†
EAR 99'ERS
East Anglia Region 99'ers User's Group
VOLUME 2 - ISSUE 6 - OCTOBER '88
† † † † † † † † † † † † † † † † † † †
Mail - Great Britain: Mail - United States:
Scott Copeland EAR 99'ers User's Group 13 Elm Walk Lakenheath, Suffolk England 1979CS/XP FCS Box 5927 England 1927 9GR Tel: 063881-3457 FOR Tel: 011-44-63881-3457
President † Scott Copeland † 063881-3457 or 011-44-63881-3457 Vice-Fresident † Robert Wordsworth † 0603-38832 Secretary/Treasurer † Jo Ann Copeland † 063881-3457 Publication's Officer † Bryan Cloud † 0473-464996
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
Contents: 11 Pg 2 ft Minutes Of The Meeting 11 Pg 5 tt For Dur Newsletter Exchanges 11 Pg 6 ft TI Shows Coming Up! 11 Pg 7 ft Notes, Webs & Library Additions 11 Pg 10 ft Mini-Memory Part XI f by Robert Wordsworth ft Pg 12 tt Play The Game Of Life f by John Stocks ft Pg 14 ft For Big & Little Kids f Music Maestro! ft Pg 15 ft Disk Peripheral Software Overview f by Colin Hinson ft Pg 23 ft Treasury Report 11 ENC ft Members Questionnaire Form
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
This newsletter is not copyrighted and may be reprinted in whole or in part. PLEASE give credit where credit is que, and notate the Club and the Author. Any user assumes sole liability for use and please note any information by authors' does not guarantee it to be error-free.
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
Now, turn the page for an EAR-ful of information
1

East Anglia Region 99'ers User's Group Page 1

MINUTES of the Meeting 3 September 1988 by JoAnn Copeland

■ Minutes of the Meeting ■
■ 3 September 1988 ■
Submitted by: Jo Ann Copeland, Sec/Treas

(This was over a two hour meeting, so those who either receive this newsletter as an exchange or are otherwise disinterested (members excluded from this category!) can feel free to turn the pages to get into the newsletter...)

The meeting was called to order at 15:02 PM with 10 present. There was a 100% quorum on Officers, and less than the 80% quorum required for voting by members. Thus, members (only) will find a questionnaire enclosed which requires completion and return by no later than 15 OCTOBER 1988.

The meeting was adjourned at 17:07 PM. The following points were covered in the meeting, and require voting on:

 Discussion was made on the following options: a) Keep the group and the newsletter, b) keep the group with no newsletter, c) discontinue the group and newsletter.

2) Changing back to the original 8 1/2 % 11 format for the

newsletter, or keeping the A5 format. 3) After we move, where would be b

- 3) After we move, where would be hold the meetings? One choice is the standard meeting at the Copeland's (Woodbridge, Ipswich area) or "floating" meetings. We have had volunteers who have opened their home up for a meeting.
- A) If we want a good neweletter (or better) we will need input FROM MEMBERS. We are not the best, but there are those we receive exchanges on who produce only 2 pages. (But they're a group and they're trying!) Every one agreed we do not want to turn into a 'newsletter reproducing from other newsletters' as we all take turns reading the exchanges. Richard Tywning volunteered to handle Extended Basic. We have regular columns from Robert Wordsworth, Colin Hinson, Neil Wilson has taken over reviews, Derek Allen provides our 'cartoon corner', Tony Bowden handles copying needs outside of the newsletter, and although we have had allot of contributions from members (and we appreciate it, don't get me wrong!) we still need REGULAK submissions. This newsletter is a good example of what can be done with articles by members. (Fuller than last month eh'?)
- 5) We want to start a monthly 'meet your fellow members' column whereby each month a member will write a little bit about themself (so the next time someone asks Jo where so-and-so works or what so-and-so does, and she doesn't know, they can read it in the newsletter!). DEREK ALLEN is member # 1 and we'll ask him to be the first (and break the rest of us in!).
- 6) For every new subscriber we GET we LOSE one! Apparently news of the group IS spreading but we're apparently not meeting members needs or they would keep subscribing. Yes, we have had renewals, but we've also lost 8 over the last year, with 8 new subscriptions. ## What do you want to see in the newsletter and ## how can we get more across to members to keep you interested?
- 7) A Question and Answer column will be started. When a member calls us with a question we'll answer it then and there, however we'd like to include the question in the newsletter for those of us who might have the same question (and were too shy to ask!).
- 8) Keep reproduction for the newsletter on the Base or go off-base for reproduction services. See more...

9) Summer came and things dropped off (understandable), but we're in need financially of building up once again and Library Orders aren't what they used to be. Are we getting what you want (when we can)? What do you want to see in the Library? Are the explanations of the software enough to get you interested enough to get a copy? How much of an explanation would you like to see in our next Catalog? (Detailed, Brief, Extremely Detailed, etc.?)

The reasons for the questions about keeping the group (and newsletter, etc.) are due to the Copeland's moving in November. Currently we are affiliated with the Base MMR office (Morale, Welfare and Recreation Office). We have to abide not only by British Laws but also by the Base rules and regulations. When we wanted to have the Raffle for the group as a fundraiser the British Laws said OK but the Base said No. We had to (in the past) produce a monthly treasury report to submit to the base, but now they want a photo-copy of our monthly bank statement. Reproduction is CHEAF but we all know the problems of getting the newsletter reproduced within the 5 day turn-a-round promised. When we move to RAF Bentwaters. It will take approximately 2 to 3 months to get approval for our group through that base. We can not keep reproduction at RAF Lakenheath with someone else picking up the copy as the officers have to be affiliated through the base they are working from. Thus, RAF Bentwaters. So, for Question #8, do we want to affiliate ourselves through RAF Bentwaters or just go off-base for reproduction services, with a possible rise in cost for reproduction?

Guestion # 1: Members at the meeting said they wanted to keep the group \underline{and} the newsletter, along with the monthly meetings. The reason the question was approached was the lack of submissions by members to produce a nice sized newsletter. We \underline{need} regular input from members.

Question # 2: What format do we want? The A5 format is <u>cheaper</u> to reproduce and we can put in twice as much content for the same price. However members are finding it difficult to read. 8 1/2 X 11 reads easier, but costs more for the size of the newsletter. Your choice?

Question # 3: Should we continue to meet monthly at the Copeland's in the Woodbridge, Ipswich area or go to 'floating meetings' at members homes? If floating, would you volunteer your home for a meeting?

Guestion # 4: Can you offer a regular monthly article? This doesn't have to be a technical drawn-out boffin column! Is there any subject you find regularly interesting? Did you come across something interesting lately you'd like to tell others about? Anything can be submitted, but we'd like this newsletter to be a MEMBERS newsletter.

Question # 3: Would you submit a small column about yourself for the 'meet your fellow members' column? We'd like to start it monthly, going from member # 1 all the way through to member # 38 (or on if we have new subscribers). We'd also like to include those who ticked the 'Contact Box' as No on the Questionnaire (but will respect your wishes if you choose not to include your name in this column!). You would be contacted a month prior to needing it and submission can be on paper, disk, or anything handy. We also need a NAME for the COLUMN! Please think of something and list it on the questionnaire.

Guestion # 6: What do you want to see in the newsletter and how can we get more across to members to Leep you interested? As stated, for every new member we get we're losing one. Any ideas?

Guestion # 7: The Question and Answer Column - if we need assistance can we call you for an answer? The subscription forms will let us know whether you have an interest in the area questioned or not.

Question # 8: See above paragraphs pertaining to this please.

