simple variable (A, B, A$ etc.) is used for the first time. Since all variables must be stored somewhere, if it had not been used before, the computer has to make room for the new variable by moving all the arrays up in memory by 7 bytes. With large arrays this can take a noticeable amount of time. The problem can be overcome by first defining all variables at the start of a program by, for example, A=Ø:B=Ø:C=Ø etc. Alternatively, the DIM command can be used instead by: DIM A,B,C etc. Note that this in fact does the same thing as the normal DIM - it simply assigns memory space for these variables, or in other words, sets A, B, and C to zero.

3) "Garbage Collection"

Space limitations prevent me from explaining this phenomenon. We refer our readers to page 144-146 of 'Programming the Commodore 64' by R.C. West. However Basic 7, as used in the 128, does not suffer from this problem.

--ooOoo--

C-64 TRIVIA

by David Lee

REPEATING KEYS:- To get character keys to perform like cursor or space keys:

```
POKE 650,128
```

Disable the repeat, including cursor,space and inst/del:

```
POKE 650,127
```

To return to normal:

```
POKE 650,0
```

SAVE DISABLE:- This POKE will allow you to disable the SAVE COMMAND, the second POKE will return it to normal:

```
POKE818,32
POKE818,237
```

FIND DATA ERROR:- The next time you get a ILLEGAL QUANTITY ERROR when reading DATA, try the following in direct mode:

```
PRINT PEEK(63)+256*PEEK(64)
```

This will give you the line number of the data error. To go one step further, put the following into the program, with the appropriate line number:

```
100 READ A:PRINT A,PEEK(63)+256*PEEK(64)
```

When the program is run it prints both the data and the line on which it occurs. When all the data checks out just remove the line, and replace with the original READ statement.

--ooOoo--

FOR SALE

300 BAUD MODEM, Auto Dial & Auto Answer - Ideal for Bulletin Board Direct Connect incl. Software - $125.00 (+$5.00 for Postage)
1200 BAUD MODEM - Full Duplex - Auto Dial - Auto Answer - $350.00 (+$5.00 Postage)
RS232 INTERFACE - User Port to True RS232 - $35.00 (+$2.00 Postage)

Contact Lester Bennett on 800 1243 before 8 pm,
or P.O.Box 460 - Sunnybank - Q'ld - 4109

Cockroach Turbo-64 Editor (Disk) c/w Labels.
New price $47.00 - Sell for $30.00 ono + $2.00 Postage.

Contact Peter Brewer on (075) 631070 (a.h.) or P.O.Box 99 - Boonah - 4310

VIC-1010 Expansion Module - 6 Slot - Built in power supply - $40.00
VIC-1212 Programmers Aid Cartridge - With Manual - $7.00
ABACUS Tiny Basic Compiler - Tape based c/w Instructions - $2.00
BOOKS: VIC Progr. Ref. Guide $5.00 - Compute's 1st Vic Book $5.00
Tricks for Vics - Vic Programs Vol.1 - Innovative Computing: $3.00 each

Contact Walter Pidwarko on 275 9462 (work) or 345 9153 (home)

Megatransfer & Megatransfer IV + Utilities (Fast Format - Fast Copy - Disk to Disk - Disk to Tape). Efficiently transfers almost all Tapes to Disk (2 Disks). Original English imports $70.00

Contact Jamie Cooper, 19 Bottlebrush St., Allingham via Ingham, Qld, 4850
or Phone (077) 778870 (a.h.)

3032 PET Computer c/w 'Toolkit' & 'Disk-O-Pro' Chips
Ideal learners machine - $150.00

Contact Phil Lyon on 398 8063

Commodore RS232C Terminal Type Interface V.G.C. - $35.00

Contact Heather Todd on 229 7713 (a.h.)

Modem: Modcom 1 incl. phone - $100.00
Eprom Card for Vic-20 c/w Mach. Code Monitor & Fast Cassette Load Eproms and instructions - $35.00
2 Cockroach Write Light Kits for 1541 Disk Drive - $15.00 each

Contact Barry Wilson on 399 6204 (a.h.)

