VICTORIAN COMPUCOLOR INTERCOLOR USERS GROUP

OCTOBER 1984

PRESIDENT : Ken Winder
Phone (03)848-1422 8 Brindy Crescent
East Doncaster, 3109, Victoria, Australia

SECRETARY/TREASURER : Ted Stuckey
Phone (03)836-8732 c/o "CUVIC", Box 420
Camberwell, 3124, Victoria, Australia

LIBRARIAN : c/o The Secretary
Mail Orders Only "CUVIC", Box 420
Camberwell, 3124, Victoria, Australia

EDITOR : Barry Holt
Phone (03)890-8471 19 Woodhouse Grove
Box Hill North, 3129, Victoria, Australia

THIS ISSUE OF CUVIC INCLUDES:

NEXT MONTH'S ISSUE (Deadline for Copy, 22nd November 1984)
Will include:
"Operating MAILING LIST" (Still no room this month!)
and Ray Halliday's SUPER MENU (features and operation)

THE OCTOBER MEETING will be held on
WEDNESDAY, 14th November 1984
at the **SURREY HILLS NEIGHBOURHOOD CENTRE**
157 UNION ROAD, SURREY HILLS, 3127
("Melways" 46:H,10 - between Bedford Ave & Montrose St)
Commencing at 8.00 p.m.

A G E N D A

1. **Bernie Muldowney's ASMTUT revisited (Barry Holt)**
2. **A solution to the Bowling Draw Problem (Barry Holt)**
3. **Jim Farquhar's TYPE.PRG (see article this issue)**
   automatically entered in best program competition !!!
4. **A Demonstration of IDA (if it arrives on schedule)**

$$ A $200 INCENTIVE $$

FOUR prizes of $25.00 or six disks from the library, are to be offered to the **BEST** program submitted to the library, during the period July 30th 1984 and March 1st 1985, in each of the following categories:

[1] A GAME **[2] A CCII UTILITY**
[3] A HOME APPLICATION **[4] A BUSINESS APPLICATION**

A N D

CUVIC will offer **$100** to the person who produces a realistic **FLIGHT SIMULATOR** for the **CCII** with the majority of the features as specified in the rules.

See Page 1, September issue of CUVIC for details!!

A BRIEF EDITORIAL **Barry Holt**

I will be a Bricklayer's sopranno if I don't get this issue of CUVIC finished, printed and to Ted Stuckey by this time tomorrow. I will purposely leave off the date of this editorial so the members will never know how compromised I am at this time! I am out of the habit - Ted's "Local Mode" took over this column space last month.

End-of-Year Barbeque

This is to remind you that our Christmas Windup for 1984 will be held in place of our regular monthly meeting which is scheduled for Wednesday, December 12th. At the time of writing the venue will be the Surrey Hills Neighbourhood Centre - however this has to be confirmed. See the next issue for details. What is important is that it will be on and will be catered for (within reason!! - remember that the Booze Buses will be out in force)

A donation from Bernie Muldowney

Ted Stuckey has asked me to convey on behalf of CUVIC membership our sincere thanks to our Foundation President and Life Member, **Bernie Muldowney** for his generous donation to the club library of his copyrighted **"COLORSOFT"** software. Bernie's "Colorsoft" series includes:

CATALOG and **CATALOG32** - Disk Library Management Systems
WORDKING - A wordprocessing system
LDIS - A two pass Labelling Disassembler which converts .PRG files to equivalent .SRC files ready for editing or relocating.
ASMTUT - Assembly Language Tutorial Series written for the user with reasonable BASIC skills who wishes to achieve the speed and power of MLP. (This particular package will be re-reviewed at this next meeting -see agenda). This particular series contains ten tutorials the first SEVEN being on disk. The final three tutorials will be serialised in CUVIC at appropriate time intervals.

I believe that the plan will be to release these packages gradually over the next two or three months as time and space in CUVIC permits. There has been a conscientious decision to publish all documentation that accompanies the software within the pages of CUVIC.

I believe that we are all in your debt, Bernie. We are very grateful for your valuable contributions to the CUVIC Library.

KIRKY'S KOLUMN **Alan Kirkpatrick**

NOTES FROM THE OCTOBER MEETING (10/10/84):

This meeting was attended by 14 people.

Ken Winder has the new version of COLORWORD available now. The disk version is more comprehensive than the EPROM version because the latter is limited to 8K. Ken also reminded us that the PRG Assembler & Utilities are very useful for members who are contemplating doing any programming in Assembly Language.

Ken gave us another brief talk on Assembly Language and set us a project to try before the next meeting. The project consists of converting a Machine Language program which is POKEd into memory from a Basic program into its corresponding Assembly Language version. Both the Machine Language program and the 'answer' should appear elsewhere in this CUVIC. I think that attempting to do small projects like this is an excellent way of learning the principles of Assembly Language.

Ted Stuckey told us of some of the trials and tribulations he underwent whilst updating his early model MX80 printer so that it is capable of printing dot matrix graphics. If anyone is contemplating doing the same, I would recommend you contacting Ted before doing so.

Barry Holt showed us a book which he has recently purchased called "Usborne Introduction to MACHINE CODE FOR BEGINNERS" by Lisa Watts & Mike Wharton. This book looks like it would be quite useful for us learners and it is not expensive. Barry hopes to demonstrate the Mailing List program at the next meeting. He gave us another brief demonstration of the Execugraph programs which he had done in more detail at the previous meeting. This set of programs would take a bit of practice to become proficient in their use, since they are very versatile. The instructions were published in the August issue of CUVIC.

LETTERS TO THE EDITOR

Joe Norris (Editor "COLORCUE", Moorestown, NJ, USA) WRITES:

September 30th 1984

Dear Barry

Thank you for your back issue order and subscription to Colorcue. I also appreciate, truly, your words of praise for Colorcue this year. It is a lonely, lonely job, as you implied, and it has been difficult as well as fun for me.

In response to your order, I am returning your check with thanks, since it seems inappropriate to have you pay for materials you deserve "gratis", because of your efforts on our behalf. I am enclosing the complete Vol VI of Colorcue for your use. The extra copy of #3 you may dispose of as you wish.

We are in a state of transition here, regarding CCII publications. With the declining membership, it has seemed foolish to continue with too many publications. CHIP, in Rochester, has been the mainstay for the CCII for seven years. We feel it is important to strengthen CHIP. Since Colorcue dilutes CHIP to a large extent, and because Colorcue has become just too expensive to handle with only 200 readers, the next issue, a double issue - Sept/Oct, Nov/Dec 1984 - will be the last. The editorship is passing to Rick Taubold, in Rochester, with my continued support. Details regarding price and publication dates are now being worked out.

This will leave one publication each in England, the United States and CUVIC in Australia, to whom owners can give more concentrated support. While sad at the demise of anything as valuable as Colorcue has been, it is apparent that the necessity for this step has finally come, and we hope to greet it with good cheer and good energy.

I have drawers fullof material for articles. You are welcome to regular contributions, if you wish. A note to me indicating areas of interest, or particular problems needing solution would be most helpful. I hope that there will be much more interaction among publications in the intermediate future, both to encourage subscriptions to all three major periodicals, and to join memberships in spirit and effort - a requirement to keep the CCII enlivened with continuing energy. My efforts in this direction have not been fruitful, but it must be done if we are to stay together.

As you may know, we are preparing an index to published materials on the CCII for the last issue. It would be splendid to include CUVIC in the listing. I have only the issues from this calender year. If you can be persuaded to forward me all available back issues, on loan for indexing, I will index them, along with FORUM, COMPUKOLOUR, and COLORCUE. I will return the materials to you promptly. If this is to be done, it must be done now, because the last issue is already being prepared for press. I can read Comp-U-Writer and COLORWORD both, so disk material would be useful, however printed matter is much easier to index. Maybe someone has a stack of back issues.

I have enjoyed hearing from you, and I hope we shall be communicating more with one another in the coming months.

With kind regards,

JOSEPH NORRIS

C U V I C
U V I C C
V I **C C** U
I C C U V
C C U V I

19 Woodhouse Grove
Box Hill North
Victoria 3129
Australia
03-890-8471

October 15, 1984

Joseph Norris
"Colorcue"
19 West Second Street,
Moorestown,
New Jersey, 08057
United States of America

Dear Joe,

Thank you for your letter of September 30, I am sorry to hear of the final demise of COLORCUE - it is a pity but we understand the situation. I will contact you at a later date regarding future contributions to CUVIC when I get my mind in order. However, to the immediate business at hand.

Please find enclosed copies of all the CUVIC issues that I have at my disposal. Formal publication of CUVIC (as a newsletter) commenced in January 1982 under the editorship of Keith Ochiltree, I took over the reins in the September of that year and I am still in that position (till Dooms day I expect!). Prior to that time, although we were a very active group - formality was not our scene so I am afraid the small amount of information that was published has not been archived systematically. I do have some of that material which I will publish in future issues.

I also enclose copies of three (and only) issues of the **AUSTRALIAN** COLORCUE which was a weak hearted attempt by the Australian distributor of the CCII to help promote the CCII here after the introduction of 8/79 in 1980. Primarily a rehash of the original COLORCUE it died a premature death after six months of sickness. I guess that their only value is that of history.

There is no need to return any of this material, I have duplicate copies. I only hope that the package arrives before you put your final issue to rest.

With kind regards, Yours Sincerely,

A. B. Holt

R.HALLIDAY (325 Enoggera Rd, Newmarket, 4051, Qld) WRITES: 4/10/84

Dear Barry

Please find enclosed a Demo Disk with "ACTIONMENU" and other programs submitted for comment by CUVIC members.

With reference to page one of the March issue of CUVIC regarding suggestions for the use of utility disks etc., I decided to incorporate additional features into a MENU program to deal with the Directory etc.

Full instructions are on the Demo Disk as INSTRC.SRC which is loaded by INSTRC.PRG.

Should any member request a fully implemented version the cost is $15.00. Please include Compucolor Serial Number; version; and your name and address for the Title heading.

Looking forward to your comments.

Kind regards to all **Ray Halliday**

J.FARQUHAR (Canberra, A.C.T.) WRITES:

Dear Barry

If you're getting desperate for articles, you may like to use the enclosed.

One side of the disk contains all the versions of "TYPE" - .BAS, .PRG, .SRC & .DOC. The other side contains the C.U.W. file of the article. You may also like to include TYPE.PRG in the CUVIC library.

Thanks for your tips on MAILMERGE C.U.W. I eventually got it working satisfactorily.

All the best **Jim Farquhar**

On The PRG Trail (#2) **Ken Minder**

At the October meeting a PRG routine was given to members to decipher and encode into a .SRC format. The program was as follows;

```
10 REM MEMORY FILL WITH 00 TO FFFF, START
20 DATA 17,0,131,1,0,125,62,0,18,19,11,120
25 DATA 177,194,160,130,201
30 FOR P=0 TO 16:READ D:POKE 33434+P,D:NEXT P
40 X=CALL(0)
50 END
```

The above was the basic version, and this routine will start at 8300H and fill the RAM up to FFFF with 0.

The exercise was to convert this basic version. For those who have had a go at it the answer is as follows;

```
        ORG     829AH   ;start address
START:  LXI     D,8300H ;start fill address
        LXI     B,7D00H ;bytes to be filled
FILL:   MVI     A,00H   ;put fill value in A
        STAX    D       ;store A at D address
        INX     D       ;increment D by one
        DCX     B       ;decrement B by one
        MOV     A,B     ;move B into A register
        ORA     C       ;compare C with A
        JNZ     FILL    ;if flag not zero repeat
        RET             ;if flag is zero return
        END     START   ;must use END statement
```

The way to convert the basic data statements to assembly code is to start with the first number, this HAS to be an instruction and from tables you will find it is 'LXI D' for the CCII microprocessor CPU which is a type 8080A.

Further to this first number you should note from the table the number of bytes which follow this code, if any, and this will tell you if the next byte(s) are code or address/numeric data. This LXI D is always followed by two bytes forming a sixteen bit number, thus the next two numbers in the basic data are numbers and belong to the LXI D instruction. This then indicates that the fourth number is another assembly instruction, it is 'LXI B' this also has a two byte number following so that you will now know that the seventh number is the third instruction, it is 'MVI A', this has only one byte to follow and so the next number is numeric data, one byte, and is followed by the next instruction. This is the procedure to decode the data but you will need tables.

The club has tables which print out onto paper on club disk No. 32 together with a HEX/DEC conversion chart, for those who have a printer but no tables. There are some other programs on the same disk. You may notice that instructions with an 'X', called 'extended' are sixteen bit (or two byte) instructions and operate on a register pair. (8 BITS = 1 BYTE). Also note that two byte numbers/addresses are low byte first, high byte second.

If you have the MLDP program you will be able to look into the memory from 829A and see the program in the assembly code, note that the label 'FILL' will be replaced by the address of MVI A,00H in the memory. This is the reason why you cannot recover Labels from a disassembled .PRG program, you will get a 'X01:' etc. label on disassembly, you have to work out what the function of that section is yourself. This is the reason why .SRC files are valuable if you can get the original one for your .PRG program.

Time is one of my problems, and so I will have to stop at this point. Members at meetings will be able to ask questions about this exercise, and I will do my best to assist. As mentioned before, I cannot take you far as I am only at the early stages myself - I thought it better that you were shown the first step, the rest of the stairs has to be climbed by your own effort. Eventually you should get the idea and with each bit of information it will start to shape up like a jigsaw puzzle does, each bit tying more bits together into a better picture. There is nothing to stop others from getting into this tutorial act, the more the better.

A small error crept into the last article in the assembly code, the label must always be identical in the 'JNZ' line to that in front of the INR M in that program, this should have been 'INCRE' in each case, the ':' is not required to end the label following the 'JMP' instruction. I probably missed this due to work load and being tired.

The idea of having a collection of assembly subroutines seems to be valuable, it is easier to alter one of these

for some other purpose than re-invent the wheel. The exercise above can of course be altered to other address ranges, starts and finishes, and ORG'ed at other addresses, just as long as you don't ask it to rub itself out. I hope to provide some more routines later.

The best of luck, this may well provide some holiday fun in January.

MORE ABOUT THE GRAPHICS PRINTER **Ken Winder**

Some more information has been uncovered in respect of the screen graphics printing program. The program has been quite a success in some cases, and had a few problems in other situations.

The first problem is that the program is for those printers having a dot-graphics line specification of 480 dots per line, the AMUST DT80 has unfortunately a 640 dot specification and, although it will work on graphics quite well it does not print characters in the same proportion as the dot-graphics print. This means that screens with mixed dot-graphics and characters are overlaid on two different scales and of course this spoils the printout. Epsons, the AMUST P88, and printers having the 480 specs are correct and give excellent displays to the same scale. There are of course many other printers which will print out correctly. It was noted too, that some EPSONs are made for block graphics and will need conversion if dot graphics is required, this was done by Ted Stuckey and I will leave it to him to tell his own story. The program also sets line feed at 24/216.

The program sometimes appears not to run, I have noted this myself with the disk version but have not found the time to find out why. That it does function - and function well, has been proven in my machine many times. At the moment suspicion is with either the user address getting clobbered, or possibly the top of memory clashing with the graphics program, at this point it is only conjecture since I have not had time to sort this out. The reason(s) will be published as soon as possible.

In EPROM the program runs well indeed, I have it at 4000H, and ESC P is all that is needed to run it. The program is also available at 5000H, ESC \, and runs just as well.

Of the types of printers available I would recommend the P88 over the DT80, it costs a little more but it is well worth it. I have just received news of a new printer available which is much the same as the P88, the SUPER 5 EN-P1091. It has the 480 specification for graphics, plus others, and has also near letter quality printing at 22 CPS, with other modes at 75 and 120 CPS. This printer could be better than the P88.

Enquiries should be made to:
EUROPACIFIC COMPUTERS (INT) Pty. Ltd.,
P.O. BOX 336,
GLEN WAVERLEY, 3150 VIC.
Telephone 233-8652, or 798-7201

This is a club special so you should indentify yourself properly to get the right attention. There is an optional RS232 Interface with a 2K buffer available. Centronic connector is standard. Ribbon lasts 3 million characters, and the price is right.

More data on the graphics as it comes to hand.

VICTORIAN COMPUCOLOR/INTECOLOR USERS GROUP
FINANCIAL REPORT FOR PERIOD 18/6/83 TO 11/7/84

	1982/83	1983/84
INCOME	$	$
MEMBERSHIP FEES LOCAL @ $10.00	640	770
O'SEAS @$25.00	100	125
LIBRARY SALES:		
- COPYING TO OWN DISKS	234	203
- PROGRAMMED DISKS	4805	2103
SALE OF DISKS:		
- SINGLE DISKS 9 @ $4	-	36
- BOXES OF TEN 10 @ $34	-	340
SALE OF PAPER 5 @ $24	-	120
SALE OF BOOKS 4 @ $5	-	20
SALE "THE WORD PROCESSOR" & UPGRADE	-	244
SALE OF MISCELLANEOUS ITEMS	200	30
SALE OF 2 * CCII's	2600	-
ADVERTISEMENTS CUVIC	60	-
RAFFLE	9	-
DONATION FROM K OCHILTREE	130	-
DONATIONS TO N.SOUTH APPEAL	110	-
TOTAL	8888	3991
EXPENSES		
POSTAGE	155	152
PO BOX RENTAL	36	36
REGISTRATION CUVIC	20	20
CHEQUE BOOKS - 2 @ $3.50	-	7
PRINTING CUVIC	394	415
MISCELLANEOUS STATIONERY	20	59
DISKS FOR SEC. & EDITOR, 30 @ $3.40	120	102
DISKS FOR LIBRARY - 60 @ $3.40	360	204
HIRE OF MEETING HALL (FEB TO JUN)	-	63
PURCHASE OF DISKS (430)	3089	1424
PURCHASE OF DISK STORAGE BOXES	70	22
PURCHASE OF SOFTWARE	-	50
PURCHASE OF PAPER - 10 @ $22	-	220
PURCHASE OF BOOKS - 12 @ $5	-	60
PURCHASE OF 2 * MODEMS & ACCESSORIES	-	570
ROYALTIES - "THE WORD PROCESSOR"	-	195
COST OF XMAS BARBECUE	-	80
DONATION TO FORUM	211	-
DONATION TO N.SOUTH	110	-
PURCHASE OF 2 * CCII's	2500	-
PURCHASE OF CHIP DISKS	50	-
SUNDRY EXPENSES	95	-
TOTAL	7230	3679
ASSETS		
SOFTWARE LIBRARY 200 @ $1	150	200
DISKS IN LIBRARY 210 @ $4	600	840
DISKS USED (Sec. & Ed.) 40 @ $3.40	120	136
DISK STORAGE BOXES	20	22
POSTAGE ADVANCE (Librarian)	30	-
2 * MODEMS & ACCESSORIES	-	570
2 * BOXES OF PAPER @ $22 ea.	-	44
8 * BOOKS @ $5 ea.	-	40
DISKS IN STOCK 10 @ $3.40	240	34
BANK BALANCE	1658	2000
TOTAL ASSETS	2895	3886

TED STUCKEY

PROGRAM DEVELOPMENT IN AN ENTIRELY UNSTRUCTURED MANNER
or
HOW I LEARNED TO STOP WORRYING ABOUT STRUCTURED PROGRAMMING AND REDISCOVER THE FUN AND FLEXIBILITY OF BASIC

[Scene: Domestic bliss on a Friday night] "'The A-Team' doesn't turn me on that much darling. I might go and see if I can get an 'Automan' to come out of the old Compucolor. No, I won't stay up late. Yes, I'll be in bed by 11."

[5 minutes later] Hmmm. Woudn't it be nice to be able to use the printer (EPSON MX80 FT) directly as a typewriter so I could just slip in a piece of paper and type a short letter without worrying about loading up a word processor, or use it to complete forms and things, where a word processor is not really feasible. There must be a way. Didn't I see somewhere an article about the TMS 5501 and use of the OUT command? **[After 3 hours rummaging through various back-issues]** Ah! There it is! (FORUM NOV/DEC 81). Looks like OUT 6,n will send the ASCII value of "n" directly to the RS 232 interface and hence to the printer. Let's try it. We'll type in OUT 6,65 to send an "A".

Nothing happens! Now why is that? Out with the EPSON manual. **[After a further 2 hours head-scratching]** Looks like the data is being sent to the printer buffer, but the EPSON won't empty the buffer until it's full or it receives a command to do so. O.K. then, there must be a command to tell the EPSON to empty its buffer without doing anything else. Damn! There's no such command! The closest there is is ASCII 18, which also turns off compressed characters. Let's type in OUT 6,65 followed by OUT 6,18. How about that! An "A" just where we want it.

Let's try a small program with a GET KEY routine (we don't want to hit RETURN after each letter when typing a letter, that is a letter made up of a lot of letters or ... forget it) followed by OUT 6,n: OUT 6,18. The old standby POKE 33278,0 followed by Y=PEEK (33278) in a loop should do the trick. It works! The character is printed on the screen and sent to the printer. We're on our way.

Let's make this thing as close as possible to a real typewriter and at the same time try to utilize most of the printing features of the EPSON.

The only characters we want printed range from ASCII 32 to ASCII 127, therefore we've got ASCII 0-31 as well as 128-255 to tell the printer to do things. Stacks!

First, let's see if we can print in condensed mode and double width mode. The printer recognises "15" for condensed, and the sequence "27,87,1" (ESC "W" 1) for double width. Hmmm. The CCII uses "15" for BLINK/A7 OFF. Well, let's do a little translating for the EPSON. "A7 ON" is the logical key for double width, so we'll use BLINK/A7 OFF to turn it off again. Now what about compressed characters? BLINK ON seems a logical choice, so to keep things systematic, we'll use the key to the left of that (BG ON/FLG ON) to turn off compressed characters. Here we go! IF Y=a THEN OUT 6,b: IF Y=c THEN OUT 6,d etc. (Lines 280-310) Now we're going along like a house on fire. Hey, I didn't know that! I thought the EPSON could print either normal width, OR compressed OR double width. But in fact the two basic modes are normal and condensed, and you can make both of these double width. I like the style of the compressed, double width print - I'll have to use it more often. Hang on! We're using ASCII 18 to empty the buffer and that resets compressed characters. Oh well, let's use a flag (CF) and, if the flag is set, set compressed characters back on again after we've turned them off by OUTing "18".

But what's this? Things are going along nicely with the printer, but the CCII screen is going crazy, blinking and printing big letters etc. If we're gonna do this properly, we'd better make the screen presentation more presentable. The old standby GET KEY routing won't do any more, because it sends everything to the screen. Didn't I read somewhere that the patch on the CAP Electronics music disk will get the ASCII values of keys without displaying them, as well as doing scrolling if you want to? That feature might come in handy later, so let's get it out. Ah, thank you Mr CAP: you've included REMs which tell me what values to use in the CALL statement. Now we use Y=CALL(2) to get the character. Great! To display the letter on the screen we can either PRINT CHR$(Y) or better still PLOT Y. Now we can control the visual presentation. Boy, this is getting pretty sophisticated.