Guestion # 9: Is the Library meeting what you're looking for? Do you want a more detailed explanation of software coming in, or do we not have what you want? What do you want to see us get in? Do you have anything you'd like to INCLUDE in the Library?

Question # 10: Any suggestions?

OK, moving isn't going to cause too much of a problem (being in the military you get used to it!). There might be a slight problem in getting one or two issues out exactly on time (not breaking any records in that area anyway). This will depend on whether we have a place to go to before we move, or in case we don't, our computer will be packed until we do find a place to move into. We may not even have a forwarding address (except for the box number on the base!). So part of the group meeting discussion was about this, and members agreed and understood about a possible delay in newsletter production. Jo Ann feels members aren't getting anything out of the newsletter (for every new member we get we lose one) so we tried to think of ways of keeping every one interested. The newsletter size was discussed and the choice is up to you. Floating meetings were OK'd, but again it's your choice. And once again, we need regular input from members. Remember, it's YOUR group and YOUR choice is important! So please consider everything listed and circle the appropriate YES or NO choice on the questionnaire. Please return it no later than the third week in October so we can show the results in the next newsletter. An addressed (sorry, not stamped) envelope is enclosed for your convenience in getting this back to us. THANKS!

lat's make 1988-1989 a bigger and better year for the group and newsletter! Become active and participate — it helps others and it helps yourself!

BECOME A "BACK BONE" and not a "WISH BONE"!

Which bone are you? It is said the membership of any organization is made up of four bones. There are "Wish Bones" who spend all their time wishing someone else would do all the work. There are "Jaw Bones" who do all the talking and very little else. Third are the "Finuckle Bones" who knock everything that anybody else tries to do. Finally there are the "Back Bones" who get under the load and do the work. Which bone are you?

Remember, if you submit an article (on disk or cassette, etc.) you get something from the Library in return! All you have to do is let us know what you want!

→ NEWSLETTER EXCHANGES →

There seems to be a bit of confusion which we can hopefully take care of with a simple (") explanation... EAR 99'ers User's Group does not have two (2) offices. EAR 99'ers is physically located in the UK - Great Britain, England (East Anglia Region, thus the name EAR 99'ers). Scott (Fresident) is associated with the United States Military Services (Air Force) and is currently stationed here on a four year tour (recently extended for another three years). Mail from the United States can be sent directly to England but would cost twice as much to do so. To save expenses we list the APO New York address. APO meaning something to the extent of American Post Overseas. There is also an FPO address for civilians overseas, FFO meaning Foreign Post Overseas. Mail coming from the US goes directly into New York and gets put on a Plane (or slow boat to China for second and third class mail) and comes directly into a Military Base for distribution to all Military Bases Overseas. ANY mail directed to an APO address MUST HAVE a military members NAME on the FIRST line of the mailing address, no matter if it is coming to a group (such as ours) or not. Any mail NOT listing a military members name on the first line WILL BE SENT BACK. If disks are sent they require a just follow them...

If you are in ENGLAND you can use the physical address of 13 ELM WALK but just to mess things up you can also use the BASE address (but in another way!). So, when it comes down to it we can receive mail at least 3 different ways! Groups or friends in Australia, Belgie, Switzerland, etc., MUST use the England address. Canada and the US can use either, but the APO NY address is MUCH cheaper (and gets here alot faster!).

When it actually comes down to it it's as easy as having a Post Office Box Number and a Street Address. You can use either but the mail is going to the same person at the same address. I'll list the addresses you can use but the ones on the first page of this newsletter work as listed.

IF IN ENGLAND (or Australia, Belgie, Switzerland) USE

EAR 99'ers User's Group → or ← 13 Elm Walk Lakenheath Suffolk ENGLAND IP27 9GR

SSgt Donald S. Copeland RAF Lakenheath P. O. Box S927 Lakenheath, Suffolk ' England IP27 9PN

→ this arrives at our HOME

+ this goes to the base post office box

IF IN THE UNITED STATES USE

SSgt. Donald S. Copeland 1979CS/XF PCS Box 5927 APO NY 09179-5379

this gets put on a plane and arrives at a military base to be distributed to separate military bases overseas at a post office box on base.

 Remember, a military member's name MUST be on the first line of the mailing address.

Another point for the record is that Scott, our President, is actually SSgt. Donald S. Copeland (but he doesn't like the name Donald So he uses his middle name). Or should I say his friends call him Scott and others call him Don

MEMBERS TAKE NOTE

→ BLOXWICH +

Gordon Fitt has advised us that the next BLOXWICH meeting will be held on SATURDAY - OCTOBER 22, 1988 - FROM 10:00 AM to 5:00 PM. As usual the Sneyd Community School will be the place to meet at! Remember Gordon Pitt can be reached for further details via:

Gordon Pitt 259 Sneyd Layne Bloxwich, Walsall West Midlands England WS3 2LS

tel: 0922-476-373

Richard Sierakowski will be there with the Geneve AND hopefully the MYARC HARD DISK Controller (if the British postal strike gets over soon!). Gordon Pitt's BBS will be demonstrated, and you can again see the Mechatronics 80 Column Card, My-Art, My-Word, etc.! Reserve SATURDAY OCTOBER 22 on your calendar!

→ ALTERNATIVE PC SHOW ←

NOVEMBER 12, 1988 will be the ALTERNATIVE PC computer show in BIRMINGHAM. Members will get a separate Leaflet on this! We plan to attend with our set up! So mark your calendar! (SEE PAGE 13 FOR MORE...)

Remember, if you don't read your newsletter you won't know what's happening in the TI world! Gotcha' Mike!

→→ Turn the page for more ... ++





NOTES & NEWS



and Library Additions

For the Record...

NOTES - NEWS - LIBRARY ADDITIONS S

→ We welcome aboard new member NICK SUMMERS! Let's hope we can help him feel a part of our TI Family! If you're in the (Cheam) Surrey area, why not give Nick a call so he'll know there are other TI User's in his area!

ROY ROBINSON wishes to thank everyone for all the calls he got when he was looking for his Extended Basic Module and TI Contacts. He asked that we advertise the following:

Wanted: TI Contacts from anywhere in the world to exchange ideas, listings, DIY ideas, etc. Write to me at:

112 Cliff Road Hornsea, N. HUMBERSIDE England HU18 1JE

tel: (0964) 534611 after 6 PM

"Got a couple of letters from Mark and Chris (Ziegler) who have since PCS'd to the States. The plane trip went well, but also poor Willie got left at the airport for three days and now he won't get into his "cage" that he was transported in. Imagine an airline forgetting to tag the cage? Wouldn't mind it if dogs were shipped free, but they cost as much as the half-price ticket on children! And then they put the poor dog in the storage department! Gee, an idea there... Can we do that for children also? (Bet I'll get letters on that one...)

Anyway, Melissa is doing well and getting in all those baby teeth and things are going well for the family. Still haven't gotten those pictures they promised us though. Should we cancel their subscription? (I know you're reading this Chris — so get to it!)

We had a call from Mr. Course who is collecting some TI software, etc., and he'd appreciate it if anyone out there (MEMBERS WAKE UP) has a Speech Synthesizer they wouldn't mind giving up. Any one got a line on a Synthesizer ($\pm 10.00 - \pm 12.00$ area)? See Jo Ann alease...

please...

**REMEMBER % this is your space to advertise, whether you are looking for something or wanting to sell something. Just get in touch with Jo before the third week of the month and she'll get it into the newsletter! Hey, it doesn't cost anything either! (That'll turn some heads...)