WANTED TO BUY

PRINTER in Good Condition to suit C-64

Contact Cor Geels on 263 2839 (a.h.)

--ooOoo--

WEATHER MAPS ON THE C-64

by Eddie Brook

Here is something else (a bit different) to use your computer for, namely to print out weather pictures (Facsimile) or written messages (Radioteletype). The equipment needed is:

(1) A Shortwave Radio(covering from 2Mhz to 30Mhz with SSB.
(2) A Demodulator ("Listening Post" from Australian Electronics Monthly July 1985).
(3) A Computer.
(4) A program.

Now it's the last item (the program) that we can help you with! It was written by Tony Woods VK6ATW and is in machine code. He was limited to 320 dots per line and 200 lines on-screen to display a FAX picture. Most FAX transmissions have around 567 lines per picture,so only every third line is printed. He also only allows 256 bits per line to be printed, to be able to present a circle as a circle and not as an oval shape. This is why on the C-64 the writing on the FAX picture is not legible.

The weather pictures arrive via satellites and are re-transmitted by the Australian Bureau of Meteorology's own stations, AXM (near Canberra) on 5100 and 11030 Khz, AXI (near Darwin) on 7535 and 10555 Khz, and also FAX from New Zealand on 13550 Khz. The picture is built up a line at a time an takes about 10 minutes to complete - afterwards it can be saved to disk for later viewing.

A program I use to view the saved pictures with is "Flexy Slide Show". This can be found on TPUG disk 129, but the saved pictures must first be renamed to an 'FDDF' file.

In RTTY mode printed messages can be received on 45,50,75,110 baud - 50 baud being the most popular. English news services from Agence France Press (AFP) from Hong Kong on 7542.5 and 10730.6 Khz, The Central News Agency in Taiwan on 7695 and 13563 Khz, Reuters on 6845, 9120, 10960 or 14514 Khz, TAS (Russian News Service) on 6870, 7760, 10270, 13410 Khz, Xinhua News Agency (People's Republic of China) on 7250, 9491 Khz in the middle of day or, on 11680, 14923 Khz in the afternoons, as well as KCNA (North Korean News Agency) on 13780 Khz.

These are just a few that can be received if you are interested in this side of computing.

We have a radio Sub-Group so you could contact either Ron Rich (355 2623) or myself Eddie Brook (349 4394). If enough interest is shown I will arrange a demonstration either at our Group- or Workshop meeting.

--ooo--

Eddie's programs will shortly be available on a new Public Domain disk.

Editor

--ooOoo--

LETTERS TO THE EDITOR

Dear Sir,

I was surprised to read Stinger's criticism of "Cursor".
"Cursor" provides the most objective assessment of software & hardware that I am aware of and it carries more hard information per dollar than any other publication. Not living in Brisbane , I can't judge the validity of Stinger's other criticisms but if they are as invalid as the abuse he pours on "Cursor" than the club must be running very well.

Tony Moore (Townsville)

Thanks for your kind words Tony. Despite the fact that I have been able to shoot some of Stinger's criticisms down in flames, I am still very grateful for his comments. The simple fact is that I rather receive some negative comments than no comments at all!
The one important thing that Stinger has done for me is to make me look in a more critical fashion at the computer scene in general, and that's probably not a bad thing. This does not mean that every issue of this newsletter will be full of whinges or attacks on dealers and Commodore! However, under present economic conditions, I feel that is our duty to look after the financial interests of our members. If in the process we have to step on a few toes - well, so be it.