Let's try underlining and italics. We should never need SHIFT _ (CRT) or SHIFT ^ (USER) so let's use them to turn the underlining on and off (lines 320 & 330). What about italics? Ah, I haven't used the colour keys yet. Let's use the left column of keys to turn things on and the right column to turn them off, and we'll start at the top row using the BLACK and BLUE keys for italics (lines 340 and 350). Works fine! Maybe I should have used these keys for the other functions.

But what's this? Has my typing deteriorated to such an extent that I'm typing words backwards, like "TI" instead of "IT", and "WIHT" instead of "WITH"? I know its 2 a.m. but this is ridiculous. Damn! Poor old BASIC is so slow finding its way through all these IFs and THENs that I can catch it up! Well I'm not going to do the whole thing again in Assembly (mainly because I can't), so let's see if Peter Hiner's Compiler can help. (10 minutes later) Marvellous! Thanks Peter. Now the program can keep ahead of me even at my blazing speed of 30 word a minute.

["Coming to bed soon darling"]. I know its late but let's continue for a bit. What else can a typewriter do - backspace of course - a must for typists like me. Now let's see, ASCII 8 will empty the buffer then backspace the print head one space. Huh? It doesn't work! Now, OUT 6,65: OUT 6,8: OUT 6,66: OUT 6,18 will print a "B" on top of the "A", but OUT 6,8: OUT 6,18 and all other combinations won't do it. Obviously, if there's nothing in the buffer to start with, it won't backspace, and there's nothing in the buffer because we're emptying it every keystroke. Damn, they didn't tell us that in the fine print. If I'd known this at the start, I would have used this (OUT 6,8) to empty the buffer instead of OUT 6,18. Then I wouldn't have needed to use the flag for compressed characters. Well we need a backspace, so we'll just have to simulate it, and to do that we'll need to keep track of our position via a character count. Let's increment "C" each time we print a character, then when we want to backspace, return to the beginning of the line (carriage return without a line feed) and loop up to the number of characters printed before we

print again (line 260). Have to remember to reset C to 0 after each carriage return. Ah! That does the trick, but it's a bit dodgy when we mix character widths on the same line. That's a pity, but I think that's the best I can do. We'll just have to remember that limitation.

Now, what about a line feed. That should be simple enough. Let's see, ASCII 10 empties the buffer and does a line feed and Resets the buffer pointer to zero - that I think is a carriage return & we don't want that. ASCII 11 might do the trick. That's just supposed to do a single line feed and hopefully nothing else. Let's try it. Bother! (says he, euphemistically). ASCII 11 acts just like an ASCII 10 and does a carriage return as well. Oh well, we'll just have to use the character count again and loop up to where we should be. This has the same drawback as backspacing when using a mix of character widths, but we'll just have to remember that too. What a pity!

Now, woudn't it be nice to colour-code the scren to reflect the print mode we are using. Let's see, we have two basic modes (Normal [N] and Compressed [C]) and we can underline [U], do double width [D], plus italics [I]. So we need colours to reflect N, ND, NU, NI, NUI, NDI, NDUI, ... etc. that's 2^3 isn't it? 16 colour codes! That's ridiculous! I'll never remember what's what. Let's keep it simple and use GREEN for normal, YELLOW for compressed, and inverse these for underlining. Then we can put a message on the screen to show what mode we're in. But to do that, we'll need a message line which will stay on the screen while the rest of the screen scrolls. Ah! the old scrolling patch. I've never used it before so let's give it a go. We'll use line 31 for messages, line 30 as a blank line and scroll lines 0 to 29. Now let's see (it's a pity Mr CAP didn't put in REMs to explain these) - X and Y are obviously the coordinates of the top left corner of the area to scroll, W and H are the width and height. What's C? I dunno. Maybe it's the number of lines to scroll each time. Let's keep it at 1. Hang on! We've already used Y and C so we'll call them YY and CC instead (Line 210). Now we need a line count to tell it when to scroll. LI seems and obvious choice for this. O.K., now we increment LI every time we do a line feed or a RETURN, and when it reaches 29, XC=CALL(1) does the trick (Line 400). Terrific!

Now here's a bonus. To print messages on the message line, we lose track of where we are on the screen, but LI and C are keeping track of our line and column number, so we can use PLOT 3,C,LI to get us back to the right spot again.

Now a bit of tidying up and a few instructions and we're done! We can simulate the EPSON as a fairly effective typewriter. Hmmm. The EPSON seems to be working overtime. I hope this constant emptying of the buffer isn't causing any damage. Maybe it's best if I don't use this program just to make sure.

I wonder of CUVIC readers may be interested in this? I'll send a copy of the compiled version for the library plus a copy of the listing in case someone is silly enough to adapt it for other printers.

"What's that darling? Don't worry, you stay in bed and I'll bring you some breakfast". [Later] **"I'm feeling a bit weary honey. Do you mind if I have a little lie-down for a while?"**

PROGRAM LISTING

```
  0 GOTO 60000
100  CLEAR  :PLOT 12,15,3,10,12,6,2:PRINT "DO YOU WANT
   INSTRUCTIONS (Y/N)? ";:Y= CALL (2):IF NOT (Y= 89 OR
   Y= 78) THEN PRINT :GOTO 100
110 PRINT CHR$ (Y):IF Y= 89 THEN GOSUB 1000
120  PLOT 12,15,3,10,12,6,2:PRINT "SET PAPER & TURN ON
   PRINTER"
130 PLOT 3,10,13:INPUT "HIT <RETURN> WHEN READY....";A
140  PLOT 3,10,14:INPUT "ENTER PAPER LENGTH (INCHES):
   ";PL:OUT 6,27:OUT 6,67:OUT 6,0:OUT 6,PL
150  PLOT 3,10,16:INPUT "ENTER NO. OF LINES FOR SKIP
      OVER PERF: ";SK:IF SK< > 0 THEN OUT 6,27:OUT
   6,78:OUT 6,SK:GOTO 170
160 OUT 6,27:OUT 6,79
170 PLOT 3,10,18:INPUT "ENTER LINE SPACING (1,1.5,2,3):
   ";SP:OUT 6,27:OUT 6,65:OUT 6,SP*12
180 PLOT 27,24,12,14,27,18,7,15,6,2
190 PLOT 3,0,31:PRINT "NORMAL    "
200 TM= PEEK (32940)+ 256* PEEK (32941)+ 1
210 X= 0:YY= 0:H= 30:W= 64:CC= 1:GOSUB 65040:REM SCROLL
220 CO= 2:UL= 1:REM  COLOUR CODE
230 Y= CALL (2):IF Y= 27 THEN PLOT 27,13,27,64:POKE
   33265,0:END
240  IF  Y= 10  THEN  LI= LI+ 1:PLOT 10:OUT 6,10:IF C>0
   THEN FOR I= 0TO C- 1:OUT 6,32:NEXT :GOTO 230
250 IF Y= 13 THEN 370
260 IF Y= 26 AND  C>0  THEN  PLOT  26:OUT  6,13:C=  C-  1:IF
      C>0  THEN  FOR  I=  0  TO  C-  1:OUT 6,32:NEXT :GOTO
   230:REM  BACKSPACE
270  IF  Y=  25  THEN  C= C+ 1:PLOT Y:OUT 6,32:OUT 6,18:GOTO
   380:REM  CURSOR RIGHT
280  IF  Y=  31  THEN  OUT  6,15:CF= 1:CO= 3:PLOT3,0,31,6,3:
         PRINT  "COMPRESSED":PLOT  3,C,LI:GOTO  230:REM
   COMPCHARS ON
290 IF Y=  30  THEN  OUT  6,18:CF=  0:CO=  2:PLOT3,0,31,6,2:
        PRINT  "NORMAL       ":PLOT  3,C,LI:GOTO  230:REM
   COMPCHARS OFF
300  IF  Y=  14  THEN  OUT  6,27:OUT 6,87:OUT 6,1:LF= 1:PLOT
      3,12,31,6,1:PRINT  "DOUBLE  WIDTH":PLOT  3,C,LI:GOTO
   230:REM  LARGE CHARS ON
310 IF Y= 15  THEN  OUT  6,27:OUT  6,87:OUT  6,0:LF=  0:PLOT
      3,12,31,6,1:PRINT  "              ":PLOT 3,C,LI:GOTO
   230:REM  LARGE CHARS OFF
320  IF  Y=  127  THEN  OUT  6,27:OUT  6,45:OUT  6,1:PLOT
       3,27,31,6,1:PRINT  "UNDERLINING":PLOT  3,C,LI:UL=
   8:GOTO 230:REM  ULINE ON
330  IF  Y=  126  THEN  OUT  6,27:OUT  6,45:OUT  6,0:PLOT
      3,27,31,6,1:PRINT  "               ":PLOT 3,C,LI:UL=
   1:GOTO 230:REM  ULINE OFF
340  IF  Y=  16  THEN  OUT  6,27:OUT  6,52:PLOT3,41,31,6,1:
     PRINT  "ITALICS":PLOT  3,C,LI:GOTO  230:REM  ITALICS
   ON
350 IF Y=  20  THEN  OUT  6,27:OUT  6,53:PLOT  3,41,31:PRINT
   "       ":PLOT 3,C,LI:GOTO 230:REM  ITALICS OFF
360 IF Y< 32OR Y> 127 THEN 230
370  PLOT  6,(CO* UL):C=  C+  1:PLOT  Y:OUT 6,Y:OUT 6,18:IF
   Y= 13 THEN C= 0:OUT 6,11:OUT 6,18:PLOT 10:LI= LI+ 1
380 IF CF= 1 THEN OUT 6,15
390 IF C> 63 THEN C= 0:LI= LI+ 1
400 IF LI> 28 THEN XC= CALL (1):LI= LI- 1:PLOT 28
```

```
410 GOTO 230

1000 PLOT 12,3,15,1,14,6,5:PRINT "T Y P E W R I T E R
     I N S T R U C T I O N S"
1010 PLOT 15,6,3:PRINT :PRINT :PRINT "THIS IS A FAIRLY
     ROUGH AND READY PROGRAM TO MAKE THE EPSON"
1020 PRINT "MX80 FT FUNCTION AS A NORMAL TYPEWRITER. A
     SUMMARY OF THE "
1030 PRINT "COMMANDS ARE AS FOLLOWS:"
1040 PRINT :PRINT TAB( 10);"BLINK ON                    =
     COMPRESSED CHARACTERS ON"
1050 PRINT TAB( 10);"BG ON/FLG ON                       =
     COMPRESSED CHARACTERS OFF"
1060 PRINT TAB( 10);"A7 ON                        = DOUBLE
     WIDTH ON"
1070 PRINT TAB( 10);"BL/A7 OFF                    = DOUBLE
     WIDTH OFF"
1080 PRINT TAB( 10);"SHIFT (CRT)                        =
     UNDERLINING ON"
1090 PRINT TAB( 10);"SHIFT (USER)                       =
     UNDERLINING OFF"
1100 PRINT TAB( 10);"BLACK KEY                    = ITALICS
     ON"
1110 PRINT TAB( 10);"BLUE KEY                     = ITALICS
     OFF"
1120 PRINT :PRINT "THE LEFT, DOWN AND RIGHT ARROW KEYS
     ALL WORK"
1130 PRINT :PRINT "THE ESCAPE KEY WILL RETURN YOU TO
     BASIC"
1140 PLOT 3,10,30:INPUT "HIT <RETURN> TO
     CONTINUE....";A:RETURN


60000 TM= PEEK (32940)+ PEEK (32941)* 256

60001 IF TM> 65500 THEN 60006

60002 POKE TM+ 1,50:POKE TM+ 3,15

60003 REM 60027 CHECKS IF MACHINE LANGUAGE ALREADY
      LOADED

60004 A= 0:FOR N= TM+ 176TO TM+ 182:A= A+ PEEK (N):NEXT
60005 IF A= 1445 THEN 60015

60006 AD= 32940:TM= TM- 200:DA= TM:GOSUB 60016:CLEAR
      100

60007 TM= PEEK (32940)+ PEEK (32941)* 256

60008 PRINT "WORKING"

60009 RESTORE 60019

60010 REM 60050 LOADS MACHINE LANGUAGE

60011 FOR N= 1TO 200:READ A:IF A> 255 THEN N= 220:GOTO
      60014

60012 IF A< 0 THEN DA= TM- A:AD= TM+ N:GOSUB 60016:GOTO
      60014

60013 POKE TM+ N,A

60014 NEXT

60015 AD= 33283:DA= TM+ 8:GOSUB 60016:GOTO 100

60016 Z1= INT (DA/ 256):Z2= DA- Z1* 256

60017 POKE AD,Z2:N= N+ 1:POKE AD+ 1,Z1:RETURN


60018 REM MACHINE LANGUAGE DATA

60019 DATA 50,0,15,0,0,0,0,123,254,0,194,-121,30,8,243

60020 DATA 58,-6,87,58,-7,254,0,194,-33,122,50,-7,42,-1

60021 DATA 34,-49,42,-3,34,-46,33,-22,1,50,0,11,120,177

60022 DATA 194,-51,123,238,2,211,4,95,43,124,181,194

60023 DATA -48,58,-5,254,0,202,-100,61,50,-5,58,-49,130

60024 DATA 50,-49,58,-7,103,122,50,-7,84,195,-45,50,-7

60025 DATA 50,-4,50,-2,62,15,50,-3,62,50,50,-1,251,201

60026 DATA 254,1,194,-158,33,0,0,6,4,14,17,17,128,0,25

60027 DATA 126,17,128,255,25,119,35,0,13,194,-133,17,8

60028 DATA 0,25,5,194,-131,201

60029 DATA 254,2,194,-182,245,229,197,205,36,0,194,-166

60030 DATA 95,175,87,50,255,129,193,225,241,201,201

60031 DATA 300

65000 REM

65010 Z1= INT (DA/ 256):Z2= DA- Z1* 256

65020 POKE AD,Z2:N= N+ 1:POKE AD+ 1,Z1:RETURN


65030 REM SCROLLING SUBROUTINE

65040 DA= 28672+ 128* YY+ X+ X:AD= TM+ 126:GOSUB 65010

65050 DA= 128- W- W:AD= TM+ 149:GOSUB 65010

65060 POKE TM+ 129,H- 1:POKE TM+ 131,W* (CC+ 1)

65070 POKE TM+ 143,35* (1- CC):RETURN
```

Jim Farquhar

MELBOURNE COMPUCOLOR USER GROUP---- MEMBERSHIP LIST — As At: Thursday, October 11, 1984

NUM	NAME	MAILING NAME	PHONE	TOWN	COMP TYPE	VERS	MEM	K/B	DSK	L/C	SND	MDM	PRINTER	
101	Terry	T. Rickard	03 3809007	Brunswick	CC II	8/79	16k	STD	1	Y	Y	N		PAID
102	John	E. Spencer	054 270365	Gisborne	CC II	8/79	32K	EXT	2	Y	N	N	Microline 80	PAID
103	Robert	R. Ferguson	03 387-7446	Parkville	CC II	8/79	32k	EXT	1	N	N	N		PAID
104	Ian	I. Morton	03 4354202	Macleod	Int 3652	9/80	32k	DLX	2	N	N	N	TALLY T 1705	PAID
105	Peter	P. Webster	051 47-1748	Maffra	Int 3651	8/79	32k	EXT	2	N	Y	N	Microline 80	PAID
106	Milton	M. Fox	057 214271	Wangaratta	CC II	6/78	16k	EXT	2	Y	Y	N	DEC LA36	PAID
107	John	J. Mangani	03-363-4124	Ardeer	CC II	8/79	32k	DLX	2	N	N	Y		PAID
108	Terry	T. Crawford	560-3661	Glen Waverley	CC II	8.79	32k	STD	1	Y	Y	N	Amust DT 80	PAID
109	Alan	A. Kirkpatrick	03 5705885	Oakleigh	CC II	8/79	32k	EXT	2	N	Y	N	Microline 80	PAID
110	Misha	M. Verplak	053 467535	GLENLYON	CCII	8/79	32k	STD	N	N	N	N	N	PAID
111	Howard	H. Rice	03 277-2957	Mount Waverley	Int 3600	2/79	64K	DLX	2	Y	N	Y	Epson MX80FT	PAID
112	Alan	A. Lewis	091 895367	Paraburdoo	CC II	6/78	32k	DLX	2	Y	Y	Y	Epson MX80	PAID
113	Trevor	T. Wilson	098 81-1985	Narrogin	CC II	8/79	32k	STD	1	Y	Y	N	NEC PC 8023	PAID
114	Ted	E. Stuckey	03 8368732	Canterbury	CC II	6/78	32k	EXT	2	Y	Y	N	EPSON MX80/3	PAID
115	Robyn	Mrs R Groffett		Glen Waverley	CC II	8/79	32K	EXT	1					
116	Ken	K. Winder	03 8481422	Doncaster East	CC II	8/79	32k	DLX	2	Y	Y	N	Microline 80	PAID
117	John	Attn: John Foster	.	Moera, Lower Hutt	CC II	6/78	32k	DLX	2	N	N	N	Yes	PAID
118	Eric	E. Van Rhijn	03 3692035	Laverton	CC II	8/79	32k	DLX	2	N	Y	N	Cent/Trendcm	PAID
119	Neville	Mr N Schefe	07 269-6328	Brackenridge	CC II	8/79	32k	EXT	1	N	N	N	Microline 80	PAID
120	Peter	P. Nugent	03 898-6326	Box Hill North	CC II	6.78	16k	STD	1	N	N	N	None	PAID
121	Barry	A. Holt	03 898471	Box Hill	CC II	8/79	32k	EXT	2	Y	Y	N	Epson MX80	PAID
122	Neil	N. Ashman	03 221-5745	Wantirna	CC II	8/79	32k	EXT	1	N	N	N		PAID
123	Leon	L. Moscovitch	03 857-8195	North Balwyn	CC II	8/79	32K	DLX	1	Y	Y	N		PAID
124	Barry	B Frazer		North Ryde	CC II									PAID
125	Gar.	Mr G. Butler	03 762-4699	The Basin	CC II	6.78	32k	DLX	2	Y	Y	N		PAID
126	Ray	R. Halliday	07 356-4236	Newmarket	CC II	6/78	32k	DLX	2	Y	Y	N	ANADEX 9500	PAID
127	Adrian	A. Fox.		Latham										PAID
128	Reg	R McLean	03	North Melbourne	CC II	8/79	32k	STD	1	N	N	N		PAID
129	Terry	T. Wardle	08 260-3693	Ingle Farm	CC II	8/79	32k	DLX	1	Y	N	N	Microline 80	PAID
130	Greg	G. Kelly	052 48-1116	Newcomb	CC II	8.79	32k	DLX	1	N	N	N		PAID
131	Jim	J. Toohey	03 8577263	North Balwyn	CC II	6/78	32K	STD	1	N	N	N	LA 34	PAID
132	Mark	M. Gearing	03 878-9073	Nunawading	CC II	8/79	16k	EXT	1	N	Y	N	Epson MX 80	PAID
133	Gavin	G. Arndt	03 2887011	Burwood	CC II	8/79	16k	EXT	1	N	Y	N		PAID
134	Brian	B. O'Sullivan	08 352-3296	Torrensville	CC II	8/79	32k	DLX	1	Y	N	N	Omega 996	PAID
135	Gary	G. Osborn	03 877-3231	Blackburn South	CC II	6/78	32k	EXT	2	Y	Y	Y	Microline 80	PAID
136	Vincent	V. Hunt	03 509-0646	Armadale	CC II	6/78	32K	STD	1	N	N	N		PAID
137	Paul	P. Irna	850 6173	Lower Templestowe	CC II	8/79	32k	EXT	1	N	Y	Y		PAID
138	Peter	P. G. Standen	049 43-1030	Charlestown	CC II	6/78	32K	DLX	1	Y	Y	N	Microline 80	PAID
139	Austin	A. Vaughan	062 54-9896	Higgins	CC II	8/79	32k	EXT	1	Y	Y	N		PAID
140	Brian	B. G. Griffiths	Nil	Dalby	CC II	8/79	32k	EXT	1	N	Y	N	HP 82905 A	PAID
141	Les	L. Paget	03 569-8774	Cnadstone	CC II	8/79	32k	EXT	1	N	N	N	N	PAID
142	Ron	Ronald Mackenzie		Commack, NY, 11725	CC II	8/79								PAID
143	Hugh	H. Darragh	07 370-7480	Taringa East	CC II	8/79	32K	DLX	2	N	N	N	Microline 80	PAID
144	Michael	M. Kozminsky	03 5966422	North Brighton	CC II	8/79	32k	DLX	1	Y	Y	N	Epson MX80	PAID
145	John	J. Garner		Wynyard	CC II	8/79	32k	STD	N	N	N	N		PAID
146	Ross	R. Power	560-7399	Mulgrave	CC II	8/79	32K	DLX	2	Y	Y	N	Printerm	PAID
147	Bill	W. Roussac	03 380-4701	Brunswick East	CC II	6/78	40k	STD	1	Y	Y	Y	Microline 80	PAID
148	Peter	P. Scurrah	848-2414	Doncaster East	CC II	8/79	32K	STD	1	N	Y	N		PAID
149	David	D. Haskin	03 523-5058	North Caulfield	CC II	8/79	32k	DLX	2	N	N	N		PAID
150	Russel	R. Knox	03 728-1008	Kilsyth	CC II	8.79	16k	STD	2	N	N	N	-	PAID
151	Bob	R. Sterling	03 876-3496	North Ringwood	CC II	8.79	32k	EXT	2	N	N	N	Tally T 1705	PAID

MELBOURNE COMPUCOLOR USER GROUP---- MEMBERSHIP LIST — As At: Thursday, October 11, 1984