Got a letter from Chris Bobbit (Asgard fame) who somehow got the idea Scott is in the Army (ugh, yuck, pew!). Anyway, we hope to hear from him more in the future as he didn't realize the UK (Britain) actually had an active user's group and he hopes to advertise in our newsletter more often.
News on the Legends front also brings us the rumor of a new program coming in the fall from the authors of Legends — maybe a sequel idea? We look forward to it!
Scott has found several bugs in the newest version (1.1) of Legends but is wading through it all on level 15 to complete the quest. We'll be putting in a listing of all the monsters (hit classes, experience, gold, etc.) in the near future.
In the August '88 (Volume 2 Issue 4) Newsletter it wasn't a typo (but a bad printing copy) on the FIX to LEGENDS 1.1. To correct this before I get alot of nasty calls, we understand from mid-south 99'ers User's Group (V6 #7) that an error can be corrected in LINE 1200 (no, I don't know which actual program) by changing the □Newell □Newell

The bad print copy came out reading Lk=F but I checked my original newsletter copy and there it was in its correct form... Anyway, good luck to those of you reaching the end of Legends (and yes, I have the back of Scott's head memorized...) I think I just heard a snicker from the peanut gallery...

- *TI BASE Version 1.2, REMIND ME, CATLIB COMPANION, a program to catalog those disks for PLATO modules, a new Infocom RAPID LOADER, LISP* (artificial intelligence for the TI) are alive and well and residing in the UK. TOWER and NORAD have arrived from Not-Polyoptics and run in Extended Basic (for those interested in flight simulators, etc.). Can be seen at The Copeland's if interested. Not-Polyoptics is as fast as Asgard Software in getting here (great service)!
- # Still don't have the FunlWriter Version 4.11 but have asked a contact in Canada to pass it to us (who's a really neat person!) and we'll have it in shortly. Those of you requesting 4.10 will be asked to wait for 4.11 to arrive as Tony McGovern advised us of the nasty bug in DM-1000 (and fixed in 4.11). Shouldn't be long now!
- * Well, our car is being repaired at this writing, our Myarc Controller Card was brought back to life (how did I ever survive without it!!), the BBS is up and running Is our run of year-long-bad-luck giving us a months break? Hope so! Everyone cross your fingers please...!
- **Richard Twyning has given alot to the Library mixed between a LOT of disks with Graphx pictures and slides on them, and some extraordinary Extended Basic programs. Some you have to see to believe! Will be listing them as soon as I organize and collate them (and crawl out from under a bunch of disks and paperwork)! Thanks for all the input RICHARD!

- LIBRARY ADDITIONS -

ADD ARO027 > Graphx Pictures and Slides home-grown by our own Richard Tywning. Includes: 0-5; 6-9+A-Q; A-I; Carbret; Computer Clipboard; Alpha Clipboard; ChessMen Clipboard; Ears; Eyes 1 and 2; Icons; J-R; Mouth; Nose; K-Z+A-Z; S-Z; Small Alpha Letters; Texas; Texas Z; Texas Clipboard. Disk (327)

ADD AR0028 > Graphx Pictures and Slides by Richard Tywning. Includes: 2_Stroke 1 and 2; 4_Stroke; Clutch 1 and 2; Col_Rings. Disk (324)

ADD AR0029 > Graphx Pictures and Slides by Richard Tywning. Includes: Commodore; Disk Brake; Drum Brake; Farm; Graph; Ignition. Disk (324)

ADD AROUGO > Graphx Pictures and Slides by Richard Tywning. Includes: Match Boxes; Modem; Patterns; Plane; Radiator; Shuttle. Disk (324)

ADD AR0031 > TI-Artist Pictures. Includes: Birthday Card 1; Birthday Card 2; DougQuin; Eye Picture 1 and 2 (color is nice!); F Face 15; Geisha. Disk (350)

ADD AR0032 > TI-Artist Pictures. Includes: Hard Porn (a good laugh here!); JMB Guote; Mad Hatter; Marlen01; Pre-Man 1; Rural; Taj Mahal. Disk (351)

ADD AR0033 > TI-Artist Pictures. Includes: Pyramid 1; Rogue; Sad Duck; Snow White; Star Trek 15; Sunset; TerraHawk. Disk (350) ADD AR0034 > TI-Artist Pictures. Includes: TI-99 Console; Tiger 02; Trident; Week End; Winny the Pooh 1; Winny the Pooh 2. Disk (300)

ADD AR0035 > Graphx Pictures and Slides by Richard Twyning. Includes: Alien; Alien 1; Car; Chess Board; Clibprevchr; Clip Shuttle; Colours; ScrnPlan. Disk (348)

ADD AR0036 > Graphs Fictures and Slides by Kichard Tywning. Includes: Condorman; CSignature; Dracular; England; Ferrari 1; Ferrari 2; V and V Clip. Disk (347)

ADD PROOO3 > Richard Tywning at it again with some excellent programs for Basic or Extended Basic run. Includes a Loader with a great demonstration of colors and sprites. Includes: Ball_Demo (demo of a soccer ball (football) bouncing on screen; and a load program to print out a great picture of Beetle Juice (from the movie). Disk (346)

ADD PROODA > Richard Tywning's 2nd half of Beetle Juice. Includes: Beetle; Beetle2; Juice0; Juice1; Juice2; and Skull. Disk (140). Richard shows what your TI is capable of!

ADD PROODS > Richard Tywning's fantastic examples of programming. Includes: Alpha: Calculator; Commando: Commando24 Commando4; End; Game; Newsletter; Print_News; Stop; The_Fly; Top; Window. Disk (351)

ADD PROOO6 > More of Richard Tywning's examples of what your II can do! Includes: Fly Text 1; Fly Text 2; Fly Text 3; Fly Text 4; and Frint_Text. Disk (284)

ADD SE0042 > CATLIB COMPANION. If you've used CatLib before to catalog your disks you'll find the CatLib Companion a necessity! DIS/VAR files help you along the way! Disk (357) XB

ADD SE0043 > LISP* and PLATO*. LISP* is Artificial Intelligence for your TI-99/4A. -ReadMe file includes instructions! You can also be correct in saying it is a Language primarily used for artificial intelligence. Includes (besides -ReadMe file) a brief list of functions available, a description of the language, and the interpreter itself (E/A Opt * 5). Version 1.0 by Charles Rentmeesters (Freeware). PLATO* can be used for those Plato Disks that require a catalog. Includes documents. Disk (205)

ADD AD0008a > INFOCOM RAPID LOADER offers you color choices with key commands; conversion to individual Infocoms and RLOAD/DOCS for instructions. Extremely GUICK Infocom loader by Ray Kazmer (Version 2.0). Disk (320)

More Graphx Pictures and Slides next month when I finish wading through four more DSDD disks and catalog them! Please feel free to call if you need any further information on what's listed this month! jc

MINI-MEMORY



PART XI by Robert Wordsworth

MINI HEMORY Part XI
by: ROBERT WORDSWORTH

As the wet-moodle treatment (see last month's issue) definitely doesn't appeal, it's back to the grindstone for me...

The previous program moved a + sign around the screen according to which of the "arrow" keys, D, S, E, X, was pressed. This month's program is basically the same except that to make the + sign move, we don't have to repeatedly press a key. Instead, holding down a key will cause it to move. Rather than checking the status byte, the program relies on the fact that byte >8375 contains the ASCII value of the key pressed and that if no key at all is pressed, the value in this byte will be >FF.