--ooo--

With the club membership now being in excess of 600 there must be many like me , owning a C-64, and perhaps a disk drive. As for the purchase of a disk drive it seems to be a near foregone conclusion that such will be a Commodore 1541.
Venturing further in the direction of using the C-64, other than for playing games, one starts to contemplate the purchase of a printer. Commodore is marketing a simplistic but reasonably solid model printer under the label 803, and at round about $300.00, at a moderate price. If one wants a printer with a few more features then of course one has to outlay a few more dollars, and apart from Commodore the rest of the world seems to be into printers, but the field seems to turn into a virtual minefield, with or without interfaces, graphics capabilities, coded with M's, P's, and X's like sports cars and numbers in the thousands, and even 'modes' that have nothing to do with the latest fashions.
Every now and then you, as Editor, or some other old hand (computer wise) gets hold of a particular printer and we get a summary with a lot of worthwhile information, but there is no comparison unless one has kept all copies of CURSOR, and that again only applies to members who computerised long, long ago. Enquiries in shops usually result in answers suited to the person who has been buying printers all his life.
It would be nice if whatever printer is being tested, the relevant information could be divulged in a standard type of format and perhaps a perusal of past copies of CURSOR could be a basis for a small booklet on the scale of "Starting with Disk Drives". Updates could be in the form of loose leaf, and purchasers of such a booklet could at a later stage bring their issue up to date.
It would be in the dealer's and retailer's interest to supply relevant information (in the standard format) of any new printer they intend to put in stock, and also those brands on the shelf that they feel should have sold better. Such information, if supplied by a retailer, could then mention the name of the firm and perhaps the approximate price bracket.

Similarly, any of our members who is proud of his machine and feels that more members could be interested in his type of printer, could supply the necessary information in standardised form. If you think that something like this is feasible, and I can be of any help in getting it going, please let me know.

Cor Geels

Dear Cor,

First a word or two about disk drives. Yes, it's probably safest to stick with Commodore. Eighteen months ago the Skai drive was all the rage, but supplies seems to have dried up in Queensland. It was quite a nice drive, but members who had purchased it from down south have reported servicing problems, so back to Commodore. The 1541C drive released in the States looks externally like the 1541, except for the colour, but some important changes have been made internally. A built-in sensor prevents the head banging which is caused by quite a few copy-protected programs such as Easy Script. This will considerably reduce alignment problems. Australian release date unknown at this stage. Some members have picked up 1570 drives at very reasonable prices. These drives are compatible with most C-64 software (see last month's issue of Cursor) and could be a worthwhile alternative.

I feel that a 'booklet' on printers is likely to become a nightmarish 'bible' on printers. Precisely because there are no standard specifications it would be well nigh impossible to classify printers, interfaces etc. Take for example 'Epson compatibility'. The only 100% Epson compatible printers are Epson printers, and even their own printers aren't all compatible! My Riteman printer has a so-called 'Plus Mode', which is roughly equivalent to Epson Mode. (Clever people at C-Itoh, by not calling it Epson mode!) By using an Epson print file with the Superscript wordprocessor it works like a charm. However, when I recently tried my printer with GEOS, using an Epson printer file, it did not work!

Another example. Take the case of a printer which employs a separate interface in conjunction with the program 'Doodle'. This program allows you to customize it with several printer files. One of the questions asked is if you want a line feed or not. Now here the fun starts, because your interface will probably have a dip-switch which controls line feeds, and if you are lucky (!) your printer may also have a dip-switch which controls line feeds! By the time you have sorted this lot out (miles of paper wasted with graphics either printed out all on one line or with double and triple line feeds), you are probably ready to try another graphics package like Printshop, Newsroom, etc. Now, if you are very lucky everything will print out beautifully, but don't blame me if you have to start from scratch again. No, the permutations of printers, interfaces, programs, printer drivers etc. are so enormous that it would take months to sort that lot out, and by the time you are finished half the printers and interfaces will either have been discontinued or replaced by new models with different features.

Other questions that spring to mind include such things as: "How long does it take to print out a Doodle / Koala / Printshop etc picture?" or "Does it print out in Commodore or Non-Commodore mode?" or "Is there a buffer in the printer or interface?" or "How long do the ribbons last and what are the replacement costs?" or "What are the servicing facilities like?" etc. etc. The only logical solution seems to lie in a special printer night where as many printers with or without interfaces are on show, and do an 'on-the-spot' comparison.