NUM	NAME	MAILING NAME	PHONE	TOWN	COMP TYPE	VERS	MEM	K/B	DSK	L/C	SND	MDM	PRINTER	
152	Jim	J. Farquhar	062 41-2870	Kaleen	CC II	8/79	32K	EXT	1	N	Y	N	Epson MX80	PAID
153	ken	K. Kerrison	062-733743	Pialligo	CC II	8/79	32k	EXT	2	Y	Y	N	Microline 80	PAID
154	George	G. Newnham		Cairns	CC II	8/79	32K	EXT	1	N	Y	N	Microline 80	PAID
155	Ken	K.D. Smith	03 749-5579	Hoppers Crossing	N*STAR	CPM	64k	DLX	2	Y	N	Y	DS180	PAID
156	Ian	I. Holthouse	03 233-8652	Vermont South	CC II	8/79	32k	EXT	2	N	N	N	Microline 80	PAID
157	Bill	W. Stenzler	516 931-4684	Plainview, NY, 11803	CC II	8/79	16k	DLX	1	Y	Y	Y	Base 2	PAID
158	Peter	Mr P. Michell	03 561-1341	Glen Waverley	CC II	6.78	32k	DLX	2	N	Y	N	Printherm	PAID
159	Geoffrey	Dr G. Price	.	Box 54, Mt Waverley	CC II	8/79	32k	STD	1	N	N	N		PAID
160	Frank	F. van der Hulst	04 6605571	Box 30-089 Lowr Hutt	CC II	8.79	32k	STD	2	Y	Y	N	Epson MX 80	PAID
161	Ron	R. McKenny	03 725-6363	Kilsyth	CC II	8/79	32k	STD	1	N	N	N	.	PAID
162	Ian	I. Woodburn	02 44-7175	Turramurra	CC II	8/79	32K	DLX	1	Y	N	N	Microline 80	PAID
163	Rob	R. Priestley	051 44-2856	Sale	CC II	8/79	32k	EXT	2	N	Y	N		PAID
164	Fred	F. D. Haerich		Altamonte Spr 32714	CC II	8/79	32k	DLX	2	Y	Y	Y	Epson MX 100	PAID
165	Haig	H. Jason	03 288-5648	Burwood East	INT 8353	2/79	24k	DLX	2	Y	N	N	Model 800MST	PAID
166	Bert	B. Berzins	03 337-4213	Essendon	CC II	8/79	32k	STD	1	N	N	N	Microline 80	PAID
167	Bruce	B. C. Marshall	045 77-4824	South Windsor	CC II	8/79	32K	DLX	2	Y	Y	N	TANDY LP VII	
168	Bruce	B. Reimers	.	Toowoomba	CC II	8/79	32k	EXT	2	N	N	N	Microline 80	PAID
169	Richard	R. Collins	03 874-8320	Vermont	CC II	8/79	16K	EXT	1	N	N	N		PAID
170	Richard	R. Ferguson		Meipa	CC II	8/79	16k	EXT	1	N	N	N		PAID
171	Des	D. Hill	03 870 1705	Heathmont	CC II	8/79	32k	STD	1	N	Y	N		PAID
172	Roger	R. C. Thompson	03 762-2619	Boronia	CC II	6/78	32k	DLX	1	N	N	N		PAID
173	E.....	E. H. Culph	03 697-1381	South Melbourne	CC II	8/79	16k	DLX	1	N	N	N	TI Silent700	PAID
174	David	D Kerlin	.	Eumundi	CC II	8/79	32k	STD	2	N	N	N	Microline 80	PAID
175	Dave	D J Linnett	.	Bracken Ridge	CC II	8/79	16K	STD	1	N	N	N		PAID
176	Wayne	W. Power	03 699-7148	Albert Park	CC II	8/79	32K	DLX	1	Y	Y	Y		
177	Aub	A. Stark	03 8781679	Nunawading	CC II	8/79	32K	DLX	1	Y	Y	Y		PAID
178	Joe	J. Ardarno		Northcote	CCII									PAID
179	Adrian	A. Fox.		Lathan										PAID
181	Norman	N.M.Johnson	0	JOHNS ISLAND SV29455	CCII	8/79	32k	EXT	2	Y	Y	Y	UNKNOWN	PAID
187	Keith	K. Ochiltree	03 813-3348	East Hawthorn	CC II	8/79	32K	DLX	2	Y	Y	Y	IDS 560	HON
188	Joe Norris	The Editor, Colorcue	609-234-8117	Moorestown NJ 08057										HON
189	-	The Editor, Micom		Canterbury										HON
190	-	The Post Master	03 830-4020	Canterbury										HON
191	Neil	N. Brandie	795-1609	North Dandenong	CC II	0	0 k	0	0	0	0	0	0	HON
192	Bernie	B. Muldowney	057 214567	Wangaratta	CC II	6/78	32K	DLX	2	Y	Y	N	IDS 560	HON
193	Peter	P. Hiner	05827 64872	Harpenden, AL5 2PD	CC II	8/79	32k	EXT	2	Y	N	N	Yes	HON
194	Bill	W. Woods	02 84-6764	Beecroft										HON
195	John	J. Newman	09 299-6153	Darlington										HON
196	Tony	T. Lee	02 449-8824	St Ives										HON
197	Doug	Douglas A. Van Putte		Rochester, NY, 14624	CC II									HON
198	Doug	D. Pankhurst	NONE	Amberly	CC II	8/79	32K	DLX	2	Y	Y	N		HON

```
 ######  #     #  #     #  #     #  ######
 #       #     #  #     #  #     #  #
 #       #     #  #     #  #     #  #
 #       #     #  #     #  #     #  #
 #       #     #  #     #  #     #  #
 ######   #####    #####    #####   ######
```

VICTORIAN COMPUCOLOR INTERCOLOR USERS GROUP

JULY 1985

REGISTERED BY AUSTRALIA POST PUBLICATION No.VBH 5086

PRICE 40 CENTS **MONTHLY**

PRESIDENT : Ken Winder
Phone (03)848-1422 8 Brindy Crescent
East Doncaster, 3109, Victoria, Australia

SECRETARY/TREASURER : Ted Stuckey
Phone (03)836-8732 c/o "CUVIC", Box 420
Camberwell, 3124, Victoria, Australia

LIBRARIAN : c/o The Secretary
Mail Orders Only "CUVIC", Box 420
Camberwell, 3124, Victoria, Australia

EDITOR : Barry Holt
Phone (03)890-8471 19 Woodhouse Grove
Box Hill North, 3129, Victoria, Australia

THIS ISSUE OF CUVIC INCLUDES: **page**

NEXT MONTH'S ISSUE (Deadline for Copy, 19th August 1985)

thcoming issues will contain:

A comparision of Database packages for the CCII.
ASM Tutorial Series Appendices 8 thru 10.
Instructions to Jim Helms "READ" program.

THE AUGUST MEETING will be held on
WEDNESDAY, 14th AUGUST 1985
at the **SURREY HILLS NEIGHBOURHOOD CENTRE**
157 UNION ROAD, SURREY HILLS, 3127
("Melways" 46:H,10 - between Bedford Ave & Montrose St)
Commencing at 8.00 p.m.

The Editor tenders his apologies for this meeting. He will be in Hobart at the appointed time.

DISKS FOR SALE

5.25 INCH SSDD NASHUA DISCS (BULK) @$2.50 EACH
PLUS POSTAGE (MINIMUM OF TEN DISKS POSTED)

KIRKY'S KOLUMN

NOTES FROM THE JULY MEETING (10/07/85):

This meeting was attended by 10 people.

Ken Winder told us of a bug which occurs in earlier versions of the **FASBAS** Compiler (Version 12.25 is OK!). Ken also demonstrated to the meeting, two methods of correcting the bug. The first method involves loading the program, doing a couple of POKES to appropriate memory locations, then resaving the corrected program. **[See Peter Hiner's letter and "fix" this issue, Ed.]** The old version then needs to be deleted from the disk, with subsequent repacking of the disk. The alternative method which Ken showed us made use of **IDISK** which allows you to alter the appropriate sector directly on the disk, thus eliminating the need for saving the new program and deleting the old version. Details of both these methods will appear elsewhere in this CUVIC. The **IDISK** program and other similar programs (eg. **DISKZAP** etc.) are available from the CUVIC library and are very useful and fascinating programs to use.

Whilst we were dealing with compilers, we measured the speed advantage gained when a BASIC program is compiled. We chose Bruce Marshall's **DISKZAP2** as our example. The time taken for the uncompiled version to read a sector from the disk and print it on the screen was **24.8 sec**, whereas the compiled version took **6.9 sec**. This is approximately a 3.5 times improvement! Incidentally the latest version of **DISKZAP2** can give hardcopy printouts, which is a very useful option. This program is now available from the CUVIC library.

For those people who are interested in computer graphics, Barry Holt informed us that there is a very good article on 'fractals' in New Scientist 4 April 1985.

Alan Kirkpatrick

GHOSTBUSTERS!!

```
100 REM GHOST BUSTERS!
110 REM by Alan Kirkpatrick (26/05/85)
120 PLOT 12,6,2,15
130 PLOT 3,30,6,2,254,128,204,8,255
140 PLOT 3,28,7,2,254,128,200,255,6,16,32,32,32,6,2,2,
    254,19,255
150 PLOT 3,28,8,2,254,252,255,6,16,30,104,32,32,102,6,
    2,2,254,143,255
160 PLOT 3,28,9,6,16,32,2,254,112,255,106,2,254,112,
    255,106,32,6,1,2,254,204,204,136,8,255
```

Ghostbusters Program (continued)

```
170 PLOT 3,26,10,2,254,128,200,255,6,16,32,32,103,105,
    32,32,6,10,2,254,140,255,32,32,32,6,1,2,254,206,
    140,8,255
180 PLOT 3,24,11,2,254,128,232,255,6,17,2,254,63,1,255,
    32,32,6,16,2,254,192,12,255,32,32,32,6,8
190 PLOT 3,35,11,32,32,32,32,32,2,254,16,115,247,255
200 PLOT 3,23,12,2,254,63,3,255,32,6,17,2,254,143,255,
    32,32,32,6,16,2,254,48,3,255,32,32,32,6,8
210 PLOT 3,35,12,2,254,238,206,8,255,32,32,32,32,2,254,
    16,247,255
220 PLOT 3,22,13,2,254,31,1,255,32,32,32,6,17,2,254,8,
    255,32,32,6,16,118,119,32,32,2,254,16,17,17,49,255,
    6,8
230 PLOT 3,38,13,2,254,79,255,32,32,32,32,2,254,48,255,
    6,2,2,254,128,8,255
240 PLOT 3,21,14,6,8,2,254,63,255,32,32,32,32,6,17,2,
    254, 23,255,6,16,107,32,32,32,32,32,32,32,32,32
250 PLOT 3,37,14,6,17,2,254,200,254,255,6,10,32,32,32,
    32,2,254,200,254,255,6,2,2,254,3,255,6,8
260 PLOT 3,21,15,2,254,15,255,32,32,32,6,17,2,254,31,
    255,32,32,32,32,32,32,32,32,32,2,254,200,254,255
270 PLOT 3,37,15,6,10,32,32,2,254,128,252,140,232,127,
    3,255,6,1,2,254,12,255,6,10
280 PLOT 3,21,16,32,32,32,6,17,2,254,127,7,255,32,32,
    32,32,32,32,32,2,254,128,252,255
290 PLOT 3,35,16,6,8,32,32,6,17,2,254,127,19,255,6,16,
    97,32,32,32,6,17,2,254,16,17,255,32,6,2,2,254,110,
    2,255
300 PLOT 3,17,17,2,254,136,136,255,32,32,6,10,32,32,32,
    2,254,252,255,6,17,32,32,32,32,32,32,2,254,128,252,
    255
310 PLOT 3,33,17,6,10,32,32,2,254,128,236,255,6,16,32,
    126,32,105,32,32,32,6,17,2,254,96,255,6,8,2,254,
    16,255
320 PLOT3,17,18,6,2,2,254,16,51,247,206,255,6,17,2,254,
    51,119,3,255,32,32,32,32,32,2,254,128,236,255,6,10
330 PLOT 3,31,18,32,32,32,2,254,200,254,255,6,16,32,98,
    105,104,101,101,6,10,32,32,2,254,49,51,255
340 PLOT 3,17,19,6,16,2,254,159,8,1,255,32,32,32,32,32,
    2,254,128,8,255,6,10,2,254,127,19,255,32,32,32,2,
    254,128,254,255
350 PLOT 3,34,19,6,17,32,32,32,32,32,32,32,2,254,128,
    255,6,8,32,32,32,2,254,192,255
360 PLOT 3,18,20,6,16,2,254,55,19,1,255,32,32,103,6,10,
    2,254,19,255,6,8,2,254,16,19,255,32,32,32,6,17
370 PLOT 3,30,20,2,254,127,19,255,32,32,32,32,32,32,32,
    32,32,2,254,252,255,6,8,32,32,32,2,254,252,255
380 PLOT 3,18,21,6,2,2,254,16,17,17,17,255,6,10,2,254,
    49,243,14,255,32,32,32,6,8,2,254,128,236,255
390 PLOT 3,30,21,6,17,32,32,32,32,32,32,32,32,32,32,2,
    254,252,255,6,8,32,32,32,2,254,192,255
400 PLOT 3,22,22,2,254,207,12,255,6,10,2,254,1,255,32,
    32,6,8,2,254,32,255,6,16,2,254,63,1,255
410 PLOT 3,30,22,101,29,101,102,32,32,32,32,32,6,10,2,
    254,23,1,255,32,32,32,6,8,2,254,192,254,255
420 PLOT 3,23,23,2,254,239,14,255,32,32,32,32,6,17,2,
    254,142,136,255,32,32,32,2,254,128,200,236,254,255
430 PLOT 3,38,23,6,8,32,32,32,32,2,254,232,254,255
440 PLOT 3,25,24,2,254,207,8,255,32,32,32,32,32,32,32,
    32,32,32,32,32,32,2,254,200,254,255
450 PLOT 3,27,25,2,254,238,12,255,32,32,32,32,32,32,32,
    32,2,254,128,204,254,255
460 PLOT 3,30,26,6,1,2,254,16,17,17,17,17,17,17,255
470 PLOT 14,6,5,3,27,28:PRINT "GHOSTBUSTERS!!"
480 GOTO 480
```

CORRESPONDENCE.

FROM: Doug Van PUTTE, Rochester, N.Y. U.S.A. 16 June 1985

Thanks for including the note with the Newsletter to update me on the status of the ASMTUT disks. I had wondered why there was such a long delay.

On the question of CCII software released for Public Domain - I have contacted only one individual, Jim HELMS, to ask for copies of his programs and his permission to add them to the CHIP library. I received his permission very quickly, but since he had sold his CCII and put most of what he had on his Epson format, getting copies was not so easy.

We did, however, purchase a copy of his LEDGER program and his Assembly Language Data Base program from ICS. If you will provide two disks and postage (probably about $10 for Small Parcel Air) I would be glad to send copies of programs and manuals.

Jim also sent us an 8" Epson formatted disk with more of his software. Until we have it down-loaded to the CCII (hopefully by Ben BARLOW) and assembled, we do not know what it contains. I would assume, however, that any software which you collectively have purchased from Jim HELMS can now be distributed to Club members.

When ICS leaves the US, Joe NORRIS will be handling the distribution of that software collection. Since much of it is unsupported, I do not know what that means. For instance, COMTRONICS gave up some time ago.

One more item: If any of your members is interested in a copy of either J.CHARLES' book or David SUIT's book, I have some copies at $5 each plus postage. I could send them either Small Parcel Airmail or by surface mail.

Best regards, **Doug Van PUTTE**

FROM: "The Desk of William PARKER"
2812 Berkley, Flint, MI 48504 U.S.A. 15 June 1985

When I first took the Forum Library, it contained 18 two-sided disks. It still does. Not one later contribution was received, and it has been almost a year since I was sent any request for copies of disks.

Our catalog listing is on disk as are the North California and CUVIC catalogs. I suspect you already have copies, but if not, let me know, and I will send one at once. (I also have a more-or-less complete set of the North Calif. disks - two requests for corrected copies of several have gone unanswered. And I have CUVIC disks Nos.1 to 25)

Thanks, **Bill PARKER**

FROM: Christopher J. ZERR
10932 - 156th Court N.E. Redmond, WA 98052 U.S.A.
10 April 1985

Joseph NORRIS of COLORCUE has referenced me to you that perhaps I may be able to obtain a copy of a program called TRENDSPOTTER from your library if you have it. Could you give me more information on this and any other ISC software you might have? I would also like more information

on your user group and how to obtain software from your library.

Sincerely **C.J.ZERR**

FROM: Peter HINER, 11 Penny Croft, Harpenden, Harts, U.K.

6 June 1985

Thank you for your letter of 10 January - can it really be that long ago? Meanwhile I have been receiving your interesting newsletter regularly - thanks very much and congratulations on such a strong, healthy publication.

The main reason for this letter is to accompany a bug warning note **[See directly below, Ed]**. Would you please give it some publicity among interested parties, or even print it in your newsletter.

I am afraid I don't have any more programs from the UK to contribute at present, but will let you know if I get hold of anything.

Best wishes **Peter HINER**

BUG IN FASBAS (VERSION 12.24) **Peter HINER**

With a suitably red face, I regret to advise those of you who have got the latest version (VER 12.24) of my FASBAS compiler, that I have allowed a fatal bug to creep in. This bug was not present in earlier versions, so the corrective action outlined below should be applied to VER 12.24 only.

The bug can be eliminated by loading FASBAS to memory, keying in a BASIC POKE statement in immediate mode and then saving FASBAS on disk again, as follows:

ESC M (to make sure of a clean start)
ESC D (to enter FCS)
LOAD FASBAS.PRG (in response to **FCS>**)
ESC E (to return to BASIC)
POKE 39885, 4 (to eliminate the bug)
POKE 39234, 53 (optional entry to change display header from VER 12.24 to VER 12.25)
ESC D (back to FCS)
SAVE FASBAS.PRG;25 82A0 173F

Please check each entry carefully before hitting return, and at the end check the disk Directory to make sure that you have saved FASBAS.PRG;25 with the same values for size and load address as for version 24.

Columns	SIZE	LBC	LADR	SADR
should read	002F	3F	82A0	82A0

Credit for discovery of this bug goes to Doug van PUTTE, who presented me with a Basic program that appeared to compile satisfactorily but caused the assembler (FBASM) to crash. I discovered the reason for this to be an error in the size of output buffer allocated within FASBAS. Running on a 32K machine, FASBAS would overflow the top of the RAM if a medium to large size of Basic program was being compiled. On a 16K machine not even a small Basic program could be compiled!

I am sorry for any inconvenience caused. This debugged release of FASBAS will now be officially called VER 12.25.

oOo

ASSEMBLY LANGUAGE TUTORIAL SERIES **Bernie Maldouney**

TUTORIAL #10 - SAMPLE PROGRAMS

10.1 THE PROGRAMS

In this final tutorial we will consider five (5) complete programs rather than short routines. The five we will consider are:

1. LIFE - a patch to BASIC to assess a screen of information, modify it and redraw it.

2. BASTED - or BASIC TEXT EDITOR, converts a BASIC program which contains an Assembler Source File in REM Statements into the appropriate .SRC file ready for assembly.

3. QSORT - a dedicated implementation of the QUICKSORT algorithm used in CATALOG to alphabetise a random file by swapping records into alphabetical order.

4. SORTA - an implementation of the SHELL-METZNER sort algorithm which sorts a buffer, already filled by another program, into alphabetical order.

5. MONTOR - a machine language system monitor for the CCII.

10.2 SUGGESTED METHOD OF STUDY

The Source Listings, fully commented, for each program are provided in Appendices 6 through 10. **[Appendices 6 & 7 are contained in this issue, Appendices 8 - 10 will appear in subsequent issues of CUVIC. Ed.]** They have deliberately not been provided on disk to give practice in using a Text Editor and the first phase of debugging, checking the Source.

It is recommended that you first study the Source Listing, and Flow Chart where applicable, to see how the program works and how various instructions have been used to achieve a particular result.

You should then type them in using a Text Editor. Check the resulting Source File for accuracy, Assemble them and, where applicable, RUN them.

Once they are up and running, it is time to look at that "you-beaut" modification that you thought of to save a few bytes or speed execution.

While doing this, you will probably need to revise much of the earlier tutorials and find it all becoming progressively more meaningful. Copious notes of what you discover are in order during this process, particularly in different ways that you discover of looking at various parts of the Assembler. This series is one man's vision of what it is all about.

As you type in the Source code, you may wish to alter the comments to something more meaningful to you. It is good if you do, but avoid over- or under- commenting; do not put opposite LXI H,028D9H, the comment 'Load HL' for example. On the other hand, become worried if ten instructions have passed without comment.

The end object is not these programs themselves, although some may be useful to you, but rather the writing of your own Assembler programs, perhaps using some of the techniques described in the other Tutorials.

10.3 LIFE.SRC

This program was first written in Assembler by this writer and it really shows. However, it works, and no attempt has been made to pretty it up.

The NEED to write it arose from entering a program in BASIC to do the task and finding it needed up to five

minutes to calculate one generation. This was an obvious task for machine language coding.

The aim of the routine is to assess an array of cell spaces which, if they contain life are shown as an asterisk (*), otherwise as space characters; for survival, death or birth in the next generation according to simple but fixed rules, viz.:

1. In one generation, Birth and Death are simultaneous.

2. Cells are arranged in a two dimensional array giving eight(8) possible adjacent cells for each cell considered.

3. If an empty cell space has exactly three living adjacent cells, a new cell is born in that space.

4. If a living cell has either Two or Three living adjacent cells, that cell survives to the next generation, else it dies of either loneliness or overcrowding.

The initial array of living cells is set up in BASIC, with easily implemented I/O and random number function; as input by the operator or randomly selected according to given size and density parameters. This routine is then called from BASIC to apply the above rules, configure a new generation and download it to the screen.

Because of Rule #1, the display could not be updated as each cell space was assessed; a buffer in RAM was required to store the new display until the last space was assessed. A further refinement was added in showing newly-born cells in Cyan rather than Green.

A loop was required to assess each cell space from the first on-line 1 of the display through to the last line 30, ignoring those on lines 0 and 31, but assessing them as potential neighbours. (No wrap-around vertically although horizontal wrap-around occurs treating the array as the surface of a cylinder without ends).

Within that loop, each cell was tested for the presence of a living adjacent cell in each of the eight possible locations (present =1), and a score of 9 if the cell space itself contained life. Subroutines COUNT and COUNTS do the counting. After checking the last neighbour, an assessment is made on the basis of rules #3 & #4, and the result stored in the RAM buffer. At the same time, a population count is carried out and stored in NUMH & NUML.

On exiting the loop, the RAM buffer is downloaded to the screen by BLOCKM, and the population count is passed back to BASIC in DE on RETurn.

On reviewing the program now, I see that I was using a very limited subset of the 8080 Instruction Set which made 16-bit counting cumbersome. I can see that I felt compelled to comment every line, which was not a bad fault at that stage. Better use of the registers could have been made, leading to register compares rather than immediate ones which would have saved a few bytes but?

The BASIC program which drives it, and loads the patch, is appended in the Source Listing in Appendix #6. Note the delay factor to slow it down for viewing!

The flow chart is also appended but it is of little consequence as the program flow is essentially linear.

10.4 BASTED.SRC

This program was the second one written by me, but it is not terribly original in that it used SAMPLE.SRC from the Assembler disk as its framework.

