EAST ANGLIA REGION 99'ERS

Volume 1 Issue 12
April 1988

HAPPY EASTER!

Great Britain Mail:

Scott Copeland, President
EAR 99'ers U.G.
13 Elm Walk
Lakenheath, Suffolk
England IP27 9QR
Tel: 063881-3457

United States Mail:

SSgt. D. S. Copeland, President
% EAR 99'ers U.G.
1979CS/LGG
PCS Box 5927
APO New York 09179-5379
Tel: 011-44-63881-3457

Contents:

Now, turn the page for an EAR-full of information.....

MINUTES OF THE MEETING

12 March 1988

by: Jo Ann Copeland
Secretary/Treasurer

The meeting was called to order by the President at 2:32 PM with 12 members present. The minutes of the last meeting were read and approved, as was the treasury report.

Old business: The Technical SIG was discussed and members were notified how the SIG went. The TI Fayre was discussed once again, with members volunteering for booth services.

New business: Members were asked if they were attending the Bloxwich meeting. Notice was made of the Library Order to LA 99'ers and the library would be updated in upcoming newsletters. The A5 format discrepancy for the newsletter was discussed and a motion was made, approved and seconded to go to A5 format in May '88. It was noted the User Group owed JoAnn 30 disks. Approval was made to purchase 4 sets of joysticks and 5 dual cassette leads from John at a cost of L30.00, and additional joysticks would be purchased when available. These would be sold to non-members requesting them at L7.50 for the joysticks and L3.00 for the dual cassette leads. Membership prices are set at L5.00 and 2.00 respectively. Since we were put on the TI Consumer Services list, we have had numerous requests for these items.

Postage costs were discussed, and the treasury report for postage expense was not just for mailing newsletters. This cost includes mailing Library Disks, Cassettes, Correspondence, and Newsletters.

Approval was given the Secretary Treasurer to order 50 disks from Tenex whenever the library supply dropped below 50.

Tony had a copy of a Digitizer Manual and this was shown at the meeting.

The NEXT MEETING is scheduled for 16 APRIL 1988 at 2:00 PM, while the next SIG MEETING is scheduled for 30 APRIL 1988 at 2:30 PM. This SIG will include using BugOuto to put cartridge based modules onto disk.

The meeting was adjourned at 3:15 PM. At that time, demonstrations were shown on the MANCALA strategy game and the WALT DISNEY SERIES produced through, and for, TI including PETER PAN. The speech, graphics, and operation of these 'games' are fantastic! Purchases were made from the library at this time, along with the sale of three sets of joysticks.

Our thanks this month to the Ziegler's for supplying the goodies to eat at the meeting!

Mark and Mike left late as usual! (Actually, I just put this in to see if Mike was reading the newsletter, which we already know he isn't.....)

>>>>> NEXT MEETING <<<<<
16 APRIL 1988
2:00 PM
13 ELM WALK

>>>>> NEXT SIG GROUP <<<<<
30 APRIL 1988
2:30 PM
13 ELM WALK

>>>>> E N D <<<<<

FORMS TO BE RETURNED: If you received a BLUE renewal notice return the ORANGE form (with your check) before the end of the month to keep your subscription current. Please note that subscription rates have NOT gone up! Also enclosed is your form to VOTE. Please return this form also before the end of the month (or as soon as possible). Going color blind yet? Well, it's better than black ink on black paper! But with Jo you never know...

>>>>> AWARDS OF THE MONTH <<<<<

EDDY CARTER receives the "I'M IN THE DOG HOUSE" AWARD for (well, he knows why!). DEREK DUDDY receives the "I KNOW HOW TO PACKAGE DISKS WHERE >NO ONE< IS GOING TO OPEN THE PACKAGE" AWARD. And MIKE BRICK receives the "I NEVER READ THE NEWSLETTER SO I DIDN'T KNOW" AWARD. JO ANN receives the "OOPS, I REALLY SHOULD'VE WRITE-PROTECTED THE LIBRARY DISK AND NOW IT'S TOO LATE" AWARD!. Congratulations to Eddy, Derek, Mike, and Jo! Watch out, you never know who's next...!

>> FOR SALE <<

Lots for sale this month, but by the time this gets through your mail slot some of it may be gone (how come you mail things out on the same day to the same areas but they get there on different days?) I just post 'em, I don't deliver 'em! And considering the poor mailman (mailperson) delivers mail on bicycles - it's no wonder!

>> BLOXWICH <<

A review on the BLOXWICH workshop will be in the next newsletter, as this will already be in reproduction when we attend BLOXWICH. Let's see now - whose name am I going to involuntarily volunteer to write a review...?

BYTES AND PIECES I
by: Bigfoot

Firstly, the Winner of the Mystery Disk of the Month for the February Meeting was Chris Ziegler. Congratulations to Chris!

I apologize for the rough copy of the last newsletter. Really not up to standards, however there was Dated Material in it and it had to be sent out by a certain date. We had moved our normal meeting date up to make room for Bloxwich and the issue had to be mailed before March 12. If I had turned it back into Repro, 5 more days would've elapsed, and you would have missed the meeting date. Wait a minute, why all the groans for getting the issue on time? What was that about WANTING to miss the meeting??? Ouch! Anyway, with future meetings on schedule, be assured I won't accept the issue unless it's up to standards. (By the way, do we have a standard?)