Ralph De Vries

REVIEW: "MACH 128" - Fast DOS cartridge for C-128

Review Cartridge made available by Sundown Computer Centre - RRP $59.95

by Ralph De Vries

This reviewer is in a bit of a quandary. Those nice people at Sundown Computer Centre of Chermside made two cartridge programs available for review (the other one "Power Cartridge" for the C-64 will be reviewed in next month's issue), and in the case of the present review I can only recommend it with considerable reservations. Hopefully they won't take offence and keep on making new material available for review, as this service is very much appreciated by the group.

Last October Norm Chambers reviewed the "Mach 5" cartridge for the C-64, with which he was suitably impressed, as he purchased one himself! For the benefit of new members "Mach 5" plugs into the cartridge port and by means of simple 'DOS Wedge" - like commands speeds up loading by appr. five times, as well as simplifying all other disk handling commands. Additionally there are some extra features like fast formatting with a 1541 drive, screen dumps to the printer etc. There are some commercial programs which won't load with this utility cartridge but a simple command disables the cartridge, thus allowing a 'conventional' load for those programs that don't like fast loaders. The important thing to remember is that "Mach 5" does NOT SAVE at higher speeds. If you want this feature as well, you are better of with the "Turbo-Rom" which is available through the group. However this utility has to be installed internally in your computer.

Access Software, the manufacturers of the Mach 5 have now decided that it was time to release a new fast loader for the C-128, so now we have "Mach 128". This cartridge comes with a sliding switch on top [for Mach 5 (C-64) or Mach 128 mode - not clearly marked] as well as a reset switch. After testing it on my C-128 and a 1571 drive I regret to have to report that there was no appreciable increase in loading times with commercial programs such as "Superscript", "Superbase", and "Micro Swift". As most commercially written programs for the C-128 make use of the so-called "Burst mode" of loading (and Mach 128 claims to use the same mode) it becomes obvious why it isn't any faster. However, when I tested it out with some conventional Basic 128 programs it didn't load any faster either. So much for the claimed load speed increase of 500%!

If however you do own a C-128 in conjunction with a 1541 drive you may well find that this cartridge is a worthwhile investment as you can expect in increase in loading speed by approximately 300%. Again, if you mainly use your C-128 for running C-64 programs you may well find this cartridge a good investment. One rather nice feature which C-128 owners may like is that there is a command in C-64 mode which allows you to access the C-128 keys (numeric key-pad, TAB, HELP, etc) which normally are not available in C-64 mode.

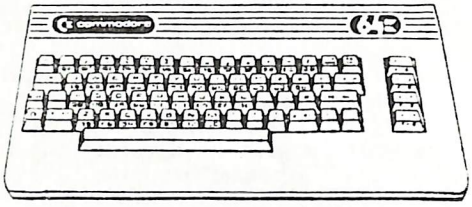
So, if you own a C-128 with 1571 drive (or a C-128D), and you mainly use it in the 128 mode, I can't really recommend this cartridge. Now, if it only had some fast SAVE routines - that would have been another story!

--ooOoo--

commodore COMPUTER

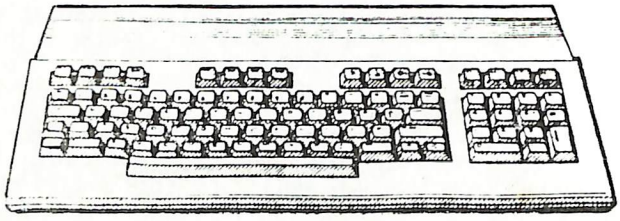
DON'S COMPUTER REPAIR CENTRE

WHITES HILL SHOPPING VILLAGE
SAMUEL STREET - CAMP HILL - 4152
Ph. (07) 395 2211



OFFICIAL
WARRANTY
SERVICE CENTRE

==========



PROFESSIONAL
REPAIRS AND
PRODUCT SUPPORT

==========

FAST TURNAROUND

ALL WORK CARRIES OUR 90-DAY WARRANTY!
SEE US FOR EFFICIENT AND COURTEOUS SERVICE!
SPECIAL RATES FOR USER GROUP MEMBERS!

==========

COMMODORE COMPUTER USERS GROUP (QLD) INC.