This program grew out of my increasing frustration with EDT.PRG; I learnt to program with BASIC using its easy editing and found it hard to use a line editor where my cursor was effectively invisible. I have since got used to EDT with more practice and find it fairly easy to use now although I prefer EDITOR. So there was the NEED - how to provide the answer? On looking through SAMPLE, as you may now be looking at BASTED, I found that all it did was to copy any file byte by byte and Re-Type the Output File as .DOC. This, in itself, was useless but it was provided to show how sequential file access was done on the CCII. That knowledge, together with the memory of easy editing in BASIC, led me to think that .SRC files could be input and edited in BASIC and the resulting .BAS file stripped of line numbers and REMS and converted to .SRC files ready for Assembly. (REMS are necessary to prevent BASIC trying to make BASIC Keywords out of your text and converting part of your text to Tokens). I therefore set about altering SAMPLE as follows:

1. Convert default types to .BAS for input; .SRC for output.

2. At the point in the loop where bytes are got from the input file, insert a routine that strips off line numbers and 'next line pointers', checks for a REM token, and sends all the following bytes to the output file until 0, the end of the line marker, is found and replace it with CR/LF.

3. Insert an error routine for 'non-REM lines'.

4. Check for end of .BAS by checking 'next line pointer'.

5. Strip a little of the surplus bumpf from SAMPLE which was there for edification rather than the immediate purpose.

In this way, after a few tries, I had what I needed, using another's program as a crutch. In the process I learnt what the routines CHRINT & CI do, eben if dimly at that stage. I also learnt the inherent beauty in the CCII's sequential file access routines which carry out all their own file management if you point them in the right direction. As a further bonus, I was forced to learn more about .BAS file structure which has helped in other directions since.

This program is useful primarily in the same way as SAMPLE is - to learn sequential file access. Secondly, this program demonstrates how a well-documented original program can be the source of others. The further payoff is that this program can provide you with a text editor of sorts for entering your first efforts before you lash out on a Text Editor, providing you can borrow one to enter this program!

10.5 QSORT.SRC

This program was written while I was preparing the CATALOG Disk Management Package. I was faced with the problem of up to 1151 records, 11 bytes long, which had to be sorted into alphabetic order for output with all 11 bytes significant.

The best sort I could find in BASIC took nearly 40 minutes to sort a full file. That clearly was not good enough. I had decided to sort the file by re-writing it in order that subsequent accesses would always be in order. I

had also settled on the QUICKSORT algorithm and had it running in BASIC. The obvious step was to convert the BASIC program to Assembler.

The main part of the program follows the flowchart and is reasonably straightforward with the exception of the sub-file stack structure which took a while to solve. I learnt a lot more about stack structures in the process, but still feel I could have used some of the 8080 Stack instructions to manage it better had I been confident enough. By this time, my subset of the instruction set was increased considerably but was not yet a full set. Perhaps too, a use of ROM routine MOVDH would have tidied the record swaps.

The next hurdle was to access a file blockwise rather than byte-wise - a different set of file access routines needed to be mastered. This project, to be successful, needed the whole of the file to be read into memory at once, and written out again after sorting. Once I had found which parameters the routines needed placed in the FPB, I was O.K. but I had to guess a bit here not having the system listing at that stage.

The next problem was to tie it in with a BASIC Menu structure that ran the rest of the package. BASIC had first fill the Random files by stripping File Names, Types and Versions off Directories (11 bytes) and then load a free byte in System RAM to tell QSORT how many files had to be sorted. It then had to RUN QSORT and QSORT had to return to MENU. These requirements resulted in:

1. QSORT must load >=829A. 829A was chosen to give maximum RAM for sorting.

2. QSORT was made to return to MENU by loading 1(PLOT1) into A and JMPing to 0033 (BASOUT).

3. The need to keep within 16K for general distribution and the 829A Load Address dictated the maximum file size allowable (1152) which gave optimum blocking for Random files (no waste bytes) and this happily suited the expected number of records from the package (with two files, but if 32K was the norm, only one file would have been needed). [Incidentally, try getting optimum blocking with record size being a prime number like 11; it is very limiting.]

Note the use of 'variables' RN & RS within the listing. s is very convenient if you wish at any time to change either; only one alteration required. Note also the messages which indicate where the program is at. These were used for debugging but it was decided to leave them to make the program more 'user-friendly'. When this program reduced sort-time from 40 minutes to 27 seconds, I just had to be sure it had indeed finished and not 'fallen through'!

This program demonstrates the superior speed of Assembler in sort routines, an interface with BASIC other than Y = CALL(X), and the Block Read and Write file access routines. It is documented well enough to be adapted to similar dedicated tasks or better still to be generalized to sort on keyboard-provided parameters, any Random file over any key field. Who shall be first?

10.6 SORTA.SRC

This program was not written by this writer but it took my eye in the way that the stack was used to access commonly needed values, enabling them to be placed in any register pair using only 2 bytes. (POP & PUSH). This routine sorts on a buffer provided by another program (BAS or PRG) and the article which described it suggested a neat way of accessing the file alphabetically.

The sort algorithm used is 'SHELL-METZNER', which is a half interval type-sort, and it appears very efficient (similar to Quicksort). It is, however, a little easier to program without sub-files to worry about. The program listing follows from the flowchart readily enough but the use of registers is very neat - watch them closely. Note the change of one instruction which yields a sort in descending order. There are no file access routines used as this is really a sub-routine which could be called from BASIC or within a larger program or indeed from another .PRG program using ROM routine 'FCS'.

Whereas QSORT (with each byte of the record significant) does its work by reloading the file with the records sorted into order, the author of this program suggested a use whereby you extract the sort field from each record on a file (say the Surname in 14 bytes), add two bytes to it to indicate the record number in Hex, and load that record to the buffer for sorting. After sorting, those extra two bytes on the sort field provide a key to access the total record in alphabetical order by Surname. This could be used immediately or saved to another file with an equal number of records, but only two bytes each. Whenever you needed to access the main file in that order, this key file would provide the order of access. Sounds pretty neat to me, and after I have finished this job and rested, perhaps I may look more closely at Personal Data Base whose speed is a pain in the Accumulator.

This example provides yet another sort implementation in Assembler which is the best approach to large sorts. It provides a model usage of registers and stack. It also provides the stimulus to build it to a stand-alone program of fairly general applicability by using I/O and File Access routines from earlier examples.

10.7 MONTOR.SRC

This program was provided by a user group in the U.S.A. I have not had a chance to check it extensively but a system monitor is particularly useful if you do not have MLDP although not as powerful. The very nature of a system monitor is such that its source must tell you a lot about the system.

I have not altered it in any way and offer it as-is. I did however add the addresses for 8.79 software as there is no overlay routine and 6.78 addresses are used. The Input Syntax used is cumbersome, and without a manual or similar reference, hopeless. The first modification I would make would be to enable Command 1 to print the Input Syntax to the screen.

Another lack I see is the absence of a re-entry vector as discussed in Tutorials #8 & #9. Re-entry via (ESC ^) would enhance this program. The Search Memory command is useful for finding byte sequences and this is lacking from MLDP. It is also not clear that the optional second and third addresses on GOTO are breakpoint addresses. You apparently clear these breakpoints by setting others but I am not certain yet. The routines used appear sound and informative but extensive modification is for experienced programmers only.

This program also utilises CHRINT & CI and it is

instructive to see how these are used to access an input table and process numeric input. As you come to grips with interrupts the programmable interrupts used as breakpoints will be informative.

One use for the fill memory command that I use with MLDP is to fill your programs RAM area with zeroes before loading an .LDA file for saving as a .PRG type. This ensures that data areas, which the .LDA file does not write specific bytes to, are not full of junk in the .PRG file, a feature of some software that I find annoying.

This program, as is, should provide some useful features after you have written your first program or two. As you become more experienced it should provide plenty of scope for modification.

ASSEMBLY LANGUAGE TUTORIAL SERIES

Appendix #5 Utility Routines

(Continued from page 8, June 85, CUVIC)

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
S1OUT:          ;ROM ROUTINE
                ;SENDS A BYTE TO THE SERIAL PORT WITH
                ;HANDSHAKE OF SORTS.
                ;BYTE MUST BE IN <E> AND YOU USUALLY
                ;NEED TO PRESERVE <A> BEFORE CALLING BY
                ;POP PSW
        IN      03H         ;READ STATUS ON TMS 5501
        ANI     10H         ;XMIT BUFFER EMPTY?
        JZ      S1OUT       ;NO
        IN      01H         ;CLEAR TO SEND OK?
        ANI     80H         ;
        JZ      S1OUT       ;NO
        MOV     A,E         ;SEND IT
        OUT     06H         ;OUT
        RET
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
SAVE:           ;ROM ROUTINE TO SAVE ALL REGISTERS
                ;CALLED ON ENTRY TO ANY SUBROUTINE
                ;SAVES ALL REGISTERS AND WHEN SUBROUTINE
                ;RET'S, RESTORES THEM ALL AUTOMATICALLY

        XTHL            ;SAVE HL WHILE LOADING ADDRESS
        PUSH    D       ;TO RESTORE EXECUTION AT
        PUSH    B       ;THEN SAVE ALL REGISTERS
        PUSH    PSW     ;AND STATUS
        CALL    JMPHL   ;PUSH $+3 AS RETURN ADDRESS
        POP     PSW     ;AND RESUME EXECUTION OF CALLER
        POP     B       ;THEN RESTORE ALL REGISTERS
        POP     D       ;AND STATUS
        POP     H
        EI              ;SAVE IS USUALLY USED IN
                        ;SERVICING
        RET             ;INTERRUPTS
JMPHL:  PCHL            ;JMP THRU HERE TO ROUTINE
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
BN2DEC:                 ;CONVERT 16 BINARY BITS TO BCD
                        ;5 DIGIT ASCII BINARY BITS ARE
                        ;IN DE ON ENTRY AND THE FIVE
                        ;BCD DIGITS ARE STORED IN THE
                        ;5 LOCATIONS COMMENCING AT
                        ;'DECDIG'
        LXI H,DECDIG    ;SET UP POINTER
        CALL    SAVE    ;ROUTINE USES ALL REGISTERS
        XCHG            ;BITS IN HL, DE IS POINTER
        LXI B,-10000    ;GET LEFTMOST DIGIT
        CALL DECNO
        LXI B,-1000     ;NEXT DIGIT
        CALL DECNO
        LXI B,-100      ;AGAIN
        CALL DECNO
        LXI B,-10       ;2nd FROM RIGHT
        CALL DECNO
        MOV     A,L     ;REMAINDER = RIGHTMOST DIGIT
        ADI     30H     ;TO ASCII
        STAX    D       ;& STORED
        RET
DECNO:  MVI     A,30H   ;ASCII 0
        PUSH    D       ;STACK DECDIG POINTER
DECNO1: MOV     E,L     ;SAVE REMAINDER TO DE
        MOV     D,H
        INR     A       ;ANTICIPATE DIGIT
        DAD     B       ;SUBTRACT DIVISOR
        JC      DECNO1;WAS ENOUGH FOR SUBTRACTION
        DCR     A       ;NOT ENOUGH LEFT SO
        MOV     L,E     ;CANCEL LAST DAD OPERATION
        MOV     H,D
        POP     D       ;RESTORE POINTER
        STAX    D       ;& STOW DIGIT
        INX     D       ;& BUMP POINTER
        RET
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

Appendix #6 Sample Program #1

'LIFE' IS A PATCH CALLED FROM BASIC TO CALCULATE NEW GENERATIONS FOR THE SIMULATION OF AN IDEALISED GENETIC GROWTH AND DECAY PATTERN WHICH IS VERY SLOW IN BASIC. SEE THE ACCOMPANYING BASIC PROGRAM FOR MORE DETAIL.

```
        ORG     9000H   ;SADR = 9000H
INIT:   LXI     H,NUMH  ;POINT TO MSB OF POPUL'N
        MVI     M,0     ;AND ZERO IT
        MVI     B,1     ;POP COUNTER TO ONE
        LXI     H,6000H ;START OF SCREEN MEMORY
SEARCH: MVI     A,2AH   ;SEARCH FOR '*'S
        MVI     D,0     ;ZERO D FOR LATER DAD OP
        MVI     C,0     ;ZERO C FOR COUNTING *'S
        CALL    COUNT   ;COUNT NW NEIGHBOUR
        INX     H       ;MOVE TO
        INX     H       ;N NEIGHBOUR
        CALL    COUNT   ;COUNT N NEIGHBOUR
        INX     H       ;MOVE TO
        INX     H       ;NE NEIGHBOUR
        CALL    COUNT   ;COUNT NE NEIGHBOUR
        MVI     E,80H   ;OFFSET FOR 1 SCRN LINE
        DAD     D       ;BUMP HL ONE SCRN LINE
        CALL    COUNT   ;COUNT E NEIGHBOUR
        DCX     H       ;MOVE TO
        DCX     H       ;CELL UNDER TEST
        CALL    COUNTS  ;COUNT TEST CELL
        DCX     H       ;MOVE TO
        DCX     H       ;W NEIGHBOUR
        CALL    COUNT   ;COUNT W NEIGHBOUR
        DAD     D       ;BUMP HL 1 SCRN LINE
        CALL    COUNT   ;COUNT SW NEIGHBOUR
        INX     H       ;MOVE TO
        INX     H       ;S NEIGHBOUR
```

```
        CALL    COUNT       ;COUNT S NEIGHBOUR
        INX     H           ;MOVE TO
        INX     H           ;SE NEIGHBOUR
        CALL    COUNT       ;COUNT SE NEIGHBOUR
ASSESS: MVI     A,3         ;BIRTH = 3 COUNT
        CMP     C           ;YES OR NO
        JNZ     LIVE2       ;NO BIRTH SO LOOK FOR LIVING
BIRTH:  MVI     D,2AH       ;LOAD * INTO D
        MVI     E,6         ;LOAD CYAN INTO E
        INR     B           ;ADD ONE TO POPUL'N
        JMP     STORE       ;STORE D AND E IN PSEUDO SCRN
LIVE2:  MVI     A,11        ;LOOK FOR CELL WITH 2 N
        CMP     C           ;YES OR NO
        JZ      SURVIV      ;PHEW - MADE IT
LIVE3:  MVI     A,12        ;LOOK FOR CELL WITH 3 N
        CMP     C           ;YES OR NO
        JZ      SURVIV      ;NO MORE CHANCES
DEATH:  MVI     D,20H       ;LOAD SPC INTO D
        MVI     E,2         ;LOAD GREEN INTO E
        JMP     STORE       ;AND STORE THEM
SURVIV: MVI     D,2AH       ;LOAD * INTO D
        MVI     E,2         ;LOAD GREEN INTO E
        INR     B           ;ANOTHER LIVING
STORE:  MVI     A,0         ;CHECK IF B FULL
        CMP     B           ;YES OR NO
        CZ      BOVER       ;BUMP MSB OF POPULATION
        MVI     A,40H       ;STORE = LAST TEST PLUS 4000H
        ADD     H           ;FOR CHARACTER
        MOV     H,A         ;H NOW BUMPED
        MOV     M,D         ;STORE CHARACTER
        INX     H           ;POINT TO COLOR STORE
        MOV     M,E         ;STORE COLOR
        DCX     H           ;BACK TO LAST TEST + 4000H
EOS:    MOV     A,H         ;CHECK H FOR LAST CHAR
        CPI     0AFH        ;AT AFFEH
        JNZ     BTSCRN      ;BACK TO WORK
        MOV     A,L         ;CHECK L FOR LAST CHAR
        CPI     0FEH        ;YES OR NO
        JZ      BLOCKM      ;LAST CHAR IS STORED
BTSCRN: MOV     A,H         ;BACK TO NEW SEARCH
        SUI     41H         ;BY DECR HL BY 4100H
        MOV     H,A         ;THEN
        DCX     H           ;DECR BY 2
        DCX     H           ;TOTAL DECR = 4102H
        JMP     SEARCH      ;BACK TO TESTS
BLOCKM: LXI     H,NUML      ;POINT TO LSB OF POPUL'N
        MOV     M,B         ;STORE LSB
        LXI     B,0EFCH     ;0EFCH BYTES TO MOVE
        LXI     H,0A104H    ;POINT TO LOWEST STORE ADDR
        LXI     D,6002H     ;POINT TO LOWEST SCREEN TEST
MOVEIT: MOV     A,M         ;BYTE TO A
        STAX    D           ;BYTE TO SCRN
        DCX     B           ;DECR COUNTER
        MOV     A,B         ;CHECK COUNTER
        CPI     0           ;AGAINST 0
        JNZ     STILL       ;GET ANOTHER
        MOV     A,C         ;CHECK COUNTER
        CPI     0           ;AGAINST 0
        JZ      BACK        ;BACK TO BASIC
STILL:  INX     H           ;NEXT BYTE
        INX     D           ;NEXT SADR
        JMP     MOVEIT      ;MOVE NEXT BYTE
BACK:   LXI     H,NUMH      ;POINT TO POPUL'N DATA
        MOV     D,M         ;GET MSB
        DCX     H           ;POINT TO LSB
        MOV     E,M         ;GET LSB
        DCX     D           ;ADJUST POP COUNT(STARTED AT
                            ;1)
        RET                 ;BACK TO BASIC CALL
COUNT:  CMP     M           ;* THERE?
        RNZ                 ;BACK IF NOT
        INR     C           ;BUMP COUNTER OF *'S
        RET                 ;BACK TO SEARCH
COUNTS: CMP     M           ;* THERE?
        RNZ                 ;NO COUNT FOR EMPTY TEST CELL
        MVI     A,9         ;9 COUNT FOR LIVE TEST CELL
        ADD     C           ;TO GET ABOVE POSSIBLE 8 COUNT
        MOV     C,A         ;FOR EMPTY CELL
        MVI     A,2AH       ;RESTORE * TO A FOR TESTS
        RET                 ;BACK TO SEARCH
BOVER:  PUSH    H           ;SAVE TEST LOC'N WHILE
        LXI     H,NUMH      ;POINT TO POPUL'N MSB
        INR     M           ;AND INCR BY ONE
        POP     H           ;POINT BACK TO TEST AREA
        RET
NUML:   DB      0           ;LSB OF POPUL'N COUNT TO 0
NUMH:   DB      0           ;MSB OF POPUL'N COUNT TO 0
        END
```

```
100 REM  ############### LIFE.BAS ###############
110 REM  26 JUN 80
120 REM  M/L DATA WITHIN PROGRAM IN THIS VERSION CORRECTED
         FOR POP COUNT ERROR
130 PLOT 27,24,12,15
140 CLEAR 100
150 GOSUB 500
160 INPUT "DELAY FACTOR  (0-10) ? ";Q:PRINT
170 INPUT "RANDOM OR DESIGNED START PATTERN ? ";P$
180 IF LEFT$ (P$,1)= "D"THEN 340
190 REM  RANDOM STARTING PATTERN
200 PRINT :INPUT "PATTERN SIZE  (0-8) ? ";S
210 INPUT "DENSITY OF PATTERN (1-5) ? ";V:V= 7- V
220 PLOT 6,2,12
230 PP= 0
240 FOR RR= 1TO 5* PEEK (33209):R= RND (1):NEXT
250 FOR Y= 12- STO 18+ S:FOR X= 24- 2* STO 40+ 2* S
260 RR= INT (V* RND (1))+ 1:IF RR< > VTHEN 280
270 PLOT 3,X,Y:PRINT "*":PP= PP+ 1
280 NEXT X,Y
290 Y= PP
300 PLOT 8:PRINT "GENERATION # 1"TAB( 32)"POPULATION "PP
310 PLOT 3,16,31:PRINT "++++++++ LIFE ++++++++"
320 PLOT 3,0,1
330 G= 1:GOTO 440
340 REM  DESIGNED PATTERN
350 PLOT 6,2,12,8:PP= 0
360 PRINT "ONLY ENTER '*'S AND PRESS RETURN AFTER LAST
    ENTRY ON LINE."
370 FOR LL= 8TO 24
380 PLOT 3,0,LL:INPUT M$
390 NEXT LL
400 PLOT 8,11:PRINT "GENERATION # 1":G= 1
410 PLOT 3,16,31:PRINT "++++++++ LIFE ++++++++"
```

```
420 PLOT 3,0,1:GOTO 440
430 IN= PEEK (33278):IF IN< > 0THEN 480
440 POKE 33278,0
450 Y= CALL (0):G= G+ 1
460  PLOT  8,11:PRINT  "GENERATION  #"GTAB(   32)"POPULATION
"Y:PRINT
470 FOR TT= 1TO 250# Q:NEXT :GOTO 430
480  PLOT 3,0,1:PRINT "AGAIN  ";:INPUT YN$:IF LEFT$ (YN$,1)(
   > "Y"THEN PLOT 27,11:END
490 PLOT 12:GOTO 160
500 POKE 33283,0:POKE 33284,144
510 FOR AD= 36864TO 37060:READ Z:POKE AD,Z:NEXT
520 RETURN
600 DATA 229,33,196,144,54,0,6,1,33,0,96,62,42,22,0,14
610 DATA 0,205,175,144,35,35,205,175,144,35,35,205,175,144,
   30,128
620 DATA 25,205,175,144,43,43,205,179,144,43,43,205,175,
   144,25,205
630 DATA 175,144,35,35,205,175,144,35,35,205,175,144,62,3,
   185,194
640 DATA 74,144,22,42,30,6,4,195,98,144,62,11,185,202,
   93,144
650 DATA 62,12,185,202,93,144,22,32,30,2,195,98,144,22,
   42,30
660 DATA 2,4,62,0,184,204,188,144,62,64,132,103,114,35,
   115,43
670 DATA 124,254,175,194,124,144,125,254,254,202,133,144,
   124,214,65,103
680 DATA 43,43,195,11,144,33,195,144,112,1,252,14,33,
   4,161,17
690 DATA 130,96,126,18,11,120,254,0,194,161,144,121,
   254,0,202,166
700 DATA 144,35,19,195,146,144,33,196,144,86,43,94,
   27,225,201,190
710 DATA 192,12,201,190,192,62,9,129,79,62,42,201,
   229,33,196,144
720 DATA 52,225,201,0,0
```

Appendix #7 Sample Program #2

BASIC TEXT EDITOR 'BASTED'

PROGRAM TO CONVERT REM ONLY BAS FILE TO SRC FILE DERIVED FROM SAMPLE.SRC ON ASSEMBLER DISC.

BASTED ACCESSES A BASIC FILE NOMINATED FROM THE KEYBOARD, SEARCHES EACH LINE NUMBER SEQUENTIALLY FOR A REM STATEMENT, REJECTS THE FILE IF REM IS NOT THE FIRST STATEMENT ON ANY LINE, ELSE PROCEEDS TO TRANSFER THE REST OF THE LINE TO AN OUTPUT SRC FILE, REPLACING BASIC'S LINE TERMINATOR (0) WITH <CR><LF>.