* * * * * *

IF YOU RECEIVED A >BLUE< NOTICE IN YOUR ENVELOPE IT'S TIME TO RENEW YOUR SUBSCRIPTION! THE >ORANGE< FORM ENCLOSED WITH IT NEEDS TO BE RETURNED BEFORE THE END OF THE MONTH TO KEEP YOUR SUBSCRIPTION CURRENT!

* * * * * *

We'll be ONE YEAR OLD in May, with 33 MEMBERS and exchanges at 38 (changing all the time). PRETTY GOOD FOR BEING 11 MONTHS OLD! Let's keep the numbers and the interest up! WE WELCOME THE FOLLOWING NEW MEMBERS TO OUR GROUP: JOHN BUTCHER, BRYAN JONES, DOUGLAS PICKLES, and NEIL WILSON. WELCOME ABOARD! >>LET'S HELP THEM FEEL WELCOME!<<

We had gotten approval for the RAFFLE on the British Law side, however disapproval was received on the Base side. (Ouch!) Our Constitution states we must abide by British Law and American Law, and being a Private Non-Profit Organization through the Military Base can cause some minor problems, however we do get reproduction at a reasonably inexpensive cost. Part of the ups and downs of military life, but NEVER FEAR, we're working on another angle!!! Where there's a will, there's a way. WATCH FOR MORE!

Well, the Technical SIG for March 5th went as planned (meaning totally haphazardly!). Members met at The Copeland's home at 3:00 in the afternoon and then headed towards the technical shop at 3:30, returning to the Copeland residence at approximately 6:30 PM, and ended at 3:30 the following morning. Yes folks, a 12 hour SIG! Mark Playle ordered in Pizza, and Veronica made the mistake of putting a video in (Karate Kid Part I and Part II) which totally got Mike Brick's attention, who was pretty much enthralled until the end of Part II. Needless to say everyone had a good time, except for Bryan Jones when his SIG didn't work as planned. After hours of searching and tracing wires it was found that a Trace Wire hadn't been cut deep enough and seconds later his console was up and running. Mike forgot his ExBasic module, and Mark brought a Mechatronics Module (which I understand is seemingly incompatible for the SIG). Anyway, several members should now be happy with their XB Module installed inside their consoles, and we sincerely apologize for the death of any other consoles (hiccup!). Only kidding! IT WAS FUN!!

Do you believe Scott's forms were approved for an extension of his tour thru to July 1991? Better yet, do you believe we even VOLUNTEERED to extend our tour? Looking back in time it wasn't a real swift idea! The rate of exchange wasn't at the rate it's at now! But, it looks like you're stuck with us until July 1991 unless something happens between now and then! (More groans?)

>>>>> NEXT MEETING <<<<<

APRIL 16, 1988

2:00 PM

13 ELM WALK

>>>>>>>>>>>><<<<<<<<<<<<

Bytes and Pieces continued on the next page...

BYTES AND PIECES II

by: Bigfoot

Did you see the reports for last month? >> $269.22 was brought into the group funds and Treasury by purchasing Library Disks! Have to run another coupon soon...! THANKS TO ALL OF YOU FOR YOUR SUPPORT!!

We received our Library order from LA 99'ers but the disk didn't work. Something happened on the Archiving side and, needless to say, we had to return this disk and request it again. I can hear their Librarian groaning now... We should, though, have 10 new library disks put into the library soon, so watch your newsletters for any updates to the Library!

We purchased a Prism 2000 Modem and have been investigating Prestel and other facets of telecommunications. Our Prestel E-Mail number is 219999131 if anyone is interested in leaving a message, etc. If you have any questions on obtaining your free Modem, contact Scott for details!

No one wanted to take a shot at the Lucky Leprechaun Contest huh? Well, for those interested (anyone?) there were >29< Gold Coins in the Leprechaun's Pot of Gold. You shoulda' guessed with February having been a Leap Year Month! What will JoAnn come up with next???

The group questionnaire forms have come in and the following comments should be noted (THANKS FOR THE COMMENTS!):

Every questionnaire was marked "Continue The Group AND the Newsletter". 100% agreement there! Well, it looks like you'll all be stuck reading Jo's burbles for another year (Groan!). Members would like to see more articles on: Basic, Extended Basic, E/A, Word Processing, Telecommunications, Data Bases, More of SIG's, C99, Logo, and, of course, Adventures. Most thought the balance of the newsletter was good. SIG groups were requested for: W/Processing, Data Bases, Telecommunications, Basic, ExBasic, E/A, M/M. If we can get this organized, it looks like we'll be busy! Extra comments included: "I admire the way you find the time for all the work" and, "The group is doing well, let's have more of the same!". More of "Keep up the good work!", and what we all knew was going to be said: "Do Not Stray off of the subjects at the meetings!" Yes, we do, but won't any more, will we? Well, it looks like every one is happy with the way things are going, so let's not just keep it the same, but let's do it better in 1988-1989!