Here is the complete program. The program is loaded at the same address as the previous one because it is so similar.

```
AORG >7E8A
XXXX XXXX
7E8A 02E0
              LWPI >70BB
                               Load Workspace pointer:
7EBC 70BB
                               program entry point
7E8E 04E0
              CLR @>8374
                               Prepare to scan whole keyboard.
7F90 8374
7E92 0200
              LI
                   0,>170
                               Initial screen print position
7E94 0170
7E96 0420 LP BLWP @>6020
                               Start reading keyboard.
7E98 6020
7E9A 1000
              NOP
                               Replace redundant instructions from
                               previous
7E%C 1000
              NOP
                               program by NOP
7E9E 1000
              NOP
7EA0 1000
              NOP
7EA2 1000
              NOP
7EA4 C060
              MOV @>8374,1 Move word at >8374 to register 1.
7EA6 8374
                               Left-hand byte is zero because
                               byte at >8374 has been cleared.
                               Byte at >8375 contains ASCII rode
                               of key pressed.
7EAB 0261
              CI
                    1, 44
                               Was D key pressed?
7EAA 0044
7EAC 1602
7EAE 0580
              JNF
                               If no, jump aim bytes forward.
              INC
                               If yes, print one character to
                               right.
7EB0 1010
              JMP.
                   PR
                              Go to print routine.
7EB2 0281
              CI
                    1,>53
                              Was S key pressed?
7EB4 0053
7EB6 1602
                              If no, jump six bytes forward.
If yes, print one character to left.
              JNE
                   $+6
7FR8 0600
              DEC
7EBA 100B
              JMF.
                   FR
                              Go to print routine.
7EBC 0281
              CI
                   1,>45
                              Was E key pressed?
7EBE 0045
7ECO 1603
              JNE
                              lf no, jump eight bytes forward.
If yes, print one character up.
                   $+B
7EC2 0220
              ΑI
                   0,-32
```

```
7EC4 FFE0
7EC6 1005
               JMP PR
                                Go to print routine.
7EC8 0281
                    1.>58
                                Was X key pressed?
7ECA 0058
7ECC 16E4
               JNE LF
                                If no, jump back to kSCAN.
If yes, print one character down.
7ECF 0220
7ED0 0020
                                print routine
Load ASCII for +.
7ED2 0201 PR LI 1,>2B00
7ED4 2R00
7ED6 0420
              BLWP @>6024
                                Print + on screen.
7ED8 6024
7EDA 10DD
              JMP LP
                               Read next key.
```

The program entry point is at >7EBA, the same as the previous program. We can therefore use the name and address already set up in the REF/DEF table previously, which you may remember was 'MOVPLS'. So now we can firsth assembling with

7EDC XXXX

Save the whole of MiniMemory RAM to cassette with the Easybug S(ave) command, as usual, saving from 7000 to 7FFF.

We are now ready to run the program using the MiniMemory RUN option-

From the MiniMemory menu select option 2, "RUN". When asked for the program name, enter "MOVPLS". The screen should turn green and go blank. Just touch one of the left/right arrow keys very lightly. You'll be amazed at the speed with which the plus sign appear. The specific that you'll probably find it difficult be lightly. You'll be amazed at the speed with which the plus sign moves. It's so quick that you'll probably find it difficult to avoid moving the plus sign out of the area of VDP RAM reserved for the screen display and causing the program to behave strangely. To make life easier add a delay loop after the plus sign has been

AORG 7EDA LI 1,>1000 DL DEC 1 JGT DL JMP LP END

The program should now be much more manageable.

Good luck!

< Mini-Memory Tutorial Part XI > < E.N.D. >



The Universe and Everything?

The "LIFE" Program by: JOHN STOCKS

My first introduction to this program was an article in the <u>Guardian</u> last January, and I was so intrigued that I sat up until after midnight putting it into Basic.

It is a graphics game, devised by John Conway of Cambridge University in 1968, and was apparently quite well known in the early days of home computers. It is based on a grid of "cells", each of which can be "alive" or "dead", and the player initiates the game by specifying a colony of live cells. The program then takes over and proceeds according to the following rules: (1) a live cell dies unless two or three of the eight surrounding cells are also live (2) a dead cell becomes alive if exatly three of the surrounding cells are alive (3) all births and deaths for a given configuration take place simultaneously. In other words, a death takes place if there are too few or too many neighbours and a birth for exactly three.

The most astonishing feature of the game is the sheer unpredictability of colony development. Consider simple straight-line colonies for example. A row of three cells cycles continuously between a horizontal and a vertical row. A row of six develops at first and then disappears after 12 generations. A row of seven creates a "stable" state after seventeen generations and no further changes occur. (A square of four is a "stable" state for example, since no conditions exist for a birth or a death.)

Straight lines of greater lengths produce some astonishingly beautiful patterns and they nearly all reach a stable state ultimately. For continuous development an asymmetric colony must be specified, but even here a stable state is reached in a surprising number of cases. An even greater surprise is the frequent production of "birds"—groups of five cells which have the property of diagonal MOVEMENT! To demonstrate this phenomenon, draw yourself a 3 X 3 grid and fill in numbers 3,4,6,8 and 9, numbering from left to right and top to bottom, thereby forming a crude letter "J". If you now treat this as a colony and allow it to procreate according to the game rules, you will find that after four generations you have an identical J shifted diagonally by one grid unit. When running in a program, the "bird" shape appears to fly or swim across the screen, and one distorted U shape that I have come across produces three of these creatures, all flying away from each other!

As can be imagined, the amount of examination work that the poor computer has to do, even for a colony of modest size, is quite prodigious. Each live cell in a given configuration must be investigated to see whether it will live or die, then each of the dead cells in contact with any live cell must be examined to see if conditions are right for a birth. Basic is far too slow for any but the simplest of shapes so an Assembly Code program is essential, but even this becomes sluggish when a hundred or so live cells are involved. The version that I have been working on proved unusually difficult to debug. Probably taking a cue from the name of the program, several "virus-type" infections appeared, so that small faults which would not normally cause more than a minor hiccup made whole churks of the program either disappear or else move bodily to another place in memory. I have two versions available — one for Mini-Memory alone which will handle up to 256 live cells, and another for Mini-Memory plus Expansion Memory, with virtually unlimited capacity.

If anyone is interested, send a cassette to me at:

JOHN STOCKS 11 STONEHILL ROAD ROXWELL, CHELMSFORD CM1 4PF (England)

And please let me know of any interesting discoveries!

(JC - program tried and tested at the Copeland's and, as stated, it produces some extraordinary results! A program to investigate!)

< The "Life" Program > < E.N.D. >

☆ TATUNG EMSTEN ☆ MSX ☆ ENTERPRISE ☆ ☆ DRAGON ☆ LYNX ☆ TEXAS T199/4A ☆ ☆ ORIC☆ JUPITER ACE ☆ SEE THEM ALL AT

THE ALTERNATIVE **MICRO SHOW**

SATURDAY 12th MOVEMBER 1988 ASTON VILLA SPORTS AND LEISURE CENTRE BIRMINGHAM

ADULTS £2.00

CHILDREN £1.50

EASY ACCESS FROM JUNCTION 6 OF THE M6
PARKING FOR 1,000 CARS, WALKING DISTANCE
FROM ASTON RAIL STATION, CAFETERIA, ALL DAY
LICENSED BAR.
FREE MICRO COMPUTER MART MAGAZINE
HARDWARE - SOFTWARE - MAGAZINES - BOOKS
- USER GROUPS - DISCS - ACCESSORIES DEMONSTRATIONS

To exhibit contact the organisers:

EMSOFT, 88 DALES ROAD, IPSWICH, SUFFOLK IP1 4.IR TEL: (DAY) 8473 688275 (EVENING) 8473 49567 OR 8286 548548

D I T T Y Extended Basic

D I T Y - Music, lights, camera, action!