ON REACHING THE END OF BASIC'S SOURCE, <00> ADDR OF NEXT LINE, THE OUTPUT FILE IS CLOSED AND WRITTEN OUT AND CONTROL RETURNS TO FCS.

```
;       SYSTEM ADDRESSES
        ORG     829AH       ;KEEP  COMPATIBLE  WITH  BASIC
MENU
START1: LDA     TEST        ;PROGRAM START ADDRESS
        CPI     NEW         ;SEE IF 8/79 OR 6/78 SOFTWARE
        JNZ     START       ;IF OLD DO NOT OVERLAY VECTOR
                            ;TABLE
        LXI     H,OLDVEC
        LXI     D,NEWVEC
        LXI     B,LENTH
OVERLAY:LDAX    D           ;OVERLAY JUMP TABLE
        MOV     M,A
        INX     H
        INX     D
        DCX     B
        MOV     A,B
        ORA     C
        JNZ     OVERLAY
        JMP     START       ;ENTER PROGRAM PROPER
;       VECTOR TABLE
OLDVEC  EQU     $
CO:     JMP     3392H       ;SEND CHARACTER TO SCREEN
OS:     JMP     33F4H       ;SEND STRING ENDING WITH 239
EMESS:  JMP     262DH       ;FCS ERROR SUBROUTINE
PFSPC:  JMP     3077H       ;PARSE FILE SPEC
RESET:  JMP     26A5H       ;RESET DISK
OPEN:   JMP     2DABH       ;OPENS A FILE
RWSEQI: JMP     30C6H       ;'REWINDS' A SEQUENTIAL FILE
INSEQO: JMP     30E7H       ;INITIALIZES A SEQUENTIAL
                            ;OUTPUT FILE
CLSEQO: JMP     3136H       ;CLOSES A NEWLY CREATED
                            ;SEQUENTIAL FILE
GTBYT:  JMP     322CH       ;GETS A SINGLE BYTE FROM FILE
PTBYT:  JMP     324AH       ;PUTS A BYTE ON A FILE
ADHLA:  JMP     3518H       ;ADDS A TO HL WITH CARRYS
LENTH   EQU     $-OLDVEC
;       RAM LOC
NEW     EQU     1FH         ;VALUE OF BYTE 8/79 SOFTWARE
TEST    EQU     092AH       ;LOCATION OF A BYTE THAT IS
                            ;DIFFERENT
INPCRT  EQU     81C5H       ;JUMP VECTOR NUMBER 31
KBDFL   EQU     81DFH       ;HOLDS NUMBER OF JUMP VECTOR
;       8/79 SOFTWARE VECTOR TABLE
NEWVEC  EQU     $
        JMP     17C8H
        JMP     182AH
        JMP     0AD6H
        JMP     14ADH
        JMP     0B48H
        JMP     11E1H
        JMP     14FCH
        JMP     151DH
        JMP     156CH
        JMP     1662H
        JMP     1680H
        JMP     194EH
;       FCB REFERENCES
FTYP    EQU     8
FLAD    EQU     17
FBUF    EQU     32
FXBC    EQU     34
;       SPECIAL ASCII CONTROL CHARACTERS
BS      EQU     26          ;BACK SPACE
ESC     EQU     27          ;ESC KEY
SPC     EQU     32          ;SPACE CHR
REM     EQU     142         ;REM TOKEN
CR      EQU     13          ;CR RETURN
LF      EQU     10          ;LINE FEED
;       START OF PROGRAM PROPER
START:  LXI     H,0         ; SAVE FCS STACK POINTER
        DAD     SP
        SHLD    FCSSP
```

```
        LXI     SP,STACK    ;SETUP RUNOFF STACK
        MVI     A,31        ;SET UP VECTOR
        STA     KBDFL       ;TO INPCRT
        MVI     A,ØC3H      ;LITERALLY JMP
        STA     INPCRT
        LXI     H,CHRINT    ;THUS
                            ;INPCRT:JMPCHRINT
        SHLD    INPCRT+1
        LXI     H,INBUF     ;POINT TO COUNTER
        MVI     M,Ø         ;CLEAR COUNTER
        LXI     H,MSGØØ     ; PRINT STARTUP MESSAGE
        CALL    OS
        CALL    SETUP       ; INITIALIZE FILES
LOOP:   LXI     H,FPB1
        CALL    GTBYT       ;FIRST LINE BYTE = LSB ADDR
                            ;NEXT LINE
        JC      LOOPE1
        LXI     H,FPB1
        CALL    GTBYT       ;SECOND LINE BYTE = MSB ADDR
                            ;NEXT LINE
        JC      LOOPE1
        CPI     Ø           ;ZERO HERE INDICATES END OF
                            ;BAS FILE
        JZ      EOF
        LXI     H,FPB1
        CALL    GTBYT       ;THIRD LINE BYTE = LSB OF LINE
                            ;NUMBER
        JC      LOOPE1
        LXI     H,FPB1
        CALL    GTBYT       ;FOURTH LINE BYTE = MSB OF
                            ;LINE NUMBER
        JC      LOOPE1
        LXI     H,FPB1
        CALL    GTBYT       ;FIFTH LINE BYTE = START OF
                            ;LINE LISTING
        JC      LOOPE1
        CPI     REM         ;AND SHOULD BE REM
        JNZ     SERROR      ;IF NOT REM THEN BAILOUT
TEXT:   LXI     H,FPB1
        CALL    GTBYT       ;FIRST TEXT BYTE BUT COULD BE
                            ;A SPC
        JC      LOOPE1
        CPI     SPC         ;IF TOKEN INCLUDES SPC THEN
                            ;WIPE IT
        JNZ     TEXT3
TEXT2:  LXI     H,FPB1
        CALL    GTBYT       ;TEXT BYTE FOR TRANSFER
        JC      LOOPE1
        CPI     Ø           ;ZERO INDICATES END OF LINE
        JZ      EOL         ;INSERT SRC TERMINATORS IN
                            ;PLACE OF BAS Ø
TEXT3:  LXI     H,FPB2
        CALL    PTBYT       ;STORE TEXT CHR
        JNC     TEXT2
LOOPE2: JMP     ERROR
LOOPE1: JZ      EOF         ; AT END OF FILE - CLOSE
                            ;OUTPUT
        JMP     ERROR
EOL:    MVI     A,CR        ;INSERT CR/LF AT END OF LINE
        LXI     H,FPB2
        CALL    PTBYT
        JC      LOOPE1
        MVI     A,LF
        LXI     H,FPB2
        CALL    PTBYT
        JC      LOOPE1
        JMP     LOOP        ;START ON ANOTHER LINE
SETUP:  LXI     H,MSGØ1     ; DISPLAY PROMPT
        CALL    OS
        CALL    RESET       ; RESET DISK IF ERROR
        LXI     H,BUFFER    ; POINT AT BUFFER
LØ2:    CALL    CI          ; READ FROM CONSOLE
        CPI     CR          ; IS IT CR ?
        JZ      XØ2         ; YES, GO PROCESS CR
        CPI     BS          ; IS IT BS ?
        JZ      GØ21        ; YES, GO PROCESS BACKSPACE
        MOV     M,A         ; STORE CHARACTER
        INX     H           ; BUMP POINTER
        JMP     LØ2         ; CONTINUE LOOP
GØ21:   MOV     A,L         ; TEST LO BYTE OF POINTER
        ORA     A           ; IF ZERO THEN RESTART READ
        JZ      SETUP
        MVI     A,' '       ; ELSE SPACE OVER
        CALL    CO
        MVI     A,BS        ; TYPED DATA
        CALL    CO
        DCX     H
        JMP     LØ2         ; CONTINUE LOOP
XØ2:    MVI     M,Ø         ; INSERT TERMINATOR
        MVI     A,CR        ; PRINT CARRIAGE RETURN
        CALL    CO
        MVI     A,LF        ; LINEFEED SEQUENCE
        CALL    CO
        LXI     H,BUFFER    ; POINT AT BUFFER
        LXI     D,FPB1      ; POINT AT INPUT FPB
        LXI     B,DEFAULT   ; POINT AT DEFAULT TYPE
        CALL    PFSPC       ; PARSE FILE SPEC
        JC      EØ2         ; IF CARRY THEN ERROR
        MOV     A,M         ; TEST FOR END
        MVI     B,9         ; SETUP SYNTAX ERROR
        ORA     A
        JNZ     EØ2
        LXI     H,FPB1      ; POINT AT INPUT FPB
        MVI     A,Ø         ; SETUP AS OLD FILE
        MOV     M,A
        CALL    OPEN        ; OPEN THE FILE
        JC      EØ2
        LXI     H,FPB1      ; COPY FPB1 TO FPB2
        LXI     D,FPB2
        MVI     B,38
LØ3:    MOV     A,M
        STAX    D
        INX     H
        INX     D
        DCR     B
        JNZ     LØ3
        LXI     H,FPB2+FTYP; POINT AT OUTPUT TYPE
        MVI     M,'S'       ; STORE 'S'
        INX     H
        MVI     M,'R'       ; STORE 'R'
        INX     H
        MVI     M,'C'       ; STORE 'C'
```

```
        LXI     H,FPB2+FLAD;POINT AT LOAD ADDRESS FOR
                           ;OUTPUT
        MVI     M,Ø        ;AND LOAD ØØØØ
        INX     H
        MVI     M,Ø
        INX     H          ;POINT AT START ADDRESS FOR
                           ;OUTPUT
        MVI     M,Ø        ;AND LOAD ØØØØ
        INX     H
        MVI     M,Ø
        LXI     H,FPB2     ; POINT AT OUTPUT FILE
        MVI     A,1        ; SETUP AS NEW FILE
        MOV     M,A
        CALL    OPEN       ; OPEN THE FILE
        JC      EØ2
        LXI     H,IBUF     ; POINT AT INPUT BUFFER
        SHLD    FPB1+FBUF  ; SAVE INPUT BUFFER ADDRESS
        LXI     H,OBUF     ; POINT AT OUTPUT BUFFER
        SHLD    FPB2+FBUF  ; SAVE OUTPUT BUFFER ADDRESS
        LXI     H,1Ø24     ; SETUP BUFFER SIZE
        SHLD    FPB1+FXBC  ; SAVE INPUT BUFFER SIZE
        SHLD    FPB2+FXBC  ; SAVE OUTPUT BUFFER SIZE
        LXI     H,FPB1     ; POINT AT INPUT FPB
        CALL    RWSEQI     ; REWIND INPUT FILE
        LXI     H,FPB2     ; POINT AT OUTPUT FPB
        CALL    INSEQO     ; INITIALIZE OUTPUT FILE
        RET                ; RETURN
CHRINT: PUSH    H          ;SAVE REGISTERS
        PUSH    PSW
        LXI     H,INBUF
        MVI     A,3Ø       ;LENGTH OF BUFFER
        CMP     M          ;SEE IF BUFFER IS FULL
        JC      INTEXT     ;IF IT IS IGNORE CHARACTER
        INR     M          ;INCREMENT COUNTER
        MOV     A,M        ;PUT COUNT IN A
        CALL    ADHLA      ;ADD COUNTER TO BEGINNING OF
                           ;BUFFER
        MOV     A,E        ;CHARACTER IS IN E
        CPI     ESC        ;SEE IF ESCAPE HAS BEEN HIT
        JZ      BREAK      ;USER WANTS TO EXIT
        ANI     127        ;MAKE SURE ITS ASCII
        MOV     M,A        ;STORE IT
INTEXT: POP     PSW
        POP     H          ;RESTORE REGISTERS
        EI                 ;REENABLE INTERUPT
        RET
BREAK:  LXI     H,4        ;POINT TO RETURN ADDRESS
        DAD     SP         ;NOW H POINTS TO RETURN
                           ;ADDRESS
        MVI     M,BAILOUT AND 255;STORE LOW ORDER BYTE
        INX     H
        MVI     M,BAILOUT/256;STORE HIGH ORDER BYTE
        JMP     INTEXT     ;EXIT INTO EXIT
INBUF:  DS      32         ;STORAGE  AREA FOR TYPE AHEAD
CI:                        ;CI READS AND ECHOS ONE
                           ;CHARACTER
                           ;FROM THE KEYBOARD
        PUSH    H          ;SAVE H
        LXI     H,INBUF
CI1:    MOV     A,M        ;READ COUNTER
        ANA     A          ;SEE IF ZERO
        JZ      CI1        ;LOOK AND WAIT IF ZERO
        DI                 ;HOLD OFF INTERUPTS WHILE
                           ;PROCESSING
        PUSH    D          ;SAVE D
        DCR     M          ;REDUCE BUFFER COUNTER
        MOV     D,M
        INX     H          ;POINT TO CHARACTER
        MOV     A,M        ;READ IT
        CALL    CO         ;ECHO IT
        PUSH    PSW        ;STACK IT
CI2:    DCR     D          ;SHIFT COUNTER
        JM      CI3        ;WHEN DONE
        INX     H          ;CHARCTER TO BE SHIFTED
        MOV     A,M
        DCX     H          ;POINT TO NEW HOME
        MOV     M,A        ;NEW HOME
        INX     H
        JMP     CI2        ;LOOP UNTIL DONE
CI3:    EI                 ;ENABLE INTERUPTS
        POP     PSW        ;THE CHARACTER
        POP     D          ;THE WORLD
        POP     H
        RET
EØ2:    CALL    EMESS      ; EMIT ERROR MESSAGE
        JMP     SETUP      ; READ NEW LINE
SERROR: LXI     H,MSGØ3    ; PRINT INPUT FILE ERROR
        CALL    OS
        JMP     BAILOUT    ;BACK TO FCS
ERROR:  LXI     H,MSGØ2    ; PRINT ERROR MESSAGE
        CALL    OS
BAILOUT: MVI    B,Ø        ; SETUP NO ERROR
EXIT:   LHLD    FCSSP      ; RESTORE FCS STACK
        SPHL
        RET                ; AND EXIT TO FCS
EOF:    LXI     H,FPB2     ; POINT AT OUTPUT FILE
        CALL    CLSEQO     ; CLOSE OUTPUT FILE
        JMP     EXIT
;       MESSAGES
MSGØØ:  DB      18,11,'COMPUCOLOR ',19,'II '
        DB      21,'BASTED ',22,'V9.8Ø',13,1Ø
        DB      22,'ENTER .BAS FILE NAME FOR DELETION OF'
        DB      'REMS AND LINE NUMBERS',13,1Ø
        DB      'AND OUTPUT AS .SRC FILE READY FOR'
        DB      ' ASSEMBLY.',13,1Ø,239
MSGØ1:  DB      19,11,'CONVERT>',18,239
MSGØ2:  DB      17,'HARDWARE ERROR !',13,1Ø,239
MSGØ3:  DB      17,'BASIC FILE HAS OTHER THAN '
        DB      'REM LINES!',13,1Ø,239
DEFAULT:DB      'BAS'
;       DATA AREAS ...  STARTED ON A NEW PAGE
        ORG     ((S/256)+1)*256
BUFFER: DS      256
IBUF:   DS      1024       ;INPUT BUFFER
OBUF:   DS      1024       ;OUTPUT BUFFER
        DS      1ØØ        ;STACK AREA
STACK:
FCSSP:  DS      2          ;FCS STACK POINTER
FPB1:   DS      38         ;INPUT FPB
FPB2:   DS      38         ;OUTPUT FPB
        END     START1
```

NCC # 10 CONTINUED-

6. MOVE FILE - MOVES ANY TYPE OF FILE FROM ONE DISK TO ANOTHER USING DOS READ AND WRITE COMMANDS.
7. NEW DUMP - DISASSEMBLES, WITH OUTPUT TO PRINTER.
8. DIR - OUTPUTS DISK DIRECTORY TO PRINTER WITH CURRENT DATE AND TIME HEADER.
9. WISE - 8080 CPU SIMULATOR WITH ALL REGISTERS, FLAGS, PROGRAMME COUNTER AND STACK.
10. TERMINAL - A SIMPLE TERMINAL PROGRAMME FOR A MODEM.

NCC # 11

1. SOURCE - A TERMINAL PROGRAM SPECIALLY FOR USE WITH 'THE SOURCE'.
2. COXFER - A TERMINAL PROGRAMME DESIGNED FOR USE WITH A REMOTE CCII.
3. STODAT - A TERMINAL PROGRAMME THAT ALLOWS INCOMING DATA TO BE STORED IN MEMORY. THIS DATA CAN THEN BE SAVED TO DISK OR PRINTED. SUPPORTS UPPER OR LOWER CASE.
4. HEX DUMP - DUMPS MEMORY TO SCREEN OR PRINTER IN HEX AND ASCII.
5. DEC DUMP - AS ABOVE BUT DECIMAL AND ASCII.
6. RAM TEST - A RAM TEST PROGRAMME. THIS ONE IS WRITTEN IN BASIC AND IS THEREFORE A LITTLE SLOW.
7. DISK COPY - ALLOWS ANY BLOCKS TO BE COPIED DIRECTLY FROM ONE DISK TO ANOTHER.
8. DISK ZAP - ALLOWS ANY DISK BLOCK TO BE EDITED ON SCREEN WITH CURSOR CONTROLS AND THEN WRITTEN BACK TO DISK.
9. MONITOR - ALLOWS MEMORY TO BE DUMPED, DISASSEMBLED OR MODIFIED. ALSO ALLOWS BREAKPOINTS TO BE SET AND MACHINE LANGUAGE PROGRAMMES TO BE RUN.

NCC # 12

1. SEARCH - ALLOWS MEMORY TO BE SEARCHED FOR ANY STRING CHARACTERS AND THEN PRINTS THE ADDRESSES OF THEIR LOCATION.
2. HEXDEC - A VERY FAST MEMORY DUMP PROGRAMME. DISPLAYS IN HEX, DECIMAL AND ASCII.
3. COPY AID - READS AND SORTS DIRECTORY INTO ALPHANUMERIC ORDER. THEN ALLOWS FILES TO BE COPIED ONTO DESTINATION DISK.
4. PRGZAP - TO EDIT .PRG PROGRAMS ON SCREEN AND THEN REWRITING TO DISK.
5. DISK EDITOR - POWERFUL PROGRAMME. COMMANDS INCLUDE DELETE FILE, DELETE OLD VERSIONS, INCREASE DIRECTORY SIZE, FORMAT DISK, RENAME DIRECTORY, RENAME FILE, NEW VERSION NUMBERS AND DUPE DISK. GOOD INSTRUCTIONS INCLUDED.

NCC # 13

RA ENERGY DEMONSTRATION - DEMONSTRATION OF THE WAYS IN WHICH SOLAR ENERGY CAN BE PUT TO USE IN THE HOME. MAKES EXTENSIVE USE OF GRAPHICS.

NCC # 14

PRISM RESEARCH LANGUAGE DISK - GRAPHICS PACKAGE DESIGNED TO TEACH YOUNG CHILDREN THE SOUNDS OF THE ALPHABET.
THE BEST HIGH RESOLUTION GRAPHICS SEEN ON COMPUCOLOR!

NCC # 15

1. 3DCOST - PRODUCES A THREE DIMENSIONAL GRAPH.
2. DESIGN - PRODUCES EFFECTIVE 3 DIMENSIONAL GEOMETRIC SHAPES ON SCREEN.
3. DRAGON - VERY GOOD SCREEN DISPLAY OF A DRAGON SPITTING FIRE.
4. CUBE - ROTATES A THREE DIMENSIONAL CUBE AT VARIOUS SPEEDS.
5. HAMMER - EXCELLENT ANIMATION OF A HAMMER HITTING A NAIL.

NCC # 16

1. HAND - SCREEN DISPLAY OF DELUXE KEYBOARD WITH ANIMATED HAND TYPING 'COMPUCOLOR' ON IT.
2. MODART - CREATE MODERN DESIGNS WITH BLOCKS OF COLOUR.
3. FACE - AN EXCELLENT SCREEN DISPLAY OF A CALIFORNIAN FACE.
4. SNOOPY - DISPLAY OF SNOOPYON HIS KENNEL WISHING YOU HAPPY BIRTHDAY.
5. NCC718 - 2 DISPLAYS OF THE STAR SHIP ENTERPRISE. WITH FIRING PHASOR.
6. FUNNY - GUESS WHO THIS IS A PICTURE OF!
7. YINYAN - SCREEN DISPLAY OF THE KOREAN FLAG.

NCC # 17

1. CHECK - THIS PROGRAMME ALLOWS YOU TO KEEP A PERMANENT RECORD OF YOUR CHEQUE ACCOUNT DEALINGS.
2. TXTEDT - THIS IS A POOR MAN'S WORD PROCESSER. COMMANDS INCLUDE INSERT, MOVE, JUSTIFY, FORMAT AND OTHERS.
3. BUDGET - ASSISTS IN DESIGNING A BUDGET. YOU MAY INSERT YOUR OWN EXPENDITURE CATEGORIES.
4. FILES - A RANDOM FILES TUTORIAL WHICH ASSISTS IN THE CREATION AND EDITING OF RANDOM FILES.
5. CHECKBOOK - ANOTHER CHEQUE BOOK PROGRAMME.
6. HOME BUDGET - THIS PROGRAM CAN EASILY BE TAILORED TO YOUR NEEDS.

NCC # 18

1. FORMS - A TUTORIAL PROGRAMME ON GEOMETRIC FORMULAE.
2. GEOSIZ - CALCULATES ALL DIMENSIONS OF ANY GEOMETRIC OBJECT. GOOD GRAPHICS ASSIST.
3. LENSES - ALLOWS YOU TO DESIGN A LENS ON SCREEN. GREAT GRAPHICS!
4. ROSES - DRAWS ROSE GRAPHS.
5. LISSA - DRAWS LISSAJOUS FIGURES.
6. ATTEN - GIVES FORMULAE AND ASSISTS IN THE DESIGN OF ATTENUATORS.
7. GAMMA - ASSISTS IN DESIGNING ANTENNA.
8. VOCAB - READERS DIGEST VOCABULARY TESTS.
9. DETERM - COMPUTES THE DETERMINANT OF A MATRIX.

CHIP LIBRARY

COMPLIMENTS OF ROCHESTER (NEW YORK) USERS'S GROUP.

CHIP # 1
1. FLIGHT SIMULATION - AN EXCELLENT SIMULATION OF A BOEING 747B. YOU MUST TAXI TO THE RUNWAY, TAKE OFF, NAVIGATE TO DESTINATION, LAND AND TAXI TO DOCKING BAY.
2. LABYRINTH - EXCELLENT GRAPHICS GIVE YOU AN INSIDE, 3D PERSPECTIVE AS YOU FIND YOUR WAY THROUGH THE PASSAGES OF A MAZE.
3. BOWLING ALLEY - PLAY BOWLS WITH GRAPHICS AND AUTO SCORING - TWO PLAYERS.

CHIP # 2
1. MONOPOLY - FOR UP TO 4 PLAYERS WITH COMPUTER TAKING CARE OF THE MONEY AND NO BREAKING OF THE RULES!
2. MAZE - FIND YOUR WAY OUT OF THE MAZE WITHOUT BEING CAUGHT BY THE MAZE MASTER - ONE PROBLEM, YOU ARE BLIND!
3. HYPERSPACE - SET THE SIZE AND STRENGTH OF YOUR OPPONENTS AND THEN WAGE WAR THROUGH THE GALAXY.