What this means of course, is that we need more input from members on the items requested. If you feel you are semi-knowledgeable in a particular subject, why not submit an article! No, we're not gonna' laugh! >Everyone< knows a little bit about something! And >everyone< can offer something that someone else hasn't considered! So why not give it a shot! ANY, I mean, ANY, article will be gratiously accepted! And don't worry about format, we'll fix it for you! You just have to get it here!

WELL - THIS ISSUE MEANS IT IS TIME TO VOTE!!! YOU'LL FIND ANOTHER FORM (yes, another one!) THAT NEEDS TO BE IN BY THE END OF THE MONTH! REMEMBER, YOUR VOTE COUNTS! SO DON'T DELAY IN RETURNING THIS FORM! The victims (?) are anxiously awaiting all the results - (probably hoping it's not going to be them!)

WE CAN'T WAIT FOR THE RESULTS! Look forward to the next issue so you can send a 'Sympathy Card' to the winners (or is it 'to the losers'?).

```
! >>>>>>>>>><<<<<<<<<< !
! SEE YOU AT THE MEETING !
!      16 APRIL 1988     !
!        2:00 PM         !
!       13 ELM WALK      !
! >>>>>>>>>><<<<<<<<<< !
!    NEXT SIG GROUP     !
!      30 APRIL 1988     !
!        2:30 PM         !
!       13 ELM WALK      !
! >>>>>>>>>><<<<<<<<<< !
```

Well, enough of Bytes & Pieces. let's get into the Newsletter! >>>>>>>>>>>>>>

NEWS ON THE EXCHANGE FRONT:
by: Bigfoot (again?)

>From Mid-South 99/4A Users Group (V6 #3 March '86): A new terminal program called TELCO has been released as Shareware by Charles Earl of Canada. Packed with features only found before in IBM compatible terminal programs! Truely an outstanding and remarkable program for modem users!

>BRUCE RYAN of RYTE/DATA< wrote in: "We're still in there struggling" and "A few problems remain but nothing insurmountable".

> Version 1.1 of< LEGENDS will soon be released by Asgard Software provided with information on updating Version 1.0 plus new features. (Some fixes supplied in their issue).

>According to MICROpendium< purchasers should stay away from Order 99 of 3512 Sun Lake Drive, St Charles, MD.

>Bud Mills Services of< 166 Dartmouth Drive, Toledo, OH 43614 HAS BOUGHT OUT HORIZON COMPUTER LIMITED. All orders for Horizon Ram Disks should now be sent to Bud Mills Services. John Johnson's 7.3 operating system and instructions are as follows:

96K SS/SD $140	384K DS/DD $225
192K DS/SD $165	512K $265
1 MEG $435.00	

Sizes and prices are for kits with all parts (you put together). If you wish the card to be put together for you, include another $60.00 in the price.

LEGENDS V 1.1 Fixes (Note these lines didn't match all of our program lines! jc):

```
>OLD DSK1.LGDN/TXT >ENTER

CHANGE LINE 1470 TO READ:

1470 CALL DP(22,"AN ICON.","READ I
TO Y OR N"): CALL KEY(V) :: ON V GOTO
1475,1905

CHANGE LINE 1475 FROM CALL C(1) TO:

1475 CALL C(16)... (didn't match)

>SAVE "DSK1.LGDN/TXT" >ENTER

>OLD DSK1.LGDN/MON > ENTER

CHANGE LINE 31 TO READ:

31 N,A,AF,P,V,D,DL,MX=0 :: H$,I$,C$
,ZL$

CHANGE LINE 1589 TO READ:

1589 IF MX=11...

CHANGE LINE 1935 TO READ:

1935 IF R<40 THEN...

CHANGE LINE 2700 TO READ:

2700 ...STR$(MX))...

>SAVE "DSK1.LGDN/MON" > ENTER
```

(Make sure the programs aren't protected before trying to save!)

NEWSLETTER EXCHANGES ARE STILL SHOWING PICASSO'S PUBLISHER AS FREEWARE!! PLEASE NOTE >>>THIS IS NOT SO<<<! It is a proprietary program copywritten by the author! (Please note our letter to Arto Heino hasn't been acknowledged to this date - jc)

>From HUNTER VALLEY 99'ers (March '88): GENIAL Computer's EPROM for the Horizon Ramdisk - A solution to the RAM based ROS unreliability or total ROS loss - this eprom provides an operating system very similar to Menu V6.3 except that it is locked up in ROM and cannot be damaged by bad data being sent to it. Eprom provides an extra 4k (16 sectors) data storage.

MULTI-FUNCTION CARD:

Peter Schubert advises: The new Multi-Function Card provides an advanced double density controller more suited to todays modern disk drives and software; provides a source of local RS232 Cards which have become scarce; memory expansion of 32k eliminates need for separate 32k; only one power supply is required; is buffered for extra reliability; all functions provided on one card to save space in PE Box.