Finger picking, guitar strumming cute little guys in action here! Get out the ol' Extended Basic Module and listen to this cute Ditty (no I didn't pick the name...) but enjoy it all the same! Music maestro!

```
100 CALL COLOR(1,15,15):: FOR CC=9 TO 13 :: CALL COLOR(CC,2,15):: NEX
T CC
110 CALL SCREEN(1):: CALL CLEAR
120 FOR CHAR=100 TO 130 :: READ A$ :: CALL CHAR(CHAR, A$):: NEXT CHAR
130 FOR X-3 TO 25 STEP 6 :: FOR Y-1 TO 20 STEP 6
140 DISPLAY AT(Y, X+1): CHR$(100)&CHR$(101)
150 DISPLAY AT (Y+1, X): CHR$ (103) & CHR$ (104) & CHR$ (105) & CHR$ (106)
160 DISPLAY AT (Y+2, X): CHR$ (107) &CHR$ (108) &CHR$ (109) &CHR$ (110)
170 DISPLAY AT(Y+3,X):CHR$(111)&CHR$(112)&CHR$(113)&CHR$(114)&CHR$(11
5) &CHR$ (116)
180 DISPLAY AT(Y+4,X):CHR$(117)&CHR$(118)&CHR$(119)&CHR$(120)&CHR$(12
1) &CHR$ (122)
190 DISPLAY AT(Y+5,X):CHR$(123)&CHR$(124)&CHR$(125)&CHR$(126)
200 NEXT Y :: NEXT X
210 CALL VCHAR(1,1,130,72)
220 CALL VCHAR (1, 30, 130, 72)
230 S=250 :: LLA=110 :: LLAS,LLBF=117 :: LLB=123 :: LB=247
240 LC=131 :: LCS,LDF=139 :: LD=147 :: LDS,LEF=156 :: LE=165 :: LF=17
5 :: LFS.LGF=185 :: LG=196 :: LGS.LAF=208 :: LA=220 :: LAS.LBF=233
260 HB=98B :: C=262 :: CS,DF=277 :: D=294 :: DS,EF=311 :: E=330 :: F=
349 :: FS,GF=370 :: G=392 :: GS,AF=415 :: A=440 :: AS,BF=466 :: B=49
270 HC=523 :: HCS,HDF=554 :: HD=587 :: HDS,HEF=622 :: HE=659 :: HF=69
9 : HFS,HGF-740 : HG-784 : HGS,HAF-831 : HA-880 : HAS,HBF-932
290 HHC=1047 :: HHCS,HHDF=1109 :: HHD=1175 :: HHDS,HHEF=1245 :: HHE=1
319 :: HHF=1397 :: HHFS,HHGF=1480 :: HHG=1568 :: HHGS,HHAF=1661
300 HHA=1760 :: HHB=1976 :: HHAS, HHBF=1864
310 FOR AA=1 TO 2
330 CALL SOUND(S,LC,0):: GOSUB 3320 :: GOSUB 2960 :: GOSUB 3340
370 CALL SOUND(S,LG,3):: GOSUB 3360 :: GOSUB 2960 :: GOSUB 3380
410 CALL SOUND(S,LA,3):: GOSUB 3320 :: GOSUB 2960 :: GOSUB 3340
450 CALL SOUND(5,L6,3):: GOSUB 3360 :: GOSUB 2960 :: GUSUB
                                                                      3.380
510 CALL SOUND(S.LD.0):: GOSUB 3320:: GOSUB 2990 :: GOSUB 3340
550 CALL SOUND(5,L6,3):: GOSUB 3360 :: GOSUB 2990 :: GOSUB 3380
590 CALL SOUND(5,L4,3):: GOSUB 3320 :: GOSUB 2990 :: GOSUB 3340
630 CALL SOUND(5,L8,3):: GOSUB 3360 :: GOSUB 2990 :: GOSUB 3380
670 NEXT AA
700 CALL SOUND(S,LE,O):: GOSUB 3320 :: GOSUB 3020 :: GOSUB 3360
740 CALL SOUND(S,LB,3):: GOSUB 3320 :: GOSUB 3020 :: GOSUB 3360 780 CALL SOUND(S,C,3):: GOSUB 3320 :: GOSUB 3020 :: GOSUB 3360
820 CALL SOUND(S,LB,3):: GOSUB 3320 :: GOSUB 3020 :: GOSUB 3360
870 CALL SOUND(S,LA,O):: GOSUB 3320 :: GOSUB 3050 :: GOSUB 3360
910 CALL SOUND(S,HE,3):: GOSUB 3320 :: GOSUB 3050 :: GOSUB 3360
950 CALL SOUND(S,B,3):: GOSUB 3320 :: GOSUB 3050 :: GOSUB 3360
990 CALL SOUND(S,HC,3):: GOSUB 3320 :: GOSUB 3050 :: GOSUB 3360
1040 CALL SOUND(S,LB,0):: GOSUB 3320 :: GOSUB 3080 :: GOSUB 3340
1080 CALL SOUND(S,LA,3):: GOSUB 3360 :: GOSUB 3080 :: GOSUB 3380
1120 CALL SOUND(S,LB,3):: GOSUB 3320 :: GOSUB 3080 :: GOSUB 3340
1160 CALL SOUND(S,LA,3):: GOSUB 3360 :: GOSUB 3080 :: GOSUB 3380
1220 CALL SOUND(S,LG,O):: GOSUB 3320 :: GOSUB 3110 :: GOSUB 3340
1240 CALL SOUND(6,16,3):: GOSUB 3360 :: GOSUB 3110 :: GOSUB 3380
1300 CALL SOUND(5,L6,3):: GOSUB 3320 :: GOSUB 3110 :: GOSUB 3380
1340 CALL SOUND(S,LG,3):: GOSUB 3360 :: GOSUB 3110 :: GOSUB 3380
1400 CALL SOUND(S,LC,0):: GOSUB 3320 :: GOSUB 2960 :: GOSUB 3340
```

```
1440 CALL SOUND(S,LG,3):: GOSUB 3360 :: GOSUB 2960 :: GOSUB 3380
 1480 CALL SOUND(S,LA,3):: GOSUB 3320 :: GOSUB 2940 :: GOSUB 3340
1520 CALL SOUND(S,LG,3):: GOSUB 3360 :: GOSUB 2960 :: GOSUB 3380
 1580 CALL SOUND(S,LD,O):: GOSUB 3320 :: GOSUB 2990 :: GOSUB 3340
 1620 CALL SOUND(S, HD, 3):: GOSUB 3360 :: GOSUB 2990 :: GOSUB 3380
 1660 CALL SOUND(S, B, 3):: GOSUB 3320 :: GOSUB 2990 :: GOSUB 3340 1700 CALL SOUND(S, HC, 3):: GOSUB 3360 :: GOSUB 2990 :: GOSUB 3380
 1750 CALL SOUND(S,C,0):: GOSUB 3320 :: GOSUB 3140 :: GOSUB 3360
 1790 CALL SOUND(S, 6, 3):: GOSUB 3320 :: GOSUB 3140 :: GOSUB 3360
 1830 CALL SOUND(5,A,3):: GOSUB 3320 :: GOSUB 3140 :: GOSUB
 1870 CALL SOUND(5,6,3):: GOSUB 3320 :: GOSUB 3140 :: GOSUB
                                                                            3360
 1920 CALL SDUND(S,D,O):: GOSUB 3320 :: GOSUB 3170 :: GOSUB 3360
 1960 CALL SOUND(S,F,3):: GOSUB 3320 :: GOSUB 3170 :: GOSUB
                                                                            3360
 2000 CALL SOUND(S,C,3):: GOSUB 3320 :: GOSUB 3170 :: GOSUB
                                                                            3360
 2040 CALL SOUND(S,D,3):: GOSUB 3320 :: GOSUB 3170 :: GOSUB 3360
 2090 CALL SOUND(S,HC,0):: GOSUB 3320 :: GOSUB 3200 :: GOSUB 3340
 2130 CALL SOUND(S, HG, 3):: GOSUB 3360 :: GOSUB 3200 :: GOSUB
                                                                             3380
 2170 CALL SOUND(S, HA, 3):: 603UB 3320 :: 605UB 3200 :: 605UB 3340
 2210 CALL SOUND(S,HG,3):: GOSUB 3360 :: GOSUB 3200 :: GOSUB 3380
 2270 CALL SOUND(S,C,O):: GOSUB 3320 :: GOSUB 3140 :: GOSUB 3340
 2310 CALL SOUND(S,6,3):: GOSUB 3360 :: GOSUB 3140 :: GOSUB
                                                                            3380
 2350 CALL GOUND (8, A, 3):: 605UB 3320 :: 605UB 3140 :: 605UB 3340
 2390 CALL SOUND(S,G,3):: GOSUB 3360 :: GOSUB 3140 :: GOSUB 3380
 2450 CALL SOUND(S,LC,0):: GOSUB 3320 :: GOSUB 2960 :: GOSUB 3340
2490 CALL SOUND(S,LG,3):: GOSUB 3360 :: GOSUB 2960 :: GOSUB 3380 2530 CALL SOUND(S,LA,3):: GOSUB 3320 :: GOSUB 2960 :: GOSUB 3340
 2570 CALL SOUND(S,LG,3):: GOSUB 3360 :: GOSUB 2960 :: GOSUB 3380
 2630 CALL SOUND(S,LC,0):: GOSUB 3320 :: GOSUB 3230 :: GOSUB 3340
2670 CALL SOUND(S,LG,3):: GOSUB 3360 :: GOSUB 3230 :: GOSUB 3380 2710 CALL SOUND(S,LA,3):: GOSUB 3320 :: GOSUB 3230 :: GOSUB 3340
2750 CALL SOUND(S,LG,3):: 60SUB 3360 :: 60SUB 3230 :: 60SUB 3380
2800 CALL SOUND($#6,LC,3):: GOSUB 3320 :: GOSUB 3340 :: GOSUB 3360 :: GOSUB 3380 :: GOSUB 3320 :: GOSUB 3340 :: GOSUB 3360 :: GOSUB 3380 :: GOSUB
3340 :: NEXT GG :: CALL SOUND(S,40000,30)
2950 END
2960 CALL SOUND(S,C,3,E,3):: CALL SOUND(S,LG,3):: RETURN 2990 CALL SOUND(S,LB,3,F,3):: CALL SOUND(S,G,3):: RETURN
3020 CALL SOUND(S,E,3,GS,3):: CALL SOUND(S,LB,3):: RETURN
3050 CALL SOUND(S,E,3,A,3):: CALL SOUND(S,C,3):: RETURN
3080 CALL SOUND(S,D,3,FS,3):: CALL SOUND(S,C,3):: RETURN 3110 CALL SOUND(S,F,3,6,3):: CALL SOUND(S,LB,3):: RETURN
3140 CALL SOUND(S, HE, 3, HC, 3):: CALL SOUND(S, 6, 3):: RETURN
3170 CALL SOUND(S,HF,3,B,3):: CALL SOUND(S,G,3):: RETURN
3200 CALL SOUND(S, HHC, 3, HHE, 3):: CALL SOUND(S, HG, 3):: RETURN 3230 CALL SOUND(S, LE, 3, LC, 3):: CALL SOUND(S, LG, 3):: RETURN
3260 DATA 00000000000031C,00000000000E01C,0,000000010101010101,6080
801C22581802
3270 DATA 03000010220D002,008080404040404,0000000001020204,84858001
23110837,105000C1E2C408F6
3280 DATA 808080804020201,040408183151509,5050D08018876011,0505053CC3
3290 DATA 1010088884844242,00000000000003,000000000000054,8E85848E8
3300 DATA OEBE7D0A05000719,183C3C9B000000C3;7F40407F4040B0,FFB254FF3B
,FC8282FC54
3310 DATA 100F000000000101.29C90909699901FF,78484848484C407F,00000000
OOOBO4OC,,,,FFFFFFFFFFFFFF
3320 CALL CHAR(118, "OEBE7D0A0A000719"):: RETURN
3340 CALL CHAR(121, "FF542CFF1C"):: RETURN
3360 CALL CHAR(118, "0F8F7D0405000719"):: RETURN
3380 CALL CHAR(121, "FFB254FF38"):: RETURN
```