CHIP # 3
1. WUMPUS - BEST VERSION YET.
2. BATTLESHIP - EXCELLENT VERSION OF OLD SCHOOLBOY FAVOURITE - YOU VS THE CCII.
3. INTEGER - SIMPLE NUMBER GAME.
4. SHOOT OUT - REAL TIME COWBOY ACTION - WITH GRAPHICS.
5. CAMEL - VERY AMUSING "MINI" ADVENTURE.
6. NOUGHTS & CROSSES - GOOD GRAPHICS, BUT YOU CAN WIN.
7. BAGELS - "MASTERMIND" WITH WORDS.
8. BAGELS - "MASTERMIND" WITH NUMBERS.
9. BINGO - TAKES THE PLACE OF A BINGO CALLER.

CHIP # 4
1. SUPER STAR TREK - STAR TREK, WITH ALL SORTS OF GIZMOS.
2. MASTERMIND - SMARTER THAN THE USUAL ONE.
3. JURY - SHOWS THE PROBLEMS IN TRYING TO PICK A JURY COMPATIBLE WITH THE YOUR CLIENT'S INTERESTS.
4. PSYCHIATRIST - "ELIZA" STRIPPED TO A RATHER SLOW BASIC PROGRAM. YOU ARE THE CLIENT FOR A RATHER UNCOMMUNICATIVE "SHRINK".

CHIP # 5
1. ROULETTE - TRADITIONAL WITH NICE DISPLAYS.
2. BACKGAMMON - 2 PLAYERS, GREAT DISPLAYS.
3. GREED - NICE GAME, YOU VS CCII, USES SCROLLING WELL.
4. BIORHYTHMS - NOT THE BEST VERSION SEEN.
5. SPACE COLONY - 2 PLAYER GAME, ONCE AGAIN GOOD SCROLLING.

CHIP # 6
1. REVERSE - NUMBER MANIPULATION GAME.
2. ROVER ROBOT - FUN, SLOW GAME. NICE DISPLAYS.
3. 15 GAME - GOOD DISPLAY VERSION OF THE OLD PLASTIC SLIDE BOARD.
4. TANKS - GOOD VERSION, YOU VS CCII.
5. BLOCKADE - SIMPLE 2 PLAYER GAME.

CHIP # 7
THIS IS A DEMONSTRATION DISK CONTAINING MANY USEFUL ROUTINES FOR GAMES AND SCREEN USEAGE.

CHIP # 8
1. NICHE - AN ECOLOGY SIMULATION WHERE YOU SPECIFY THE CONDITIONS FOR THE ENVIRONMENT OF A CHOSEN ORGANISM AND SEE WHAT HAPPENS.
2. ELECTRIC COMPANY - GAME TO USE COAL OR OIL TO PROVIDE SUFFICIENT POWER FOR THE COMMUNITY. YOU MAKE ALL THE DECISIONS.
3. INSPECTOR CLEWSO - QUESTION THE SUSPECTS IN ORDER TO DETERMINE WHODUNIT, WHERE AND WHEN.
4. TRAP - MANOUVER YOUR SNAKE AROUND THE SCREEN AND TRY TO TRAP YOUR OPPONENT.
5. 3D TIC TAC TOE - GOOD GRAPHICS AND A SMART COMPUTER MAKE THIS A CHALLENGING GAME.
6. CHECKERS - TRADITIONAL WITH EXCELLENT GRAPHICS.

CHIP # 9 & 10
TINY C INTERPRETER AND SOURCE CODE. - NO INSTRUCTIONS

CHIP # 11
1. BOUNCE - EXCELLENT 2 PLAYER GAME, GOOD DISPLAYS.
2. MILL - GOOD 2 PLAYER BOARD GAME.
3. HANGMAN - THIS WILL ADD WORDS TO THE VOCAB FOR THE GAME ON YOUR HANGMAN DISK.
4. LIFE - .....YET ANOTHER.
5. CALCULATION - SOLITAIRE CARD GAME.
6. SLY FOX - SOLITAIRE CARD GAME.

CHIP # 12
SET OF CARD & DICE GAMES:
1. COMPUDICE.
2. RUMMY.
3. CRAG.
4. UNO.

CHIP # 13
ANOTHER SET OF CARD & DICE GAMES:
1. LA BELLE LUCIE (CARD)
2. IDIOT'S DELIGHT (CARD)
3. FINANCIER (CARD)
4. KISMET (DICE)
5. ZILCH (DICE)

CHIP # 14
REAL TIME ASTEROIDS.

CHIP # 15
1. PRO-FOOTBALL - GRID IRON, EXCELLENT DISPLAYS.
2. OIL COMPANY - RUNNING A COMPANY, USES SOME GRAPHICS.
3. LUNAR LANDER - ANOTHER, AND NOT THE BEST.

CHIP # 16
1. BLACK BOX - FIND HIDDEN BALLS IN A GRID.
2. CROSSWORD - GENUINE CROSSWORD WITH CLUES.
3. CROSSWORD - CREATE NEW GAMES FOR NUMBER 2.
4. CIVIL WAR - INTERESTING 1-2 PLAYER THOUGHT GAME.
5. FOX & HOUNDS - YOU VS THE CCII - ON BOARD.
6. ZONEX - THOUGHT GAME.

CHIP # 17
1. SUPER MONOPOLY - A FEW MORE OPTIONS, NEEDS 32K.
2. TANKS - .....ANOTHER!

CHIP # 18
1. ROBOT CHASE - A GROUP OF ROBOTS ARE CHASING YOU AROUND. TRY TO ELUDE THEM WHILE CAUSING THEIR DESTRUCTION.
2. PRISON ESCAPE - IN ORDER TO ESCAPE FROM THIS PRISON YOU MUST ACHIEVE A NUMBEROF REAL TIME TASKS SET FOR YOU BY THE COMPUTER.
3. ICBM - DIRECT YOUR MISSILES TO DESTROY THE INCOMING ENEMY MISSILES. GRAPHIC, NON-REAL TIME.
4. REAL TIME STAR TREK - STAR TREK WHICH GIVES YOU ONLY LIMITED TIME TO SELECT COMMANDS. SPEED IS USER SELECTABLE.

CHIP # 19
1. DOG STAR - A MINI"ADVENTURE" GAME. RESCUE THE PRINCESS WHILE AVOIDING THE ENEMY SOLDIERS.

CHIP # 20
1. ADVENTURE - THE NOW CLASSIC GAME ADAPTED FOR COMPUCOLOR. THERE ARE NO INSTRUCTIONS - YOU'RE ON YOUR OWN.

CHIP # 21
1. DUNGEONS & DRAGONS - A COMBINATION OF ADVENTURE AND A MAZE WITH GRAPHICS. GAME CAN BE SAVED HALFWAY THROUGH.

CHIP # 22
UTILITY DISK:
1. READ DISK - READS AN ENTIRE DISK SEEKING BAD SPOTS.
2. CLEAR DISK - ERASES A DISK WITH A SPECIFIED PATTERN.
3. DUP DISK.
4. DUP FILE.
5. CHANGE DIRECTORY INFORMATION - CHANGE NAME, IT'S COLOUR OR THE NUMBER OF BLOCKS IN THE DIRECTORY.
6. SOURCE OR TEXT FILE PRINTER.
7. DIRECTORY MANAGEMENT - KEEPS TRACK OF VARIOUS DISKS BY COPYING THEIR DIRECTORIES ON TO A MASTER DISK.
8. BASE CONVERSIONS - CONVERTS NUMBERS BETWEEN HEX, OCTAL, BINARY & DECIMAL.
9. BASE ARITHMETIC - BASE CONVERSION PROGRAM WHICH ALSO CALCULATES USING MIXED BASES.
10. DISPLAY MANAGEMENT - KEEPS TRACK OF SCREEN DISPLAYS.
11. BASIC PROGRAM PRINTER PATCH - ADD THIS TO A LISTING TO PRINT OUT IN A FORMATTED OUTPUT.

CHIP # 23
1. SCREEN DISPLAY EDITOR - A SOPHISTICATED BASIC PROGRAM TO AID IN PRODUCING SCREEN DISPLAYS USING TEXT, COLOUR CHANGES AND PLOTTING ROUTINES.
2. SOUNDWARE MUSIC EDITOR.

CHIP # 24
1. HOT AIR BALLOON RACE - THE DISPLAY DRAWS A MAP OF THE U.S.A. AND RECORDS THE PROGRESS OF 2 PLAYERS RACING TOWARDS BOSTON FROM CALIFORNIA.
2. CLUMSY - MOVE AROUND SCREEN AVOIDING HIDDEN OBSTACLES.
--- CONTINUED ---

CHIP # 24 CONTINUED
3. WORDS - GUESS A 5 LETTER WORD THE COMPUTER HAS CHOSEN.
4. ASK ME - THE COMPUTER IS ABLE TO ANSWER ON SEVERAL TOPICS. ITS VOCABULARY IS EXPANDABLE.
5. MAZE1 - PRINT ANY SIZE MAZE ON YOUR COMPUTER.
6. ANIMAL. TEACH THE COMPUTER TO DISTINGUISH BETWEEN HUNDREDS OF ANIMALS.
7. FLIP - THE CCII TRIES TO OUTGUESS YOU IN THIS GAME.
8. BRAIN TEASER - GIVEN A 3X3 ARRAY OF CELLS, YOU MUST FLIP THEM ACCORDING TO THE RULES UNTIL YOU HAVE THEM IN A PRE-DEFINED PATTERN - NOT EASY.

CHIP # 25
1. 3-D PLOTTING PACKAGE - JIM MINOR'S PROGRAMS ALLOW YOU TO CREATE AND EDIT 3-D DISPLAYS. REQUIRES 32K. EXTENSIVE DOCUMENTATION IN PROGRAM.

CHIP # 26
1. OP AMP OPTIMIZATION CALCULATOR.
2. RESISTIVE PI AND T NETWORK COMPUTATION.
3. POWER, DB & VOLTAGE COMPUTATION.
4. PARALLEL & SERIES CAPACITANCE CALCULATION.
5. PARALLEL & SERIES RESISTANCE CALCULATION.
6. OHM'S LAW - POWER, CURRENT, VOLTAGE AND RESISTANCE CALCULATION.
7. CRAMER'S RULE - FIND X, Y AND Z IN 3 POLYNOMIALS.
8. SIMULTANEOUS EQUATIONS - SOLVE UP TO 40 SIMULTANEOUS EQUATIONS.
9. FUNCTION PLOTTER - GENERAL PLOTTING PROGRAM FOR ANY FUNCTION Y=F X).
10. EXTERNAL BALLISTICS - TRAJECTORY CALCULATIONS.
11. SHORTEST ROUTE - CALCULATES THE SHORTEST ROUTE BETWEEN POINTS.

CHIP # 27 (NB SOME OF THESE PROGRAMS FOR 6.78 ONLY)
1. SCREEN R-M TEST - TESTS SCREEN MEMORY AND IDENTIFIES FAULTY CHIPS.
2. 'SMART PRINTER DRIVER' PRINTS .SRC FILES ON PRINTER AT SELECTABLE BAUD RATES. ALSO PRINTS PAGE NUMBERS AND LEAVES TOP & BOTTOM MARGINS. YOUR PRINTER MUST RESPOND TO 'CTL L' - FORM FEED.
3. TRANSFER - CCII TO CCII DATA TRANSFER PROGRAM SENDS OR RECEIVES MEMORY BLOCKS OR BASIC PROGRAMS AT 300 BAUD THROUGH THE SERIAL PORT.
4. DISASSEMBLER - MACHINE LANGUAGE PROGRAM FOR SCREEN OR PRINTER OUTPUT.
5. DEBUG - MACHINE LANGUAGE DEBUGGER. REQUIRES 32K. NO DOCUMENTATION.
6. CONTRX - TURNS THE CCII INTO A DUMB TERMINAL.
7. MANAZIR'S MONITOR - SOURCE FILE INCLUDED.
8. ASSEMBLER UPGRADE - UPGRADES YOUR CCII ASSEMBLER TO GIVE YOU OPTIONS SUCH AS LIST TO SCREEN OR PRINTER.
9. LLIST - LINE PRINTER PATCH FOR BASIC PROGRAMS.
10. FORMATTER - CCII DISK FORMATTER. (NOT ISC'S)

CHIP # 28
1. TYPING - THE CCII USES DISPLAYS FOR TYPING.
2. MUSICAL PITCH PRINTER - PRINTER NEEDED.
3. STATES & CAPITALS QUIZ - QUIZ ON AMERICAN GEOGRAPHY.
4. CALCULATOR - SIMULATES A CALCULATOR. GOOD DISPLAYS.

CHIP # 29
1. GRAPHICS DEMO - 1.
2. GRAPHICS DEMO - 2.
3. GRAPHICS DEMO - 3.
ALL THESE ARE UP TO THE USUAL CHIP STANDARD.

CHIP # 30
1. QUICK TURN - SUITS 1-9 PLAYERS, SIMPLE GAME, GOOD DISPLAYS.
2. ROBOT CHASE - 1 PLAYER, GOOD DISPLAYS.
3. MINER - 1 PLAYER, GOOD GAME, QUITE GOOD DISPLAYS.
4. CRAPS - STANDARD U.S. GAME - LIMITED.
5. DRAGON & DUNGEON - SIMPLE VERSION.
6. TIME BOMB - SIMPLE GAME TO DEFUSE A TIME BOMB WITHOUT EXPLODING.

CHIP # 31
ASTEROIDS. REAL TIME, ARCADE GAME, ADAPTED BY B. MULDOWNEY FOR 6.78 & 8.79 AND USE WITH JOYSTICKS.

CHIP # 32
PRINTER UTILITIES. PRINTS FORMATTED BASIC PROGRAMS.

CHIP # 33
CASTLE QUEST - 16 & 32K VERSIONS. EXCELENT ADVENTURE TYPE GAME WITH DISPLAYS.

CHIP # 34
1. STAR MERCHANT - GOOD 1 PLAYER GAME. RELATIVELY COMPLEX - THOUGHT GAME.
2. GIANT MONSTER COMBAT - EXCELLENT 1 PLAYER GAME, ONLY LIMITED SCREEN USE.
3. TRUCKER - CLEVER THOUGHT GAME. WELL PREPARED TO SIMULATE THE CONDITIONS OF A TRUCKING TRIP.
4. INDY RACE TRACK - LIMITED REAL TIME RACE WITH SIMPLE GRAPHICS AND SOME SOUND.

CHIP # 35, 36 & 37
EXCELLENT SET OF HIGH LEVEL MATHS TUTORIALS. AIMED AT SENIOR LEVEL SECONDARY SCHOOLS AND TERTIARY EDUCATION. VERY WELL PRESENTED AND THOUGHT OUT.

CHIP # 39
1.FINANCE - STOCKS BONDS AND HOME LOAN INTEREST CALCULATOR

CHIP # 44
1. MISSING LINK - PUZZLE TO FIND THE LINK. GOOD DISPLAYS.
2. RUBIK'S CUBE - EXCELLENT GRAPHICS TO SOLVE THAT THING YOU USE AS A FOOTBALL AROUND YOU LOUNGE ROOM.
3. O'NO99 - CARD GAME WITH GOOD DISPLAYS.

CHIP # 45
1.PRESSUPS - GOOD GRAPHIC GAME
2.REACTION - REFLEXES GAME WITH SOUND
3.POKER - CARD GAME
4.BLACKBOX - GAME OF DEDUCTION

CHIP # 46 & 47
FORTH FOR THE CCII- LANGUAGE & SCREENS. CHIP MANUAL IS AVAILABLE FROM THE LIBRARY ON A LOAN (PHOTO-COPY IT YOURSELF).

CHIP # 48
ARCADE TYPE SPACE GAMES FOR 6.78 & 8.79

CHIP # 50
SIMULATION OF HUMAN DIGESTIVE SYSTEM AS A GAME

CHIP # 51
MATHS TRAINING PROGRAMS FOR THE YOUNG.

CHIP # 52
1.OCTOS - GAME OF DEDUCTION
2.MOON - SURVIVE ON THE MOON
3.HORSE - QUICK GAMBLING ON THE NAGS.
4.STONEEVILLE - ADVENTURE
5.BIORHYTHMS

CHIP # 53
1.TEACH USE OF FIELDS
2.SORT PROGRAMS
3.SEARCH FOR VARIABLES

CHIP # 55
1.COPY DISK
2.DIRECTORY PRINT
3.TINY C
4.TINY C - BASIC INSTRUCTIONS
5.TWO GOOD DIASSEMBLER PROGRAMS

CHIP # 59
1. ADVENTURE - REWORKED WITH ALL ROOMS
2. WUMPUS II
3. JOYSTICK - DEMO FOR USING TWO JOYSTICKS
4. SNAKE - WITH SOUND PATCH

CHIP # 60
1. BREAKOUT - GOOD VERSION WITH SELECTABLE SPEEDS.
2. TYCOON - KEITH.O.'S LATEST VERSION
3. ANDRIOD NIM - GREAT BIT OF GRAPHICS.
4. SOLITAIRE - GOOD VERSION

CHIP # 79
COSMIC CONQUEST, A SPACE STRATEGY GAME, WINNER OF BYTE CONTEST DEC 1982, BY A SATORORI-ANGUS, ADAPTED BY TOM NAPIER. WRITTEN IN FORTH FOR 7/68 AND 8/79

CHIP # 83
TINY PASCAL. COMPILER UTILITY -BY JIM MINOR. REQUIRES FIG FORTH DISK SET (CHIP 46/47) AND 84. NO MANUAL.

CHIP # 84
TINY PASCAL. STARTER SCREENS. 50 SCREENS INCLUDING, EDITOR ERROR MESSAGES, SCREEN PARITY CHECK, AND 8080 ASSEMBLER. (NOT AVAILABLE YET BUT COMING)

CHIP # 92
UTILITIES BY JIM HELMS
SOURCE DIVIDER - REMOVES SECTION OF SRC. FILES.
REM STRIPPER - REMOVES REMARKS FROM SRC. FILES.
SOURCE MERGE - MERGES TWO OR MORE SRC. FILES
BASE2 PRESET - USED TO PRESET A BASE2 PRINTER.
PRISM DUMP - IDS. PRISM PRINTER DUMP PROGRAM.
BIORYTHM PRINT - PRINTED COPY OF YOUR BIORYTHM.
GRAPHICS SET GENERATOR - CREATE AND EDIT YOUR OWN

CHIP # 94
SUBMARINE PATROL - BY WALLY RUST, SUB PATROL IN WWII.
SPIDER MOUNTAIN - BY R TAUBOLD, ADVENTURE IN A MOUNTAIN.
TROUBLE WITH TRIBBLES - BY C BELL, YOUR SURE TO HAVE TRIBBLE TROUBLE.

CHIP # 115
CAPTURE THE FLAG - BY DAVE SUITS, GREAT WAR GAME ON SCREEN INSTRUCTIONS PUBLISHED IN CUVIC.
MILLE BORNES - BY B GREEN, A FRENCH CARD GAME.
MINDMASTER - ENTER A CODE AND THIS HUMMER WILL BREAK IT!
LINE FIVE - TIC TAC TOE WITH VARIATIONS.

CHIP # 117
BLKDIS - WRITES DISK BLOCK TO SCREEN.
DISKED - DISK BLOCK EDITOR.
TRACE - EXECUTES A PRG. PROGRAM LINE BY LINE, WHILE PRINTING OUT TO DISK OR PRINTER, THE REGISTERS, STATUS WORD AND OTHER KEY VALUES. WITH INSTRUCTIONS BY TOM WULFF.

CHIP # 120
FORTH UTILITY (32K) - BY BILL GREEN
FORTH8 - FORTH COMPILER (LOADS AT 8200H)
FORTH4 - FORTH COMPILER (LOADS AT 4000H)
CFORTH - CONTAINS A SCREEN EDITOR, TWO DISASSEMBLERS, AND A 8080 ASSEMBLER. REQUIRES #121.

CHIP # 121
FORTH UTILITY SCREENS FOR #120.

NOTES ON CHIP LIBRARY
PLEASE NOTE THAT BECAUSE THE CHIP LIBRARY USE'S A PROGRESSIVE NUMBERING SYSTEM WHICH INCLUDES DISKS FROM OTHER LIBRARY'S, THERE ARE GAPS IN OUR LIST OF THE LIBRARY. IN MOST CASES THESE DISKS ARE HELD BY CUVIC IN THE ORIGINAL LIBRARY LISTING.

MORE CHIP DISKS ARE ON ORDER FROM THE ROCHESTER GROUP.

CUWEST LIBRARY -
COURTESY OF WESTERN AUSTRALIAN USERS'S GROUP.

CUWEST #1
A COLLECTION OF PROGRAMS THAT DEMONSTRATE HOW TO USE MANY OF THE CCII FEATURES. INCLUDES GRAPHICS AND KEYBOARD DEMOS PLUS A USEFUL DISK TRACK CHECK PROGRAM AND SOME GAMES.

CUWEST #2
MORE DEMONSTRATION INCLUDES BACKGAMMON, SCRABBLE, SPACE FLIGHT, CHECKERS AND MATCHES.

CUWEST #3
INCLUDES HIGH PRECISION MULTIPLICATION, CAR REPLACEMENT COST ANALYSIS, BREAK-OUT, RUBIK'S CUBE, SPACE INVADERS AND PROGRAMS TO RUN CARD READERS, DIGITIZERS AND PLOTTERS.

CUWEST #4
MORE GRAPHICS DEMOS AND CARD READER PROGRAMS - ALSO INCLUDES DISK DRIVE ALIGNMENT PROGRAM.

CUWEST #5
INCLUDES DISK DUMP, LIVESTOCK MANAGEMENT, YAHTZEE GAME, READS SOURCE FILE, RAM TEST AND GRAPH DRAWING PROGRAMS.

CUWEST #6
GRAPHICS EDITOR - A VERY POWERFUL EDITOR FOR SCREEN DISPLAY CREATION. DRAWING FACILITIES INCLUDE COLOUR SELECTION, BORDER DRAWING, DRAW A BOX, DRAW A COLOUR BLOCK, FUNCTION PLOTTING, POINT PLOTTING, CIRCLE ELLIPSE OR ARC DRAWING, IRREGULAR LINE DRAWING, TEXT INSERTION AND BLINKING. SCREEN MANIPULATION COMMANDS INCLUDE SWAP, TRANSFER, RUB OUT LAST ENTRY, ERASE, SCROLL, DISPLAY MENU AND GRID OVERLAY.
FULL DISK ACCESS COMMANDS ARE INCLUDED.

CUWEST #7
DIGAME.- A COMPUTER ASSISTED LEARNING GAME WHICH SIMULATES THE HUMAN DIGESTIVE SYSTEM. THREE LEVELS OF PLAY FOR UP TO 6 PEOPLE.