2 years development has resulted in a number of peripherals being developed - notably the MINI-PE System, which is a stand alone version of this card, and the single chip 32k design now used as a kit for mounting within console. Cost expectations: A/$230.00 for disk control, A/$170.00 for RS232 only, and a maximum of A/$390.00 for all including 32k. A Multi-Card is also available for loan to TISHUG regional groups. Contact Peter on (02)3585602 or write:

P. Schubert
P. O. Box 28
Kings Cross 2011
N.S.W. Australia

>For FORTH USER'S! Clearing House initiated by a user - if you are even remotely interested in Forth, write to:

JOHN H. CARVER, JR.
#1, BOX 125-2
BRINGHURST, IN 46913

Or:

TI-FORTH INT'L INFORMATION CENTER
4127 GLENWAY STREET
WAUWATOSA, WI 53222

FORTH INTEREST GROUP
P. O. Box 8231
SAN JOSE, CA 95155
(membership $30/US annually

>>> E. N. D. <<<
(Who is this 'Bigfoot' anyway?)

INTRODUCTION TO THE GENEVE

by: BRYAN JONES

This article is a users view of the GENEVE. I am just a user. (Thank goodness we are not talking about drugs.) I don't program and I don't have any hardware experience. So I won't be able to address all possibilities, but only what I know from reading the manual or articles in newsletters or magazines. When I was thinking of buying a GENEVE I saw ads for a clock card ($109), cards that would give you 80 columns ($229), and ads to improve your video ($114). Then there was always the memory expansion of 128k ($100), 256k ($229) and even 512k ($239). As I have never (well, almost never) considered getting rid of my TI, I figured that I just might eventually buy a few of these options for my Christmas or birthday presents (sure beats socks and a tie). I also had thoughts of a GENEVE one day. It didn't take much to see that I would be better off to get it all together. Not only would it be cheaper overall but I'm sure we have all seen different pieces to the supposedly same puzzle that won't work together. And if I bought the GENEVE first I could make sure everything would be compatible from now on.

I received my GENEVE from Santa Claus Dec 87. It consisted of only a card (fits into the PEB), an IBM style keyboard, a few disks, and a binder full of directions. The first step was to use a cartridge save routine included on one disk to download my cartridges onto disk. All seems to be compatible with the Gramkracker download. I was only patient enough to download a few cartridges and still have not gone back to finish. After you have downloaded all of your cartridges you are through with the old TI-99/4A computer. You can put it away along with your cartridges. There is no cartridge port on the GENEVE. The next step is to remove the PEB controller card and 32K expansion card from the PEB along with any other memory expansion cards. This includes MYARC expansion cards that have not been modified for the GENEVE. After that, you insert your GENEVE card into the PEB and plug into it the IBM style keyboard, joystick and monitor. Incidentally, the GENEVE card is set for a composite or RF monitor and requires moving a jumper before using it on an RGB analog monitor.

When you turn on the PEB the GENEVE logo swan appears and the computer does a self test. I believe this is loaded from an 8K chip. Then the M-DOS will auto-boot. The M-DOS disk has one file that takes up the entire 358 sectors on an SSSD. After M-DOS is loaded you have access to several commands. These are basically disk manager type commands. You are able to format, copy (one pass), compare disks, set the date and time, rename, and something I haven't done yet is set volume labels. While there are other commands that you can use, the only program out (that I am aware of but don't have) that runs in M-DOS, at this date, is MY-ART. At this time the batch commands do not operate. There is an AUTOEXEC file that auto-boots (like load in extended basic) and can be used to set up the system (M-DOS only as far as I know). This can be used to assign DSK.2 to DSK.6 or to load your next program like MY-ART or the GPL. I usually go straight to the next disk and load GPL which appears to emulate the TI-99/4A right down to the master title screen.

I do have trouble answering the question of "What can you do now that I can't do with my TI?" It is not the GENEVE's fault that I do not have an analog monitor to have a nice clear view of 80 columns. Nor is it GENEVE's fault that I do not own a Myarc disk controller that would allow me to run MYARC's DM III that is included. As I said in the beginning I chose to buy the GENEVE first. I have 80 columns which I am using now in My-Word (TI-Writer revisited). I have a Ram-Card (not battery backed up). I have a "nice" IBM style keyboard without any cartridges crowding my right hand. I have successively down loaded my TI cartridges to disk and now I don't need to have to reset the computer because I don't have any cartridges to lock up. I even have Extended Basic on disk so that cause for lock-ups is also cured. The MY-WORD (TI-Writer) has more memory and a few more commands that I didn't have before like View File where you can look at another file without dumping the one in memory. Each time you save a file the time is annotated in the directory from the built-in clock. I can go straight to the formatter from the command line, print, and return without actually leaving the editor and having to reload. While in the formatter you can print your active file by typing "BUFFER" instead of a file name and access Show Directory by pressing FCTN 7. There is also more memory in Multiplan (41K vs. 15K). Everything I read tells me that I have 3 1/2 times the speed. Well, I do know that I ran a simple basic program on my TI to check for prime numbers that took 7 minutes 5 seconds to check up to 1000 for prime numbers. When I ran it on the GENEVE it took only 3 minutes 24 seconds. It didn't speed up any using the MYARC Extended Basic II, so I think the speed increase is just in computing power alone.

At present there is no speech capability and some assembly programs will not run due to the use of CRU scans. I believe that for maximum benefit you need to have a MYARC disk controller card (I am waiting for the DCC that will also run a hard disk - due out this year). The MYARC memory expansion cards will not work without modification. I have to admit that I don't use many of the programs that I have to see if they work OK or not.