==========

D I R E C T O R Y

MANAGEMENT COMMITTEE

President:	GREG PERRY	Ph. 38 3295
Secretary:	NORM CHAMBERS	Ph.341 5651
Treasurer:	LESTER BENNETT	Ph.800 1243
Technical Coordinator:	ROGER HAIGH	Ph.399 8037
Sub-Group Coordinator:	TERRY STEER	Ph.808 2424
Chief Librarian:	MAURICE HAWKYARD	Ph.343 5717
Newsletter Editor:	RALPH DE VRIES	Ph.300 3477

COMMITTEE MEMBERS

Assistant to Treasurer:	JOHN VAN STAVEREN	Ph.372 3651
C-128 Contact:	CLIFF POTTINGER	Ph.277 4520
Vic-20 Contact:	BARRY WILSON	Ph.399 6204
Education Officer:	BILL WEEKS	Ph.341 2823
Sysop:	RAY KING	Ph.208 1409
BBS (Group's Bulletin Board):		Ph.808 2125
Lending Library - Main Meeting Only:	MAURICE HAWKYARD	Ph.343 5717
[This covers Books - Magazines -]	ALAN HORNE	Ph.398 7441
[and Commercial Software for all]		
[Computer Models - Except Amiga!]		
Sales - P.D. Tapes, Books, Accessories:	TERRY STEER	Ph.808 2424
	: MAX BEAN	Ph.208 1225
Library Sales - P.D. Disks (C-64):	BILL BOHLEN	Ph.208 3729

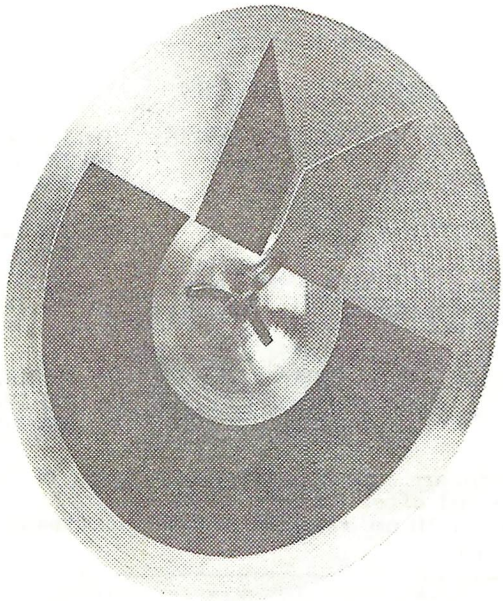
Please Note: The above phone numbers are private numbers, and most of our committee members can only be contacted after hours!
Greg Perry can be reached only between 10 am and 4 pm, and Maurice Hawkyard can be contacted between 9 am and 5 pm.

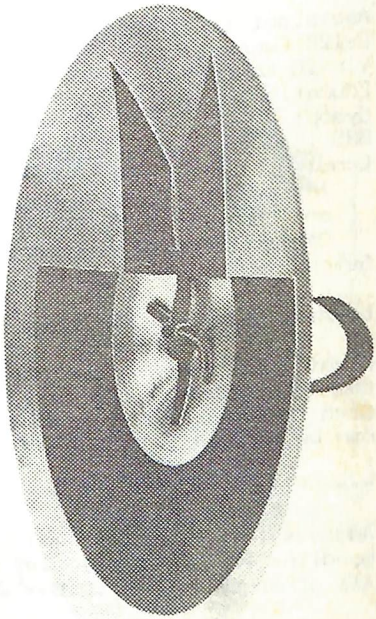
Address for newsletter mail only: P.O. Box 384 - Ashgrove - QLD - 4060.
Deadline for any particular month is the First Tuesday of that Month.
All other mail to: P.O.Box 274 - Springwood - Qld - 4127.

Published by: COMMODORE COMPUTER USERS GROUP (QLD) INC. - P.O.BOX 274, SPRINGWOOD, QLD, 4127.

Printed by: BRISBANE EDUCATION CENTRE - P.O. BOX 84, SPRING HILL, QLD, 4000

The Number One Symbol in Computers. Commodore.





commodore
COMPUTER

Keeping up with you.