Page 15

COLIN HINSON



Feripheral Software Overview

99/4A DISK PERIPHERAL SOFTWARE OVERVIEW by: COLIN HINSON

The Device Service Routine (DSR) ROM in the 99/4a Disk peripheral is designed to give the user access to the disk by means of a system using three different 'levels', which, with the addition of some utility routines gives the user complete access to a normally formatted disk without the need for any knowledge as to how the actual disk controller works.

Each level uses those features implemented at a lower level to add new features, (a sort of 'building block' system).

LEVEL 1 FEATURES

- * Communication with the FD1771 chip
- * Record read/write functions
- * Disk formatting functions
- # Soft error corrections

This level is the only level which must know precisely what the disc hardware is. This allows higher levels to be independent of both the controller chip type, and the rest of the disc controller hardware. Each of the higher levels sees the disc simply as a linear storage device, addressed by disc unit-number, a physical record number, and a read or write operation.

If the disc controller chip is changed (such as the Myarc card) then it should only be necessary to replace this part of the software. All the higher levels are designed to be independent of the actual physical disc structure which this level deals with, except for sector size which is assumed to be 256 bytes. Smaller sector sizes could easily be supported by setting up the sectors in such a way that the total adds up to 256 bytes - for instance, if a sector size of 64 is used, each sector requested from a higher level would take up 4 sectors at level 1.

LEVEL 2 FEATURES

- * All level 1 features plus:
- * Creation and deletion of files
- * File allocation dynamically extendable
- # Data accessed by filename and physical record displacement
- * 'Mixed hybrid' file format (see below)

The actual 'file' concept is created at this level, with each file being known by its name and the displacement of the physical record within the file - a physical record being defined as one disc sector (256 bytes).

On each disc is maintained a directory and bit-map of the sectors. This allows for file and record management (i.e. deletion and creation). The file format available is called the 'mixed hybrid' format, and is a mixture of contiguous and non-contiguous '(fragmented) file formats. A lot of overhead has to be carried by fragmented files in the form of pointers — these pointers are required in case relative access is required to the file and point to each data record of the file.

The files on this level are allocated in 'clusters' of contiguous records in order to combine the advantage of the flexible allocation of non-contiguous files with the low everhead, and the easy access of contiguous files. Whenever new records are requested, the clusters are expanded if possible, if a cluster cannot be expanded then a new one is started.