The GENEVE is not now and never will be IBM compatible. The M-DOS only resembles MS-DOS which makes it easier for those who use MS-DOS at work.

In its' present state, the GENEVE may be a close call on whether it was worth the nearly $500 it cost but if the software follows, and I'm sure it will, it will be a bargain. I realize that many stories still exist on the GENEVE and whether it will ever be completely finished. Well, this article was written on MY-WORD which is 80 columns and "It works for me".

I would be happy to try and answer any questions or help in any way that I can. I would like to thank Richard Sierakowski for his assistance and Mike Dodd for his articles in MICROpendium magazine. They both have made learning about my new computer easier.

MINI-MEMORY Part VII
by ROBERT WORDSWORTH

In the examples given so far, you will notice I have always coded: LWPI >70B8 as the first executable instruction. I'll attempt to explain why.

You will remember that there are three ways in which we can execute machine code programs with the MiniMemory:

(a) from TI BASIC,
(b) from the MiniMemory "RUN" option,
(c) from Easybug.

So far, we have only used the last option. This is because certain extra steps have to be taken before (a) and (b) can be used. So to keep things as simple as possible, we've used only (c) for the time being.

Perhaps you will also remember certain features which set the TMS9900 somewhat apart from other rather more (dare I say it?) run-of-the-mill microprocessors. These are:

(a) It has only three "on chip" registers: the Program Counter, the Workspace Pointer and the Status Register. TI call these "hardware" registers.
(b) It has a large number (sixteen) of general-purpose registers. With minor exceptions any of these can be used for arithmetic, addressing or indexing. (We haven't yet met registers used for indexing.) There is no "accumulator" taking precedence over the others for arithmetic purposes, nor any dedicated index registers. These registers, which TI call "software" registers, are situated in CPU RAM, occupying sixteen consecutive words. This set of sixteen words is called the "Workspace". How do we know where it is exactly? Because the address of the first register in the Workspace, register 0, is held in the Workspace Pointer. We can, if we wish, decide where in CPU RAM we want our Workspace to be rather than use the default Workspace (see below). We do this by loading the Workspace Pointer with our chosen address by means of the LWPI instruction. This gives us a lot of flexibility since, should we need a new set of registers, we can easily create one by loading the Workspace Pointer with a new address, and later revert to the old set by reloading the Workspace Pointer with the original address. Although only one Workspace, the one currently addressed by the Workspace Pointer, can be active at any one time, the total number of registers is limited only by the amount of CPU RAM available to us.

When we call a machine code program from TI BASIC or the MiniMemory RUN option, our program will by default be given the "USRWSP" workspace at >70B8 in MiniMemory RAM. If, however, we run our program using the Easybug E(xecute) command, we will by default be using the same "GPLWS" workspace at >83E0 which Easybug itself uses. This is fine until we wish, after execution, to inspect the contents of the workspace, perhaps for debugging purposes, using the Easybug M(odify) command. Since

Easybug is using this GPLWS workspace for its own purposes, any data left there by our program will have been corrupted by the time we see the contents of the registers. The solution to this is to start our program with a LWPI instruction so that our program's workspace registers will not be the same as Easybug's. In fact we could use any RAM area for our workspace except for the whole area from >8370 to >83FF which Easybug uses, but as the USRWSP area at >70B8 is available to us we might as well use that. Having said all this, the only time we have so far inspected our workspace after program execution is with our first program which added register 1 to register 2!

As you become more knowledgable about writing in Assembly Language, you may want to start building up a set of general-purpose subroutines to perform various tasks such as writing to the screen, or performing jobs which are slow in TI BASIC (plenty of those!). The programs being presented here are not to be regarded as the foundation of such a set, although I hope they might give some pointers in that direction. The aim is merely to provide some new knowledge in as simple a way as possible and hopefully to give some fun as well.

The programs we've coded so far at least write messages to the screen: this will be of use in our next program which will introduce some screen scrolling. I said in the previous article that we'd be looking at something which couldn't be done in TI BASIC. This isn't strictly true since we could scroll the screen using CALL GCHAR and CALL HCHAR but it would be so slow as to be impracticable. In machine code, however, it is practicable.

We'll first of all write a program to scroll the screen up by one line. Not very exciting perhaps as we see enough of this in TI BASIC, but we'll write it so that it can easily be modified to scroll the screen downwards, too. Two new instructions are introduced: Add Immediate, AI, and Decrement, DEC.

In Immediate Instructions, also known as Format VIII instructions, the destination operand is coded first, unlike Format I instructions. The immediate operand is coded second. In the generated machine instruction, the value of the immediate operand is held as a word following the operation code, as can be seen in the Line-by-Line Assembler listing which follows.

The instruction: AI 0,32 for example, adds 32 to the contents of register 0. There is no "subtract immediate" instruction, but the same effect is achieved by coding a negative immediate operand. For example, the instruction: AI 0,-32 subtracts 32 from the contents of register 0. Other Format VIII instructions we have already met are Load Immediate, LI, and Load Workspace Pointer Immediate, LWPI. LWPI does not need to have a destination address coded since this can only be the Workspace Pointer Register.