CUWEST #8
MATHS - A COMPUTER ASSISTED LEARNING PACKAGE FOR CHILDREN. COVERS ADDITION, SUBTRACTION, LONG MULTIPLICATION, AREA & PERIMETER PRACTICE AND A FACTOR GAME. VARYING LEVELS OF DIFFICULTY.

CUWEST #9
ENGLISH & GEOGRAPHY - A COMPUTER ASSISTED LEARNING PACKAGE. INCLUDES PROGRAMS ON AUSTRALIAN CAPITAL CITIES AND WEST AUSTRALIAN. INCLUDES GAME OF HANGMAN.

CUWEST #10, 11, 12 & 13.
ENGLISH TUTOR - THESE SERIES OF DISKS CONTAIN A SERIES OF ENGLISH TUTORIALS WRITTEN BY JOHN NEWMAN. THEY ARE WRITTEN AT A HIGH LEVEL AND WOULD PROBABLY BE SUITABLE

CANADIAN User DISK LIBRARY - Courtesy of FORUM

DISK 1A

SNOOPY	GRAPHIC	FAMOUS CANINE PHILOSOPHER
OCEAN	DISPLAY	SEASCAPE BY MOONLIGHT
DUP	UTILITY	DISK DUPLICATION 1 OR 2 DISK DUP
FORMAT	UTILITY	FORMAT CCII DISKETTES
3DCOST	BUSINESS	3D GRAPH GENERATION LISTED IN COLORCUE
TAYLOR	ENGINEERI	DEMONSTRATION OF NUCLEAR REACTOR CONTROL
LOAN	FINANCE	LOAN AMORTIZATION FROM SAMPLER DISKETTE
SALES	FINANCE	BUSINESS GRAPHICS DEMONSTRATION
COLORS	GRAPHICS	FOREGROUND AND BACKGROUND COLOR DISPLAY
TRIANG	GRAPHICS	DEMONSTRATE SPECIAL CHARACTER SYMBOLS
SCROLL	GRAPHICS	DEMO OF SCROLL PATCH
15PUZZ	GAME	ARRANGE 15 NUMBERS IN 4 BY 4 ARRAY
HYPER	GAME	SPACE SHOOT EM UP GAME
3DDSGN	GRAPHIC	DISPLAY OF AN OBJECT IN 3D & COLOR
PERCOM	MATH/STAT	PERMUTATIONS & COMBINATIONS CALCULATIONS

DISK 1B

ASC	UTILITY	ASCII DISPLAY OF MEMORY FROM ADDRESS N
DIS	UTILITY	SIMPLE DISASSEMBLER
PLOTTR	MATH/STAT	SIMPLE MATH FUNCTION PLOTTER
OILCO	GAME	OIL COMPANY SIMULATION GAME
UTIL01	UTILITY	DELETE MULTIPLE DISK FILES-SINGLE DRIVE SYSTEM
UTIL02	UTILITY	SINGLE DRIVE FILE COPIER
UTIL03	UTILITY	DIRECTORY NAME CHANGE
DUMP	UTILITY	SCREEN
SEAWAR	GAME	ARCADE SEA BATTLE- TORPEDO THE ENEMY SHIPS
ELIZA	GAME	A WELL KNOWN EXERCISE IN ARTIFICIAL INTELLIGENCE
ALPHA	UTILITY	ALPHABETIC SORT
CHARAC	GRAPHICS	DEMO PRINT SIZES
FRACT	SUBROUTIN	EXTENDED PRECISION DIVIDE SUBROUTINE

DISK 2A

CLEWSO	GAME	MUCH LIKE BOARD GAME CLUE
CAMEL2	GAME	GET YOUR CAMEL ACROSS THE DESERT...ALIVE
BASE2	UTILITY	PRINT UTILITY FOR BASE-2 PRINTER
PRTTST	UTILITY	TEST PROGRAM USED WITH BASE2 FOR THAT PRINTER
CHECKB	ACCOUNTIN	CHECKBOOK WITH DATA FILES
MAZE	GAME	DRAW AND RUN MAZE
HIQ	GAME	PEG GAME OF HI-Q
QUEST	GAME	FIND THE PIRATE GOLD..ADVENTURE GAME

DISK 2B

INSTAR	GAME	INSTRUCTIONS FOR STAR.BAS/DOGSTAR
STAR	GAME	ADVENTURE TYPE STAR WARS GAME
CALNDR	DISPLAY	DISPLAY A CALENDAR FOR A GIVEN MONTH
ESTATE	FINANCE	CASH FLOW ESTIMATION FOR REAL ESTATE PURCHASE
TIME	DEMONSTRA	TIME DISPLAY FOR WORLD CITIES
RELOCA	UTILITY	STATIC RELOCATOR
CIRCLE	GRAPHICS	DEMONSTRATION OF CIRCLE DRAWING
DRAW5	GRAPHICS	DEMONSTRATION OF CCII GRAPHICS CAPABILITIES
LUNAR	GAME	LUNAR LANDING GAME...1 OF SEVERAL IN LIBRARY
GOMOKO	GAME	GAME OF "GO" FOR CCII
HURKLE	GAME	HUNT THE CREATURE CALLED THE HURKLE

DISK 3A

ROULET	GAME	EUROPEAN ROULETTE GAME
ROVER	GAME	HELP ROVER THE DOG CROSS A SURFACE GRID
DEPCHG	GAME	FIND AND DEPTHCHARGE THE SUBMARINE
SORCER	GAME	ADVENTURE TYPE GAME WITH SWORDS AND SPELLS
JOUST	GAME	KNOCK YOUR OPPONENT DOWN & WIN THE PRINCESS
ONECHK	GAME	CHINESE CHECKERS...FOR 1 PLAYER
BOUNCE	GAME	ARCADE BOUNCING BALL

DISK 3B

FUVAL	FINANCE	CALCULATE FUTURE VALUE OF AN INVESTMENT
FURDEP	FINANCE	CALCULATE FUTURE VALUE OF REGULAR DEPOSITS
REGDEP	FINANCE	CALC REGULAR DEPOSIT REQUIRED FOR FUTURE VALUE
INVANN	FINANCE	CALC REGULAR WITHDRAWALS FROM AN INVESTMENT
INTINV	FINANCE	CALC INVESTMENT REQUIRED FOR A FUTURE VALUE
MININV	FINANCE	CALC MINIMUM INVESTMENT FOR WITHDRAWALS
EFFINT	FINANCE	CALC EFFECTIVE INTEREST RATE FOR KNOWN INVESTMENT
EARINT	FINANCE	CALC AND PRINT EARNED INTEREST TABLE FOR INVEST.
DEPRAT	FINANCE	CALC ANNUAL DEPRECIATION RATE OF INVESTMENTS
DEPAMT	FINANCIAL	CALCULATE AMOUNT DEPRECIATED FOR A YEAR OF INVEST
SALVAL	FINANCIAL	CALC SALVAGE VALUE OF ITEM AT END OF A GIVEN YEAR
COMPAP	FINANCIAL	CALC DISCOUNT AND NET COST OF COMMERCIAL PAPER
LNPRIN	FINANCE	CALCULATE PRINCIPAL ON A LOAN
REGPAY	FINANCIAL	CALCULATE REGULAR PAYMENTS ON A LOAN
LASPAY	FINANCIAL	CALCULATE LAST PAYMENT ON A LOAN
REMBAL	FINANCIAL	CALC REMAINING BALANCE ON A LOAN
ANNINT	FINANCIAL	CALCULATE ANNUAL INTEREST RATE OF A LOAN
LNTERM	FINANCIAL	CALCULATE PERIOD OF TIME REQD TO REPAY A LOAN
NOMINT	FINANCIAL	CALCULATES NOMINAL INTEREST RATE ON INVESTMENTS
INTRST	FINANCIAL	CALCULATE DECLINING INTEREST
INVEST	FINANCIAL	EST CASH FLOW TO CARRY REAL ESTATE PURCHASE
DECDMP	UTILITY	DECIMAL AND ASCII MEMORY DUMP
STOCKM	GAME	STOCK MARKET SIMULATION GAME
BCKGM3	GAME	BACKGAMMON FOR 2 PLAYERS

DISK 4A

TXTEDT	WORD-PROC	WORD PROCESSOR
FOOTBL	GAME	AMERICAN FOOTBALL GAME
FORT	GAME	COMMAND F-FORT AGAINST INDIAN ATTACK
PRETAN	GAME	INSTRUCTIONS FOR GAME OF TANK
TANK	GAME	TANK BATTLE AGAINST CCII
SECRED	UTILITY	DISPLAY DISKETTE SECTORS TO SCREEN

DISK 4B

MASTMD	GAME	TRADITIONAL MASTERMIND GAME
BACKIN	GAME	RUN INSTRUCTIONS FOR BACKGAMMON GAME
BACKG	GAME	BACKGAMMON GAME
SDINFO	TELEPROCE	INSTRUCTIONS FOR STODAT TERMINAL CNTRL PR
STODAT	TELEPROCE	TERMINAL CONTROL PROGRAM FOR CCII
STODAT	TELEPROCE	TERMINAL CNTRL PROGRAM FOR CCII (2 PARTS)
DISASM	UTILITY	DISASSEMBLER
GAMMA	ENGR/SCIE	CALC COAXIAL CABLE IMPEDENCE TO A BEAM ANTENNA

DISK 5A

POKER	GAME	DRAW POKER AGAINST CCII
BRAIN	GAME	GRAPHICAL BRAIN TEASERS
HOMBUD	FINANCIAL	HOME BUDGET PROG. SEE 117
POND	ENGR/SCIE	CALCULATE SIZE/VOLUME OF RECTANGULAR PITS
FFOURT	MATH/STAT	FAST FOURIER TRANSFORM
STATS1	MATH/STAT	CALC MEAN VARIANCE STD DEV STD ERROR OF EST
CCLOCK	GRAPHICS	DIGITAL CLOCK

DISK 5B

FLTINS	GAME	INSTRUCTIONS FOR B747 FLIGHT SIMULATOR
DAYFLT	GAME	FLIGHT SIMULATOR FOR KNOWLEDGABLE PILOTS
CRASH	GAME	PART OF DAYFLT FLIGHT SIMULATOR PACKAGE
GNDWTR	ENGR/SCIE	EVALUATE GROUND WATER POLLUTION

CUVIC COMPUTER USERS OF VICTORIA - BOX 420 CAMBERWELL - 3126
REGISTERED BY AUSTRALIAN POST - PUBLICATION No VBH 5086

PRESIDENT - KEN WINDER,
8 BRINDY CRESCENT,
EAST DONCASTER, 3109
VICTORIA, AUSTRALIA.
TELE. 03 848 1422

SECRETARY - TED STUCKEY,
TELE. 03 836 8732

TREASURER - HOWARD RICE,
TELE. 03 277 2957

EDITOR - KEN SMITH,
49 BANKSIA CRESCENT,
HOPPERS CROSSING, 3030
VICTORIA, AUSTRALIA.
TELE. 03 749 5579

NOVEMBER 1986

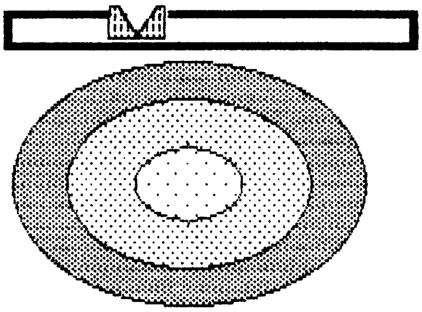
PRICE 50 CENTS MONTHLY

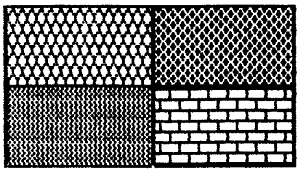
NEXT MEETING - WEDNESDAY 10th. DECEMBER 1986

IN THE SHOP AT THE
SURREY HILLS NEIGHBORHOOD CENTRE
157 UNION ROAD, SURREY HILLS, 3127
(MELWAYS MAP 46: H,10. NEAR THE STATION)
COMMENCES - 8:00 P.M.

THIS PAGE HAS BEEN SET UP USING "**FONTASY**" BY PROSOFT. AS YOU CAN SEE THIS IS A VERY VERSITILE PROGRAM SOMETHING LIKE " PRINTSHOP" BUT WITH A LOT MORE BELLS AND WHISTLES.

For a start it can draw shapes and fill them.





AND YOU CAN LINE UP THE TEXT TO FIT EXACTLY IN THE THE SPACE !

NOT ONLY DOSE IT HAVE ALL THESE GREAT FUNCTIONS BUT THERE ARE 20 DIFFERENT FONTS INCLUDED IN THE PACKAGE OUT OF THE 300 WHICH ARE AVAILABLE.

THERE ARE ALSO FEATURES THAT ALLOW YOU TO SET UP TEMPLATES . THESE ARE PAGES WITH PRE-RECORDED COLUMNS AND SPACES FOR PICTURES AND OTHER ITEMS SUCH AS THE HEADINGS FOR THIS PAGE.

You can draw blocks, move them , copy them , invert them, turn them upside down
uʍop ǝpısdn ɯǝɥʇ uɹnʇ

make a mirror image ǝƃɐɯı ɹoɹɹıɯ ɐ ǝʞɐɯ

PICTURES FROM ART.PIC FILES

FONTASY HAS THESE FONTS.

GOTHIC in UPPER and lower case

STANDARD *STANDARD ITALICS and lower*

BLOX

BANNER+ENDS

JULY 4TH

MOON & SHADOW

SCRIPT NO1 and FLOW

SPL in both UPPER and lowercase.

SPL ITALICS in both UPPER AND lower.

MICROPRINT in UPPER and lowercase.

MINICUBE in UPPER and lower

PRETORIAN with lower

CALLIGRAPHY with lower

CHANNCERY with lowercase.

ROMANS3 with low.

MERRY CHRISTMAS AND A HAPPY NEW YEAR
TO YOU AND YOURS FROM CUVIC

PUBLIC DOMAIN WORDPROCESSORS

Bruce Marshalls article in the last CUVIC promted me to have a look at the wordprocessor programs available in the public domain.

There are currently two very good word processors in the CUVIC IBM Library, PC WRITE Ver 2.6 and CHIWRITER. Each has its own strong points and while I personally have not used either for more than a few odd jottings I can assure you that they are very good value for the money.

Both are what is called SHAREWARE, that is they are placed in the public domain by their authors so that the user can try them out and if they decide to use them, send the author a fee, for which they recieve the lastest version plus support and printed manuals. The cost of US$75, are moderate compared with the cost of commercial packages.

Both come with instruction manuals on disk, which can be printed out by the user. PC WRITE's manual is some 60 pages and very through. You can print a quick version or if you just want to browse then you can view it on screen.

Both are fully functioned wordprocessors with such features as you get on the commercial packages, the differences are usually in the way the systems operate and the need to learn a bit more before being able to get the most out of them.

PC WRITE has two separate programs, one to enter and edit text and the other to print it. CHIWRITER has a single program which does both.

I found CHIWRITER easier to use the first time than PCWRITE but that may be because I rarely read the instructions before trying to run the program. NB I always make a copy of the program to do this in case I blow it, but it is for me a test of the program that one can load it and start without the instructions. The first thing that I found with PCWRITE was that I had to specify the file I wished to edit at the time of loading eg. to load type ED <filename>. If you dont specify the file the program hangs. There probably is another way in but I have not yet found it.

PC WRITE provides a menu along the top of the screen which is "coloured" a light grey? background to reduce the contrast between the text and the black of the screen.

Help is available on screen at all times in both programs, if anything PCWRITE is better in this area and very similar to Wordstar.

In most case the text can be manipulated with single key strokes and PCWRITE has provision for programing keys to give multiple functions. This is a feature which I dont care for in most cases as it tends to devistate your operations when withy out thinking you hit the right key for something only to find you ha changed its effect. I have enough proble learning the keystrokes for the standard program.

Both packages offer some fonts, mainly elite, with italics, sub and supperscripts (if your printer will print them, mine wont), and CHIWRITER has some maths sybols etc. Enough for most of us to get all we want down on paper.

Both offer footnotes, formatting, search and replace,etc. and all the other fancy bits we have come to expect in wps these days.

I wont tell you any more I want somebody else to take up the club offer to get a free replacment for their disk by writting a review of the programs.

Send $6.00 to the librarian for you copy of either disk. Ted Stuckey.

REVIEWERS WANTED

Help the editor of CUVIC and other club members by writting up a club disk, and get another disk free.

The club will give the author of a review of any club library disk a free copy of any disk that they want, from either the COMPUCOLOR or IBM library.

To recieve this offer you must submit comments on all the programs on the disk, and report on your experiences when trying to use them.

MORE COMMENTS FROM TED ?

Well nobody else has contributed very much for the newsletter so if is going to have anything in it other than the renewal notice its got to be me.

I have had a look at some of the IBM Library disks an offer a few brief comments.

ESIE the Expert Systems Inference Engine; And well you may ask what is that? Its about Expert systems, Knowledge Bases, Inference generators, and other artificial intellligence programs.

Every computer is used to maniplulate information either by rapid computations or by amassing, searching, sorting and printing out data. Well this program written by LIGHTWAVE CONSULTANTS is a logical attemp to get your computer to advise you on what it is that your are describing to it.

ESIE loads a knowledge base and a set of rules and builds inferences out of them. The program comes with three Knowledge bases, ANIMALS, GLASS,and DOCTOR, which allow you to play with it before trying to do anything yourself.

It also comes complete with some 60 pages of documentation on the disk , including a brief history and tutorial on Artificial Intellegence, plus two manuals.

Expert Systems Inference Engines - Knowledge Engineers which discusses Knowledge engineering or the process of decision making. Its the stuff that "Starwars" will have to have if they are ever going to work.

And a USER Manual, which tells you how to use the program to set up a Knowledge base of your own with which to make decisions.

As a test I had a play with ANIMALS, I thought of a Dolphin and answered the qustions put by the program with the details applicable to that animal. Suprise! Suprise! it got it right first go.

Off hand I cant think what I or most of us would need an Expert system generator for, however if you are the sole source of knowledge on a rare plant or objects you may care to write up you knowledge in this form for others to use.

What ever happens its good fun and a great insight into the whole art of logic programming.

PS. It took about thirty minutes to print all the manuals on my MX80, but you can print the USER Manual only if you wish.

GENEALOGY - ON DISPLAY. VER 3.1

This program has recieved very high praise from the genealogical groups and was developed by one of there members in the US.

In a test / comparison of the Public Domain programs of this type it was rated as better and easier to use than the DBASE II programs.

As yet I have not got into the program to run it but have setup a disk ready for use.

The first thing I found was that you have to load BASIC with the command /s:256. This I gather reservers enough space for the programs to run in.

You then LOAD - Menu, and up it comes to tell you about sending money to Melvin Duke. only US$35 under the Sharware system.. The next screen has the useful info on what you have to do to get started.

Choices are , Create; Personal, Marriage, or Ordanace files, Update these, Create index's, Print list of relatives by alphabetic order, print family groups, and pedigree charts (Woof Woof) and the vital choice of Quit.

A HUG IS AS GOOD AS A SQUEEZE.

It may be worth mentioning this at this stage that there are quite a few programs on the Public Domain disks that are "squeezed", that is the group of programs which make up the application have been compressed into a single .LBR. program.

This .LBR program must be "unsqueezed" in order to restore the group to an operating application. You cant even look at the program with normal DOS tools until this has been done.

All the programs you need to do this are included on the disk. You will need a blank formatted disk to down load onto.

NOTES NEWS AND OTHER RUBBISH.

MEMBERSHIP SECRETARY REQUIRED?

I have decided to take the plunge and buy an Amiga, with the result that the old CCII is being past on to my nephews to play games and hopefully learn something about computers. This means that from about Christmas (that is significant) I will not have a CCII to print the membership lables and lists or the CCII library listings.

Volunteers should que up on the right?

COMPUCOLOR PRICES OF OLD.

While cleaning up some of my CCII junk I came across a price list dated 4/10/79 for the CCII. In those days a Model 5 , 32k machine, cost $2395, the second drive cost $526, and the delux keyboard $280. Disks, blank formatted were listed at $9.95 for two, and the Maintenace Manual at $65.

Software Utilities were all about $43, with games about $25. Think about how much the club library must be worth to members on that basis.

BASIC COMPILER.

members should be aware that there is a very good BASIC compiler available for the CCII.

This program, written by Peter Hiner of the UK users (I dont know if they still exist) is a two pass program which checks your BASIC program for problems first ,then creates an .ASM file which is then compiled.

For those of us who dont understand the Assembler language it provides a method of producing a machine language program. The only problems are those associated with size. That is, what you can fit on the disk controls the size of the programs you can compile.

It has the effect of making the program run about 10 times faster, and trebling its size. Good fun to mess around with if you have a few short programs that you can call.

PS. The programs respond like BASIC to breaks etc.

Well thats about it for this issue, if you want to keep getting CUVIC you will have to start making a contibution to material for it.

Even questions that others can answer are better than nothing at all!

IMB LIBRARY NOTES

Thanks to Ron McKenny we now have access to the current, (No's 93 to 200) PC-BLUE library. The club will not initially, be keeping the complete library on hand but only those selected disks as being of major interest to most members. We will however get any of the others for you on your request. We have obtained some additional disks in the IBM Library. They are marked with a "X". The library is now as follows

PCBLUE LIBRARY

102X GAMES - 3D DEMON, CATCH THE BABY, CASTLE ADVENTURE, JUMPING JANITOR JOE.

116X DBASE II PROGRAMS, - GENEALOGY, CHURCH MANAGEMENT, CHECKBOOK MANAGEMENT, & DITRIBUTION.

121X A GENERAL LEDGER ACCOUNTING PACKAGE, FILE UTIL VER 1.7, UNDELETE A FILE, SORT DIR., BI-DIRECTIONAL SCROLLING, SCREEN PAGE DUMP, DISKETTE JACKET LISTINGS, DIS NO,SIZE BACKUP.

125 3 BY 5 - INFORMATION MANAGEMENT SYSTEM.

130 PC-WRITE VER 2.6,A WORDPROCESSOR.

133 DESKMATE VER 1.0, & PARTNER VER 1.1.

146 XLISP VER 1.5 - A LANGUAGE.

153X BASIC GAMES - AARDVARK/ ABM/ BOWLING/ BRICKS/ BUGS/ CHSONG/ FXLABLE/ KENO/ KIDNAP/ LABLEMAKER/ LEASE/ MEMO/ MONOPOLY/ QUBERT/ SPACWAR/ STARLANE/ STOCKS/ STRESS/ TEM-INS/ TEMPLE.

160X GENEALOGY ON DISPLAY - RECORD YOUR RELATIVES. (NEEDS BASIC)

164X PRESENT - SLIDE PRESENTATION SYS.