LEVEL 3 FEATURES

- # All level 2 features, plus:
- * Program and data files
- * Fixed and variable record formats
- # Relative and sequential access # Internal and ASCII data types

The disc management software is completed by the addition at this level, of the relative/sequential access and the fixed/variable record formats. This level takes care of the 'blocking' of one or more logical records into a physical record (as with DIS/VAR format). When relative access is required, it computes the physical record in which the logical record is to be found, updates that record and passes the physical record back to the level 2 file update routines.

UTILITY ROUTINES

As you may have noticed, there are some functions which have not been catered for, as they are not part of the normal file I/O system. These are catered for by means of some utility routines which have been mentioned previously in this publication. These 'subprograms' are:

- # Direct level 2 file access
- Direct sector (Allocatable unit) access
- # Modification of file protection
- * Disk formatting
- # File rename.

Methods of accessing these routines will be described later.

99/4a DISK PERIPHERAL - OPERATIONAL INFORMATION

There are three basic methods used to store data on the diskette. These are:

- 1> 'Program' (Memory Image) format
- 2> Variable format
- 3> Fixed format

Variable and Fixed format are really 'variations on a theme', in that, each sector, (or AU), contains as many records of either format as it is possible to write to that sector - without overflowing it (i.e without writing more than 256 bytes).

Program format is used to store an image of the data in memory on dislette byte for byte, each byte in each sector being used (except for the last sector associated with the particular file, which may not be fully used due to the length of the file not being exactly divisible by 256).

Methods of access

There are three methods of access, each one being associated with one particular format described above. The methods of access are (in order):

- 1> Physical I/D
- 2> Sequential access
- 30 Relative access

Sequential access

When the data records in a file are accessed strictly in the order of increasing address on the medium, each record is said to be sequentially accessed. This is the access method used for accessing such things as magnetic and paper tapes etc, in which all the records up to and including the one required must be read in order to access the particular record required. In this mode of access, the parameters for

East Anglia Region 99'ers User's Group

the data transfer do not specify a physical record number, as it is implied that the logical record currently indicated by some data transfer pointer is the one which is required. Restore/rewind operations are either implicitly done or explicitly done prior to the first data transfer. As each logical record is transferred, the access pointer moves to the first byte of the next logical record (which in the case of this DSR is usually the length indicator).

Sequential access methods have the advantage that variable record lengths can be used (such as the well known "VAR 80") and so the number of records per sector can be increased by an amount dependent on the length of each particular record. For instance, ten 24 byte records could be written on a 256 byte sector, whereas if "FIXED 80" were to be used, then only 3 records (=240 bytes) could be written even though there are still only 24 bytes of usable data per record.

Variable format (sequential access) sectors are recorded on the disc with an extra byte added at the start of each record, and a final byte at the end of the last record of the sector. The first byte of each record indicates the length of the record in bytes, a negative number (usually)FF) indicating that there are no more records on that sector. The action of the computer when reading the sectors from the data buffer in VDP ram is to read the first byte (length of record), then read the required number of bytes as the record from VDP ram to a new location, read the next byte (length of record). etc etc until a negative number is found as the length. At this point another sector is read from the disc to VDP ram and the process repeated again until all the data for the appropriate file has been read.

Relative (Random) access

Relative access allows data in a file recorded in fixed format to be accessed by logical record number. The records may be accessed in any order regardless of the order in which they were written or the order in which they appear in the file.

As the DSR software must be able to locate a record solely by its number, relative access can only be supported on either Indexed Files or Filed Length files. In this DSR, only "Fixed length" files are supported. (Indexed files are files for which an "Index" is maintained by the diskette through which a particular record can be located by looking it up in the index.)

Physical I/O

With the Physical 1/O access, the data on the disk is considered to be organised in blocks of 256 bytes by the DSR software. Each byte can be of any value (ie -256 to +255) and no attempt is made to interpret these at data transfer. The existance of records or files is completely ignored by this access method.

You will notice that this method of access is a "Level 1" access. The rest of the disc software (i.e., Levels 2 and 3) reduces all access methods to physical I/O by converting logical record numbers into physical track and sector data. This information is used to specify the disc sector that is to be transferred by the Physical I/O. Physical I/O is not available directly to the user other than in the form of an assembly language sub-program within the DSR. (See more later for Sub-Programs).

Directory Organisation

The directory organisation implemented within the DSR supports only a single level directory, that is to say that no FILE can be a directory containing pointers to other files. This means that each file on the disc can be readily identified by a single name, e.g.: DSM1.filename — which would specify a file called "filename" on the diskette in drive 1.

This approach to the dislette file organisation prevents access to a catalog file as such on the disc, as no such file can physically exist. In order to overcome this problem, a semi-catalog file can be created by the DSR software and accessed by the user. The file which is

created (and remember it is not physically on the disc, so don't go looking for it with the Disc Manager!), is of the Fixed format, relative access type. The file contains 128 records, each containing information about the associated catalog entry and is described in detail below. can be accessed as: DSK1. DSK.volname or as a standard data file but without a name.

Please note that not all the file operations have been defined for this catalog file, and only the standard OPEN, READ, and CLOSE are supported. Other operations such as DELETE, EOF, etc. are considered to be illegal, and an error will be returned if these operations are used.

Catalog file access from Basic

The Catalog file can be accessed as a read-only file by the Basic user. The file has no name, and is of the INTERNAL, FIXED format type. The file can be opened by (for example):

OPEN #1: "DSK.", INPUT, INTERNAL, RELATIVE
The record length will automatically be defaulted by Basic to the correct value, so this should not be entered. If however the user wants to specify the length, then it must be specified as 38- all other lengths will result in an error message.

The Catalog file acts as if it is Protected, and as mentioned above, it will only allow INPUT access.

The file is written in the normal Basic INTERNAL format, and each record contains four items: one string and three numerics. There are 128 records in the file, and they are numbered 0 thru 127.

This record contains data about the volume for which the catalog file was created. The string gives the name of the disc (up to 10 characters) and the numerical items are as follows:

1> Always 0 (for record 0)

Total number of sectors on the disc
 Total number of free sectors on the disc

Records 1 thru 127:

These records contain information on the corresponding file in the Catalog. Non existant files will give a null string for the first item, and O's (zeros) for the numeric items. Files which exist will give the file name for the string, and the following numeric items:

1) = Filetype (if number is negative, file is protected)

1= DISPLAY/FIXED datafile

2= DISFLAY/VARIABLE datafile

3= INTERNAL/FIXED datafile

4= INTERNAL/VARIABLE datafile

S= Memory Image file (Program File) 2> = Number of AUs allocated to the file

3> = Number of bytes per record (0 for type 5 file)

Catalog file access by application program or user (Please read the above information first)

In order to enable access from assembly language programs, the following additional information is required:

The Catalog file contains 128 records of 38 bytes and is output INTERNAL format (i.e. a length byte followed by a data item). Each (i.e. a length byte followed by a data item). Each of the records contains four of these data items:

An ASCII string containing up to 10 characters, or a null string. Three numeric values in standard 8 byte floating point format.

Record O contains information about the volume itself, while records 1 thru 127 contain information about the relevant file for each "slot" in the catalog.

> East Anglia Region 99'ers User's Group Page 19

The information in the records is as follows:

- 1> An ASCII string of up to 10 characters containing the name of the file in the specified slot. For record 0 this is the Volume name.
- 2> A floating point value of between -5 and +5. These values represent the same information as given for Basic.
- 3> The number of AUs allocated for the file (record 0 = total AUs on the disc)
- 4> The number of bytes per logical record 0 for Program file. (Record 0 = Free AUs remaining on the disc)

If a catalog slot is empty, the filename will contain a null string and the numeric entries will contain floating point zeros.