The other new instruction, DEC, is the exact opposite of INC. It decrements the operand by 1. In "Single Address", or Format VI, instructions such as INC and DEC, the operand is nearly always a Workspace Register, but can be any form of "General Address", as discussed in the February Newsletter.

Another Assembler Directive we'll be using is BSS, Block Starting with Symbol. This simply reserves a given number of bytes without initializing them.

We will also use a new utility routine, VDP Multiple Byte Read, VMBR. This is the exact reverse of VMBWrite and uses registers 0, 1 and 2 in a similar way: register 0 holds the address in VDP RAM we are reading from, register 1 holds the address of the CPU RAM area we are reading into, while register 2 holds the number of bytes to read. The area of VDP RAM we are interested in is, of course, the part which holds the screen. The program we are about to code "wraps" the screen, that is, when we scroll up, the top line will reappear at the bottom. If this is not required, the part of the program that does this could be left out.

The program first reads the top line from the screen using VMBRead and stores it in a thirty-two byte "buffer", ready for writing back to the bottom line at the end of the scroll. It then starts a repetitive "loop" by reading the second line into another buffer and writing it back to the first line with VMBWrite. It then moves the third line to the second in the same way and so on until it has moved the twenty-fourth line to the twenty-third. At this point the loop finishes and the program moves the stored original first line to the twenty-fourth line. The loop is executed twenty-three times, not twenty-four, as the top line has to be treated differently from the others. We need something on the screen to scroll, of course. As we will be running the program from Easybug, we can achieve this by using the Easybug M(odify) command a few times, but we would also like to use our previously coded programs that write messages. Therefore we wish to start assembling our new program after the existing ones. If necessary, reload the MiniMemory from cassette with the Easybug L(oad) command and run the Line-by-Line Assembler using "OLD". The location counter (the hex number at the left-hand side of the screen) should be set to 7D66 - if it isn't, start with: AORG >7D66.

The rest of the program follows, in the format it will appear on the Line-by-line Assembler screen. As usual, the comments are entirely optional.

```
7D66 XXXX TL  BSS 32          Reserves 32 bytes to hold top line
7D86 XXXX ST  BSS 32          32 more for line read off screen
7DA6 02E0     LWPI >70B8      See above!
7DA8 70B8
7DAA 0200 SC  LI 0,0          VDP RAM address in Reg 0
7DAC 0000                     (row 1, col 1)
7DAE 0201     LI 1,TL         Address of buffer
7DB0 7D66                     for top line in Reg 1
7DB2 0202     LI 2,32         Number of bytes to read
7DB4 0020                     in Reg 2
7DB6 0420     BLWP @>6030     VMBRead top line into store
7DB8 6030
7DBA 0220     AI 0,32         Point to next row on screen
7DBC 0020
7DBE 0201     LI 1,ST         Address of buffer
```

```
7DC0 7D86                       for second and subsequent lines
7DC2 0209     LI 9,23           Number of lines to move
7DC4 0017                       in counter register 9
7DC6 0420 NL  BLWP @>6030       VMBRead next line
7DC8 6030
7DCA 0220     AI 0,-32          Point to line above
7DCC FFE0                       the one just read
7DCE 0420     BLWP @>6028       VDPWrite line in store to screen
7DD0 6028
7DD2 0220     AI 0,64           Point to next line to be read
7DD4 0040
7DD6 0609     DEC 9             Count down to zero
7DD8 15F6     JGT NL            Loop back if still positive
7DDA 0220     AI 0,-32          Point to bottom line
7DDC FFE0
7DDE 0201     LI 1,TL           Point to stored top line
7DE0 7D66
7DE2 0420     BLWP @>6028       VMBWrite bottom line
7DE4 6028
7DE6 045B     B *11             Return to Easybug
7DE8 XXXX     SYM
RESOLVED REFERENCES
MS-7D1C  NX-7D0C  RT-7D1A  M1-7D3E
TL-7D66  ST-7D86  NL-7DC6

7DE8 XXXX     END
0000 UNRESOLVED REFERENCES
```

Now save the whole of MiniMemory RAM to cassette using the Easybug S(ave) command. When Saving, it's always best to save the whole of the RAM by saving from 7000 to 7FFF.

You should by now have quite a few cassette-handling instructions on the screen. Run the program in the normal way with the Easybug E(xecute) command, the program entry point being 7DA6. You should see the whole screen shift up one line with the top line, which may be blank, reappearing at the bottom. Repeat E7DA6 as many times as you need to convince yourself that the program is working. Note that Easybug itself causes a line feed before obeying a command, which complicates matters somewhat and will cause some lines to be "lost".

To obtain a more spectacular effect let's change the "B " to a "JMP SC". This will have the effect of causing endless repetition of the scrolling, which is quite spectacular in our modest terms, though fairly useless! We could go back to the Assembler and change the instruction, but it's quicker to use Easybug and we won't lose all our screen. If you have keyed in the program exactly as listed above, the Easybug commands for making the change are:

M7DE6 04 -> 10
M7DE7 5B -> E1

This has the effect of substituting the machine-code instruction "10E1" which is the equivalent of "JMP SC" for the "045B" which corresponds to "B $11". Note that for this to work, the addresses and coding must be exactly as shown. Now run the program from Easybug by the command E7DA6.