174X ESIE - EXPERT SYSTEM INFERENCE ENGINE VER 1.1, CRCBTOOL / CRC VER 6.5 - CREATES THE PCBLUE , -CATALOG. NUM AND CHECKSUM TABLES.

190X COLOR GAMES - STRIKER - A HELIOCPTER ARCADE GAME, KONG - YES THAT ONE!, GOLF - SELECT PLAY FROM GREG NORMAN TO ME, MONOPOLY - USES US PLACE NAMES.

CUVIC MEMBERSHIP RENEWAL FOR 1987

SUBSCRIPTION RATES FOR JANUARY TO DECEMBER 1987 ARE AUSTRALIAN AND OVERSEAS A$20 OR A$18.00 IF RECEIVED PRIOR TO THE 31st JANUARY 1987. (OVERSEAS MEMBERS BY 28th FEB)

NAME. _____ TELEPHONE - _____

ADDRESS _____

STATE OR COUNTRY _____ POSTCODE _____

COMPUTER MAKE AND MODEL _____

MEMORY _____ DRIVES _____ MODEM _____

OTHER BITS SUCH AS PARALLEL PORT. _____

PRINTER MAKE AND MODEL _____

---

1. WHAT DO YOU WANT FROM THE CLUB? _____

_____

2. HOW OFTEN SHOULD THE CLUB MEET? *MONTHLY* ☐ *BIMONTHLY* ☐

3. HOW OFTEN SHOULD CUVIC BE PUBLISHED? *MONTHLY* ☐ *BIMONTHLY* ☐

4. WHAT SORT OF ACTIVITIES DO YOU WANT? _____

_____

5. SHOULD THE CLUB INCORPORATE? *YES* ☐ *NO* ☐

6. LIST THE SORT OF PROGRAMS YOU WOULD LIKE TO SEE IN THE CLUB LIBRARY. SPECIFY IF THEY ARE COMPUCOLOR OR IBM.

HONARY MEMBERS NOTE

WOULD ALL HONARY MEMBERS PLEASE FILL IN POSTAL DETAILS AND RETURN IF THEY WISH TO CONTINUE RECIEVING COPIES OF CUVIC. A NON REPLY WILL BE TAKEN AS A NEGATIVE RETURN.

SEND TO - CUVIC BOX 420 , CAMBERWELL, VICTORIA , 3124. AUSTRALIA.

20 IBM PC

PUBLIC DOMAIN DISKS

MEMBERSHIP RENEWAL FORM ON BACK OF COVER

CUVIC

Registered by AUSTRALIA POST
Publication No, VBH5086

If UNDELIVERABLE RETURN TO

CUVIC
P O Box 420
CAMBERWELL VIC 3124

Postage Paid

CANTERBURY
VICTORIA
3126

CUVIC COMPUTER USERS OF VICTORIA - BOX 420 CAMBERWELL - 3126
REGISTERED BY AUSTRALIAN POST - PUBLICATION No VBH 5086

CATERING FOR COMPUCOLOR II & IBM USERS

MEETINGS HELD ON THE FIRST WEDNESDAY OF EACH MONTH IN THE SHOP AT 8:00 P.M.
SURREY HILLS NEIGHBORHOOD CENTRE
157 UNION ROAD, SURREY HILLS, 3127
(MELWAYS MAP 46 : H-10 NEAR THE STATION)

A LIBRARY OF PUBLIC DOMAIN SOFTWARE IS AVAILABLE FOR BOTH THE COMPUCOLOR II AND IBM COMPATIBLES.
CONTACT THE LIBRARIANS FOR DETAILS.

PRESIDENT : - TED STUCKEY
TELEPHONE 03 836 8732

SECRETARY / TREASURER : -
HOWARD RICE
TELEPHONE 03 277 2957

EDITOR : - LES PADGET
TELEPHONE 03 569 8774

LIBRARIANS : -
IBM - KEN SMITH
TELEPHONE 03 749 55 79

CC II - MICHAEL VERPLAK
TELEPHONE 03 831 0000

INCORPORATION

The last meeting was a traumatic occasion when the office of President was refused by nearly everyone with well-rehearsed speeches. Ted accepted the position on condition that someone takeover his spot as Acting Editor.

As your new Editor, I've been asked to write something to get my hand in on the new format and probably to see if I'm worth it.

I've been a proponent of Incorporation for this Club since another Club I know of went through it 2 years ago and have offered to help with paperwork already done when the Club is ready.

It may seem like an expensive exercise when the need seems so remote but I think you will have seen in the press stories about Clubs of youngsters that suffered because someone was seriously injured accidentally, sued and were awarded $500 000 or so. The Executive of that Club are responsible for finding the money even if they have to it pay out of their own pockets.

The Club that I know had a Public Liability insurance policy for $500 000 which cost $135 and each year it was increased until they were told that they would need a $2 000 000 cover at over $350 because of the size of awards made in court. These two facts are frightening when you consider the original idea of the Club was to have a good time in each others company and to play their sport or what ever they wanted to do.

The reason behind all this is the tenet of Law which says that natural people are the only ones that can be liable under the law. To overcome this, provision has been made so that a Club can be given legal status which if sued is only liable for the money and assets it holds. No individual is liable. Existing legislation (Company Law) has been altered to include this only 2 - 3 years ago.

The State Government produces a model Statement of Purpose and Statement of Rules, the basic idea of which is to be adhered to and which replaces the Constitution. It is nothing frightening except that it's a bit long to read but the contents lay out in detail everything that is right for both the member and the Club as a whole. The idea is to write into it what was in the Constitution and any By-Laws that may exist. It is probably better either to appoint a special group or leave it up to the Executive to re-write it.

Several things need to be explained and the most important perhaps, is the need for someone to be Public Officer. He is the person who liaises with the Corporate Affairs Office and the Club and is responsible to that office for keeping the Register of each member's name, address, phone number, date of joining, date of resignation. He also has to send in the annual report after the A.G.M., including the fee which is $60.00 the first time and $15.00 thereafter. This is subject to variation. If any changes are made, they attract a fee and there are penalties laid down for late returns unless prior arrangement has been made.

The Secretary is the logical person for this but there is no reason why you cannot appoint someone else.

Eventually, (you know what Committees are like) a working document will be offered for each person to read and then vote. You end up voting in writing on four items. One is "Do you approve Joe Blow being Public Officer?" "Having read the

Statement of Purpose do you agree?" "Having read the Statement of Rules do you agree?" "Do you agree to the name of the Club including the word Incorporated or Inc?" This unsecret vote is recorded so that the results can be sent with other details to the Corporate Affairs Office, hopefully in favour.

That is it in a nutshell and if anyone has any queries please don't be afraid tolet someone know either in writing or by phone. We may as well enjoy the Club after the work is over.

Les Paget.

Help the editor of CUVIC and other club members by writing up a club disk, and get another disk free.

The club will give the author of a review of any club library disk a free copy of any disk that they want, from either the COMPUCOLOR or IBM library.

To recieve this offer you must submit comments on all the programs on the disk, and report on your experiences when trying to use them.

CRICKET IS SIMPLE

If you ever wanted - for an American friend, say - a simple explanation of cricket, here it is:

In cricket there are two sides. One is ***OUT*** ***IN*** the field and one is not ***OUT*** ***IN*** the field and that, of course, is the one that's ***IN***.

A batsman in the side that's ***IN*** has to go ***OUT*** to get ***IN*** but when he's ***IN*** and gets ***OUT*** he has to go ***IN*** and the next man ***IN*** goes ***OUT*** and stays ***IN*** until he's ***OUT***.

But he could be ***NOT OUT***, because at the endof the ***IN***nings there's always one man ***IN*** who is "***NOT OUT***". However, he still has to go ***OUT*** and as soon as he goes ***OUT***he has to come back ***OUT***.

So, when all the batsmen who've been ***IN*** are ***OUT*** they go ***OUT*** and try to get ***OUT*** the other batsmen as they come ***IN***.

When they've all been ***IN***, ***OUT*** or ***NOT OUT*** a couple of times the game's over and they often go ***OUT*** to the ***INN***.

MINUTES OF THE ANNUAL GENERAL MEETING - 1 ST APRIL 1987

PRESENT - H.Rice, K.Winder, K.Smith, C.Scown, G.Newson, R.Thompsom, M.Verplak, M.Verplak Jrn, N. Brandie, D.Hill, L.Paget, A.Lewis, D.Haskin.

APOLOGIES - A.Kirkpatrick, R.McKenny, B.Muldowney.

Minutes of last Annual Meeting taken as read. Moved - Stuckey/Verplak. Carried.

ELECTION OF OFFICE BEARERS.

PRESIDENT -E.STUCKEY,- Moved Paget/Haskin, Carried.

SECRETARY/TREASURER - H.RICE,- Moved Stuckey/Winder, Carried.

EDITOR - L PAGET,- Moved Stuckey/Brandie, Carried.

LIBRARIAN IBM - K.SMITH,- Moved Muldowney/Winder, Carried.

LIBRARIAN CCII - M.VERPLAK,- Moved Stuckey/Smith, Carried.

A vote of thanks was passed to the retiring office bearers, in particular to the President Ken Winder, for their work in keeping the club going over the past year.

TREASURERS REPORT.

No report available for the meeting, CUVIC costing approximately $66:00 per issue, plus $20:00 postage. Membership renewals for 1987 55 to date. Bank balance - $1294.35.

GENERAL BUSINESS.

1.0 FUTURE OF CLUB. The possibility of joining with other similar clubs, such as MICOM or the PC USERS was discussed. It was decided that the club should remain as separate unit for the time being.

2.0 INCORPORATION. Motion - That meeting discuss the need for incorporation by CUVIC. Moved Paget/Stuckey.

Following discussion it was decided to proceed with the preparation of a new draft constitution suitable for incorporation, and take the necessary steps to obtain the membership approval to proceed with incorporation.

MEETING CLOSED AT 2130 HRS. TED STUCKEY.

SUPPORT YOUR CLUB
SEND NEWS NOTES
AND COMMENTS

JET - THE FAST FLYING COMPUTER SIMULATION

Those of you can remember back a few years will also remember my interest in flight simulations. Well I never found a good simulation for the old CCII but there are a number available for the PC.

I have recently been trying my hand at flying an F16 (land based fighter /bomber) and the F18 (carrier based fighter /bomber) as simulated by SubLOGIC in JET.

Jet can be loaded from DOS by typing <JET>, if you have a Hercules type <JET H>, if you have EGA type <JET E> for colour or <JET M> for monochrome.

On loading you will be confronted with a number of choices as to type of monitor, joystick or keyboard, aircraft, type of action etc. you then select your weapons with the number keys and finish with the <5>.

Having made your choices you find yourself sitting in the cockpit of the aircraft looking forward. The screen display show the view, you will see the height on the right side of the screen, air speed on the leftside, with the heading at the top. Along the bottom of the screen from left to right are; frame loading in "g's", percentage thrust, afterburner on/off, active weapon, remaining ordanance, fuel, airbrake on/off, landing gear up/down, and zoom factor.

The controls are Throttle <+> increase <-> decrease, Radar <W>, range circle <H>, sound on/off <S>, attitude indicator <A>, gear up/down <G>, air brakes <B>, pause <P>, control tower view <C>, zoom view out <PG DN>, zoom view in <PG UP>, select views <SCROOL LOCK> then ARROW, up is <5>, select weapon <RETURN>, fire weapon <SPACE>, next target <BACKSPACE>, and last but not least directional control is by cursor keys ,pith up <8>, pitch down <2> etc, stop pitch/bank <5>. to launch from the carrier hit <L> after you open the throttle, one other key you should know about is <SHIFT><E> to eject, <ESC> to return to menu.

You open the throttle and taxi out on to the runway, its difficult to stop as the brakes do not have much effect on the ground. Throttle up , engage afterburner, and your away down the track, you need about 120 knots to lift off, hit pitch up <8> , then <5> to hold the controls in that position, keep climbing, until you have some height. gear up, afterburner off. Toggle on the Radar and check for enemy fighters, and targets.My advice is to start with the target missions, they are easier to learn on. When attacking the enemy remember that the range ring goes from white to black when you are within the weapons range.

Whats it like? Well I found it difficult to see some of the details on the monochrome monitor but it's OK on a color monitor. Good simulation of flight and controls with very difficult landings back on the carrier. Things happen very fast at times, but it is not like an arcade game where the flying finger always beats the calculated attack. The fully aerobatic preformance of the fighters makes it fun to fly. I haven't seen it available in many places but it is always on Imagineering sales list at $110.

Ted Stuckey.

Optical disks.

==========

Recent news gives Thomson, OSI, Philips, DuPont, and Sony Corporation as having joined together to produce basic specifications for a family of 5 1/4" optical disks, interchangeability guaranteed. In cartridge form and using a sampled servo format adaptable to a broad range of optical media. An expected capacity of 300 megabytes each side of the cartridge is forseen. Erasable disks, as well as the usual read-only and write once should become available eventually. Reading is by a laser beam as are CDs and Video disks now, for writing, the laser power is increased to "burn" in the data. The expected price of these optical units is as unknown as ever.
K.G.W.

Optical disks.
==========

Recent news gives Thomson, OSI, Philips, DuPont, and Sony Corporation as having joined together to produce basic specifications for a family of 5 1/4" optical disks, interchangeability guaranteed. In cartridge form and using a sampled servo format adaptable to a broad range of optical media. An expected capacity of 300 megabytes each side of the cartridge is forseen. Erasable disks, as well as the usual read-only and write once should become available eventually. Reading is by a laser beam as are CDs and Video disks now, for writing, the laser power is increased to "burn" in the data. The expected price of these optical units is as unknown as ever.
K.G.W.

THE EDITOR AWAITS YOUR ARTICLES FOR CUVIC, SEND NOW!

LETTERS FROM OVERSEAS.

FROM PETER HINER, SECRETARY -UK. USERS GROUP, ENGLAND.

I am not sure if its because of my position as Secretary of the UK Users group that I have been receiving a copy of CUVIC. I can only say that its arrival is always most welcome. I enjoy reading the latest news on the Compucolor and am very pleased that somebody is keeping the torch burning.

At the same time I feel a bit guilty at receiving CUVIC as the torch went out in the UK a couple of years ago for much the same reasons that other groups declined : people drifting away to new computers. In the case of CUVIC you started off as a much stronger group than we were and have adapted to the changed environment.

All I can say is thank you for sending CUVIC and to donate the last version of FASBAS and ZIP to your library. I have enclosed a disk with all the known bugs removed ie., FASBAS Version 12.25 and ZIP Version 2.0 plus a copy of the manuals. I hope someone out there is still interested in using them.

My CCII is still used by the kids for playing games, but I have not written anything for ages. I have a new computer, The ENTERPRISE which you will not have heard of, as it was not sold outside the UK. its a Z80 based machine in which everything is soft, colours range from 2 to 256. The company's gone broke. There must be something in common between the machines that caused me to buy two computers that failed in the market place.

As to the future, I have to say that much as I enjoy receiving CUVIC I do not feel that I can continue to accept the Honorary membership any longer. So I will just say "thank you" and wish you a long and prosperous future..

PETER HINER.

NOTE - We are very pleased to receive these two excellent programs for inclusion in the library. And want to thank Peter for his generous donation. He will continue to receive CUVIC for this year as we owe him A$30+ for the sale the FASBAS program some time ago. We will publish the main parts of the manuals in CUVIC in the near future. Ted Stuckey.

FROM WILLIAM PARKER - FORUM LIBRARY, USA.

In reply to your request for an updated library listing. Unfortunately there has been no new programs added to library and only a half a dozen or so requests for disks since I took over the library.

In the FEB CUVIC you published a 1984 price list of program from Intelligence Computer Systems of Huntsville Alabama. Unless they have come back to the USA, these are no longer available. The owner moved back to Germany a couple of years ago.

Some of your listings are also incomplete. When I inherited the Forum library it included the NCC (NORTH CALIFORNIAN) disks through to No 34. The listings are different to the ones you have. Many of the same programs but on different disks numbers and in different groups.

The CHIP (ROCHESTER GROUP NEW YORK) now goes to No 143, maybe more as I haven't received a newsletter for several months.

NOTE - Bill's letter raise the interesting question as to if the ICS software has been released into the Public Domain? If any other overseas groups have information on this matter would they please let us know. Ted Stuckey. PS. Bill, We do have Forum No 15B to 18B they just didn't print on the library list.

FOR THOSE WHO DONT KNOW FASBAS & ZIP.

FASBAS

FASBAS is a two pass compiler for CCII BASIC programs which will make the program run about five times faster although the speed will vary according to the contents of the original program. It comes with a very comprehensive manual and is easy to use. We will publish the manual in CUVIC over the next few months.

FASBAS produces a PR6 program which is about twice a long as the original.

ZIP

ZIP is also a BASIC compiler but is designed to make better use of those programs which only use Integer values, and therefore can be more neatly and compactly compiled.
It also produces a PR6 file that can be run from the FCS. Again Peter Hiner has produced a very comprehensive manual for the user, which we will also publish.

Ted Stuckey.

WE NEED YOUR HELP TO KEEP THIS NEWSLETTER GOING, SEND YOUR LETERS, NOTES NOW!

HANDY CALLABLE ROM ROUTINES FOR USE IN ASSEMBLER PROGRAMS.

NAME	6/78	8/79	REF.	DESCRIPTION. . .
RXSER	0020	0020	1	Reads a character from the serial line
KEYTST	0024	0024	1	Returns with the Z bit if there is a character in KBCHA(81FE)
TXSER	0028	0028	1	Transmits a character to the serial line
BASOUT	0033	0033	1	Sends one byte to the screen from <A>
FCS	25EC	0A95	6	FCS command interpreter. Command line addressed by <HL>
FCSEM	262A	0AD3	6	FCS plus EMESS
EMESS	262D	0AD6	6	FCS Error message interpreter
RESET	26A5	0B48	1	Reset disk if error found
CKEND	26E7	0B8A	1	Check for end of file
GETTO	2C0C	1042	1	Check for string 'TO' in Command line pointed to by <HL>
OPENX	2C86	10BC	1	? ? ? ? ? ? ? ? ? ? ? ? ? ? ? ?
OPENY	2C89	10BF	1	? ? ? ? ? ? ? ? ? ? ? ? ? ? ?
OPDIR	2D60	1196	1	Open Directory for a scan
GNDE	2D86	11BC	1	Get next directory entry
OPEN	2DAB	11E1	1	Open a File for input or output
READ	2EA3	12D9	1	Read from an open File
WRITE	2ECC	1302	1	Write to an open File
WR	2EF8	132E	1	? ? ? ? ? ? ? ?
RD	2EFB	1331	1	? ? ? ? ? ? ? ?
CLOSE	2F26	135C	1	File Close routine
PDV	2FDE	1414	1	Parse Device Name
PFSPC	3077	14AD	1	Parse File Spec.
RWSEQI	30C6	14FC	1	Rewind 'Sequential Input' Routine
INSEQO	30E7	151D	1	Initialize 'Sequential Output' Routine
CLSEQO	3136	156C	1	Close 'Sequential Output' Routine
WBLK	317F	15B5	1	Read Block routine
RBLK	3182	15B8	1	Write Block routine
WBLKI	31F6	162C	1	Write Block & Increment routine
RBLKI	31F9	162F	1	Read Block & Increment routine
GTBYT	322C	1662	1	Read Sequential bytes from an open file
PTBYT	324A	1680	1	Write Sequential bytes to an open file
GAREC	3257	168D	1	Get ASCII record
PVREC	327B	16B1	1	Put Variable length record
PTREC	3285	16BB	1	Put Unformatted record
CRLF	338B	17C1	5	Send <CR> & <LF> to output device using <LO>
OS (LO)	3392	17C8	2	Send a character to output device
LBYT	339B	17D1	5	Takes the byte value in <A> and lists it as two Hex. characters to the output device
B2HEX	33AA	17E0	5	Converts lower nibble in <A> to Hex. digit
OSTR	33F4	182A	1, 2	Sends a string, terminated by 239 (EF) and pointed to by <HL>, to output device
WATS	341C	1852	5	Wait routine of 0.5 msec * <A>
WATL	3429	185F	5	Wait routine of 20 msec * <A>
MOVDH	343B	1871	1, 5	Move <B> bytes from <HL> to <DE>
MOVHD	3444	187A	1, 5	Move <B> bytes from <DE> to <HL>
CMPHD	344D	1883	1, 5	Compare for <HL> (<=>) <DE>, returns set (c, z, 0)
CMPDH	3453	1889	1, 5	Ditto swapping <DE> with <HL>
SUBHD	3459	188F	1, 5	<HL> minus <DE> , result in <HL>
DIG	3476	18AC	4	Scan for a digit, C set if yes
ADHLA	3518	194E	1, 7	<HL> plus <A> in <HL>
ANHD	351D	1953	1, 7	<HL> AND <DE>, placed in <HL>, <DE> preserved
NEGH	3524	195A	1, 7	Twos complement of <HL> placed in <HL>
NOTH	3525	195B	7	Ones complement of <HL> placed in <HL>
ORHD	352C	1962	7	<HL> OR <DE>, placed in <HL>, <DE> preserved
XORHD	3533	1969	7	<HL> XOR <DE>, placed in <HL>, <DE> preserved
SHLHD	353A	1970	7	Shift <DE> left <HL> times with zero fill, result in <HL>, <DE> lost, <BC> preserved
SHRHD	3544	197A	7	Ditto but Right shift
MULHD	3562	1998	1, 7	<DE>*<HL>, placed in <HL>, <BC> preserved
DIVHD	3581	19B7	1, 7	<DE>/<HL>, dividend in <HL>, remainder in <DE>, <BC> preserved
SAVE	3FD0	0A2E	4	Save all registers

SEE YOUR NAME IN PRINT WRITE NOW!

CUVIC

Registered by AUSTRALIA POST
Publication No, VBH5086

If UNDELIVERABLE RETURN TO

CUVIC
P O Box 420
CAMBERWELL VIC 3124

Postage Paid

CANTERBURY
VICTORIA
3126

CUVIC COMPUTER USERS OF VICTORIA - BOX 420 CAMBERWELL - 3126
REGISTERED BY AUSTRALIA POST - PUBLICATION No. VBH 5086. CATEGORY B.

PRESIDENT - TED STUCKEY,

PH. 03 835 8732.

SECRETARY - HOWARD RICE,
TREASURER PHONE 03 277 2957.

EDITOR - LES PAGET,
PHONE 03 569 8774.

APRIL 1987

PRICE $ 1.50 MONTHLY

NEXT MEETING - WEDNESDAY 6TH MAY 1987.
IN THE SHOP AT THE
SURREY HILLS NEIGHBOURHOOD CENTRE,
157 UNION ROAD, SURREY HILLS, VIC. 3127.
(MELWAYS MAP 46 H 10. NEAR THE STATION.
COMMENCING - 8:00 pm.