INTERNAL DATA STRUCTURE

Physical device format

The physical device (diskette) is logically subdivided into "Allocatable Units" (AU's). An AU is defined as being an integral number of physical records on the device. The total number of AUs on any device is less than 40% (ie: each AU can be addressed by a 12 bit word). The first AU is numbered 0.

The physical record length is the size of the block of data which can be read or written to the device at one time. For the Disc Peripheral, the AU and the Physical Record are equivalent to one disc sector (256 bytes).

Summary of system reserved sectors:

Sector O contains data concerning the volume, such as available (free) sectors, disc name etc.

Sector I contains pointers to other sectors which contain

descriptions of the appropriate file. Normally there is a pointer in sector one for each file which exists on the disc.

Volume information block (VIB), Sector 0

This block contains disc configuration data as required by the disc software. This includes available number of AUs, Volume name, format information etc. Included in this block is the "Allocation Bit Map":

The allocation bit map is used to indicate to the disc software the availability of individual sectors on the disc. A "1" indicates that the sector associated with that "bit" has been allocated, and a "0" that the sector is available for use. The first bit in the map is for sector 0, the second for sector 1 and so forth. When the disc is initialised (WITH VERIFY = YES if using DM1000 or similar), then the bits for bad AUs are set to "1" along with the bits for non-existant AUs and the 2 system reserved AUs. All the remaining bits are of course set to zero.

File Descriptor Index Record (FDI), Sector 1

This sector contains alphabetically sorted pointers to each file Descriptor Record (FDR), and enables the system to keep track of the location of each FDR on the disc.

NOTE: If either Sector 0 (VIB) or sector 1 (FDR) are bad or corrupted then the whole disc is considered bad by the system, as it can no longer keep track of information stored on the disc.

File Descriptor Record (FDR) - Any Sector

This record is used to map filenames into physical locations of the files on the disc. Each entry contains information about the file such as type, record type, data type, size of file, etc.

File Control Block (FCB) in VDP RAM

This is a copy of the FDR which is maintained in VDP RAM while the file is open. It may additionally contain some more up-to-date information about the file. One FCB is required for each file which is

currently opened. It is the memory taken by these FCBs which is affected when "CALL FILES" is used in BASIC.

DETAILED DESCRIPTION OF DISC FORMAT

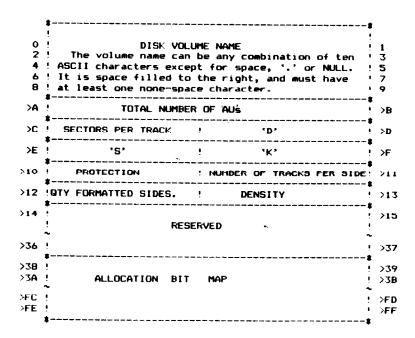
A single sided diskette used with the T.I. Disc Controller has the fullowing specifications:

Diskette type: SA 104 (ANSI standard 5.25")
Encoding method: FM single density
92160 Bytes per disc
2304 Bytes per track
256 Bytes per sector
40 Tracks per side
9 Sectors per track

The capacities given are for a single sided, single density system. Using double sided will of course double the bytes per disc, using double density (Myarc type controller) will double the capacity again.

For ease of description, the following information assumes that the diskette is addressed as a 'linear' medium, that is to say, sector 0 is the first sector of track zero, sector 1 is the second and so on sector 359 being the last sector of track 39. It should be noted that this is not strictly correct as the sectors are in fact 'interleaved' on each track to obtain faster access when reading. If a double sided set up is being used then the physical layout of the second side is the reverse of the first side, that is to say, sector 360 is physically on the opposite side of the disc to sector 359, and sector 719 is opposite sector 0.

VOLUME INFORMATION BLOCK LAYOUT



East Anglia Region 99'ers User's Group Page 21

Bytes (A-)B show the total number of allocation units on the volume. This information should match the Allocation Bit Map data.

Bytes >D->F contain the ASCII letters 'DSA'. These letters are checked by the T.1. disc managers to see if the disc has been initialised.

Byte >10 Contains the ASCII 'P' if the disc is Proprietry

Protected. This byte will normally otherwise be an ASCII space.

Rytes >12->37 are received for what were intended to be future expansion. This could be date and time of creation etc. The T.I. controller will set all these to zero.

Bytes 338-3FF contain the allocation bit map. The map can control to 1600 256-byte sectors (-400K bytes) - this will allow double sided, double density diskettes without modification to the map layout. Each bit in the map represents one sector on the disc. A logical one in the bit map means that the corresponding sector has been allocated, a logical zero means that the sector is available for use.

The Volume name can be used as an alternative to the drive name that is to say the user can specify a disk drive in either of the following ways:

> DSk.volname.filename or DSK1.filename

If the volume name is specified, then the system will look at each drive in sequence until it finds the specified volume. If more than one drive contains a volume with that name, then the lowest drive number will be assigned.

FILE DESCRIPTOR INDEX RECORD, (Sector 1)

This sector contains up to 127 two-byte entries. Each of these points to a File Descriptor Record, and are alphabetically sorted according to the file name in the File Descriptor Record. The list starts at the beginning of the block, and ends with a zero entry.

As the file descriptors are alphabetically sorted, a binary search can be used to find any given filename. This limits the maximum number of searches to 7 if more than 63 files are defined. Generally if between 2**(N-1) and 2**n files are defined, a file search will take at the most N disc searches. To obtain faster directory response times, data blocks are normally allocated in the area above sector >22, the area below this being used for file descriptors and only used for data when there are no more sectors available above >22.

To be continued.... (See more next issue!)....

< 99/4A DISK FERIPHERAL SOFTWARE OVERVIEW > < E.N.D. >

FROM THE DESK Secon ince of Joann Copeland Secretary/Treasurer EAR 99'ers U.G. INCOME Library Income \$ 48.73

Fostage/Stamps

Subscriptions Income \$.00
Income Sub-Total	48.73 +
EXPENDITURES;	

\$# END-OF-MONTH TOTAL #\$ \$ 477.12

Petty Cash **\$ 45.60** £ 4.96

Expenditures Sub-Total.....

Checkbook Balances: American \$ 97.32 British & 193.97

Checks for Deposit: 0 @ # .00

Exchange Rate at \$1.65/\$1.70=#1.00

ESTIMATED EXPENDITURES:

Estimated Postage Expenses \$ 00.00
Photocopying Expenses = \$ 57.43

Vol 2 Issue 3 188 16 6 originals @ 85 = 510 @ .02599 = \$ 13.25

Advertising Leaflet 17 originals @ 100 = 100 @ .02599 = \$ 2.59

Vol 2 Issue 4 188 14 originals @ 80 = 1120 @ .02599 = \$ 29.11

Vol 2 Issue 5 188 16 originals @ 80 = 480 @ .02599 = \$ 12.48

ESTIMATED INCOME:

Subscription Renewals (Oct/Nov 1988) \$.00 Library Income (Oct/Nov 1988) \$.00

>>>> NEXT MEETING: <<.<<

TO BE ANNOUNCED VIA SERARATE NOTICE INCLUDED WITHIN THIS NEWSLETTER (Gee, another enclosure ...?)

Until then, HAPPY TAPPING!

Treasury Report . . . E N D

East Anglia Region 99'ers User's Group Page 23

VOLUME 2 ISSUE 6 OCTUBER 188



FROM:

SSgt D S Copeland 1977CS/XP PCS Box 5927 APO NY 09179-5379

TO:

MIAMI CO. AREA 99/4A (HCUG) POST OFFICE BOX 1194 PERU, IN (29E) USA 46970