The only way of stopping the program is to turn the computer off and on again! So do just that. Reenter Easybug. We are going to modify the program to scroll the top half of the screen only. To do this, note that the word at 7DC4 contains >0017, decimal 23, the number of lines to scroll minus one. To scroll only the top half of the screen we want to set this word to eleven. This is wholly contained within the low-order byte at >7DC5, so enter: M7DC5 17 -> 0B

At this point you probably won't have much on the screen, so use Easybug to run the two screen-message programs a few times by entering

```
E7D00
E7D50
```

as often as you like. Then start the scrolling program, as before, with: E7DA6

There are a few more Easybug modifications we can make before we're done. The word at >7DAC is the immediate operand of the Load Immediate instruction used to load register 0 with the starting address at which we wish to start scrolling. At the moment, it contains zero, indicating that we wish to start scrolling at the top of the screen. If we change this to 32 x 6 = 192, we will start scrolling from line seven rather than line one. We will still only be scrolling half the screen as we have left the word at >7DC4 as it was. Modify the value by M7DAD 00 -> C0.

Finally, how about scrolling upwards? To do this we proceed as for scrolling downwards, but reversing each step. So instead of storing the top line and writing it back to the bottom, we store the bottom line and write it back to the top. At >7DBA, instead of pointing to the next row on the screen, we point to the previous screen. At >7DCA, instead of pointing to the line above the one just read, we point to the line below the one just read, and so on. The upshot of all this is that we need to reverse the signs of most of the words containing 32 or 64. This is easily done using "OLD" and the following directives.

```
AORG >7DBC
DATA -32
AORG >7DCC
DATA 32
AORG >7DD4
DATA -64
AORG >7DDC
DATA 32
END
```

Again use Easybug to run the two screen-message programs a few times by entering

```
E7D00
E7D50
```

to get some more data on the screen, and run the program with E7DA6 to see some downwards scrolling.

Routines to handle such functions as scrolling are obviously useful called from a BASIC program. So far, all our programs have been run from Easybug. Next step will be to investigate alternative ways of running our programs and, naturally, sideways screen scrolling!

>>>>> E N D <<<<<

WILL YOU LEAVE THAT DARN COMPUTER ALONE AND TAKE ME TO TESCO'S!

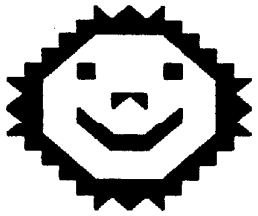
Ya' Know. Sometimes she really 'BUGS' me!

Courtesy of Derek Allen

For Big & Little Kids ...

For some programming fun, try typing this in... You should get the picture shown below (working with a Gemini 10X printer, that is). Change Line 130 to read "HAPPY EASTER" or "HAPPY BIRTHDAY", or choose to suit your needs. Have fun!

```
10 OPEN #1:"PIO"
20 PRINT #1:CHR$(27);"U";CHR$(1)
30 PRINT #1:CHR$(27);"A";CHR$(6)
40 PRINT #1:CHR$(27);"G"
50 FOR J=1 TO 14
60 FOR I=1 TO 14
70 READ A
80 PRINT #1:CHR$(A);
90 NEXT I
100 PRINT #1:
110 NEXT J
120 PRINT #1:CHR$(27);"@"
130 PRINT #1:CHR$(15);"     HAVE A NICE DAY"
140 PRINT #1:CHR$(27);"@"
150 END
160 DATA 224,224,224,224,224,254,252,254,252,224,224,224,224,224
170 DATA 224,224,224,224,239,239,239,239,239,239,224,224,224,224
180 DATA 224,224,224,239,251,224,224,224,224,253,239,224,224,224
190 DATA 224,224,239,251,224,224,224,224,224,224,253,239,224,224
200 DATA 224,239,251,224,239,224,224,224,224,239,224,253,239,224
210 DATA 253,239,224,224,224,224,224,224,224,224,224,224,239,251
220 DATA 252,239,224,224,224,224,251,253,224,224,224,224,239,254
230 DATA 253,239,224,224,254,224,224,224,224,252,224,224,239,251
240 DATA 252,239,224,224,253,254,224,224,252,251,224,224,239,254
250 DATA 224,239,254,224,224,253,239,239,251,224,224,252,239,224
260 DATA 224,224,239,254,224,224,224,224,224,224,252,239,224,224
270 DATA 224,224,224,239,254,224,224,224,224,252,239,224,224,224
280 DATA 224,224,224,224,239,239,239,239,239,239,224,224,224,224
290 DATA 224,224,224,224,224,251,253,251,253,224,224,224,224,224
```



HAVE A NICE DAY

TI-WRITER TABS
by: PETER WALKER

Most people who use the TI-Writer are probably aware that the Editor Tab settings are stored along with the text. Tab settings comprise Left and Right hand margins together with up to 15 Tab 'stops' which may include an Indent position for paragraph beginnings. How this information is stored is the subject of this article.

The Tab record is the last record in a TIW file. It's not easy to inspect its structure since TIW obviously 'absorbs' it, and the Assembler Editor can't read it either as it always strips out what it considers to be 'control characters'. In fact the Tab record uses characters in the range 128 to 213, which lie above the regular ASCII characters 0 to 127 which the TIW is designed to manipulate. To study the Tab record structure you have to read the end record using a purpose written program or one of the many sector editors that allow one to read directly what is on a disk.

The Tab record is a 22 byte record made up as follows. As usual, I indicate HEX equivalents by the ">" prefix.

BYTE 0	The length byte 22 (>16)
BYTE 1	128 (>80)
BYTE 2	134 + Left Margin (>86 + L)
BYTE 3	128 (>80)
BYTE 4	134 + Right Margin (>86 + R)
BYTES 5 - 20	Up to 16 Tab settings in the form 134+T (>86+T). These are stored in order from Left to Right and should include both the Left Hand Margin Setting and the Indent Setting in their proper place. Unused Tab stops are set to correspond to Tab Settings of 79, ie: 134+79=213 (>D5).
BYTE 21	128 (>80)
BYTE 22	134 + Indent Position (>86 + I)

The bytes at 2,4 and 22 define where the L,R and I positions will be. The bytes from 5 to 20 define where the cursor will stop when the TAB key is pressed during editing.

For those looking at this record on a sector editor, you will, of course, note that since this is the last record on the TIW file this 22 byte record is immediately followed by the End Marker (>FF).

To clarify the above let's look at the following example. The Left Margin is 10, the Indent is at 5 ('outdenting'), Tabs are at 15 and 20, and the Right Hand Margin is at 75. The resulting record looks like this in HEX:

```
16 80 90 80 D1 8B 90 95 9A D5 D5 D5 D5 D5 D5 D5 D5 D5 D5 D5 80 8B

      LM    RM I  LM T  T  T=79 repeated                      I
```

Why are the settings offset from 134 rather than 128? It may have something to do with the fact that column 0 is offset 6 from the left of the screen to allow for the line numbers on the immediate left. Poking left margin values between 128 and 134 will indeed set the cursor outside the editing frame - over the numbers - but produces bizarre and unpredictable results!

Indeed, any upsetting of this strict format produces a range of unpredictable results, including crashes during loading and editing at worst, and almost always incorrect Tab positioning.

I have produced the ExBasic program below which will append a TIW tab record to a DIS/VAR 80 ASCII file. If it is used on an existing TIW file, this second Tab record appears to replace the first. I can't think of any practical use for this program, but it neatly demonstrates the structure I have outlined above.

```
100 DIM TB(79)
110 TN=1
120 DISPLAY AT(1,1)ERASE ALL
:"APPEND TI WRITER TABS" ::
DISPLAY AT(3,1):"WHAT FILE?"
130 ACCEPT AT(3,12)BEEP:F$ :
: IF SEG$(F$,1,3)<>"DSK" THE
N 130
140 ON ERROR 340
150 OPEN #1:F$,DISPLAY,VARIA
BLE 80,APPEND
160 DISPLAY AT(5,1):"LEFT MA
RGIN": :"RIGHT MARGIN": :"IN
DENT": :"TAB": :"ENTER TAB=7
9 WHEN FINISHED"
170 ACCEPT AT(5,14)SIZE(2)BE
EP VALIDATE(NUMERIC):L :: IF
 L>79 OR L<0 THEN 170
180 TB(L)=1
190 ACCEPT AT(7,14)SIZE(2)BE
EP VALIDATE(NUMERIC):R :: IF
 R>79 OR R<L THEN 190
200 DISPLAY AT(9,13):L
210 ACCEPT AT(9,14)SIZE(-2)B
EEP VALIDATE(NUMERIC):I :: I
F I<0 OR I>R THEN 210
220 TB(I)=1 :: IF I<>L THEN
TN=2
230 DISPLAY AT(11,4):TN
240 ACCEPT AT(11,14)SIZE(2)B
EEP VALIDATE(NUMERIC):T :: I
F T<1 OR T>79 THEN 240
250 TB(T)=1 :: TN=TN+1 :: IF
 TN=15 OR T=79 THEN 260 ELSE
 230
260 DISPLAY AT(15,1):"CREATI
NG TABS"
270 A$=CHR$(128)&CHR$(134+L)
&CHR$(128)&CHR$(134+R)
280 FOR J=0 TO 79
290 IF TB(J)=1 THEN A$=A$&CH
R$(134+J)
300 NEXT J
310 A$=A$&RPT$(CHR$(213),20-
LEN(A$))&CHR$(128)&CHR$(134+
I)
320 PRINT #1:A$ :: CLOSE #1
330 STOP
340 ON ERROR STOP
350 CALL ERR(C,T,S,L)
360 IF L=150 THEN RETURN 130
370 PRINT C;T;S;L
```

Incidentally, while on the subject of characters above 127, it should be clear that the TIW Editor can't handle characters 128 to 255. (Though I understand that the very rare animal TIW Ver2 does so in order to handle accented characters in foreign languages. Its Tab structure is said to differ from that which I describe here and this can cause Ver1 and Ver2 files to be incompatible).

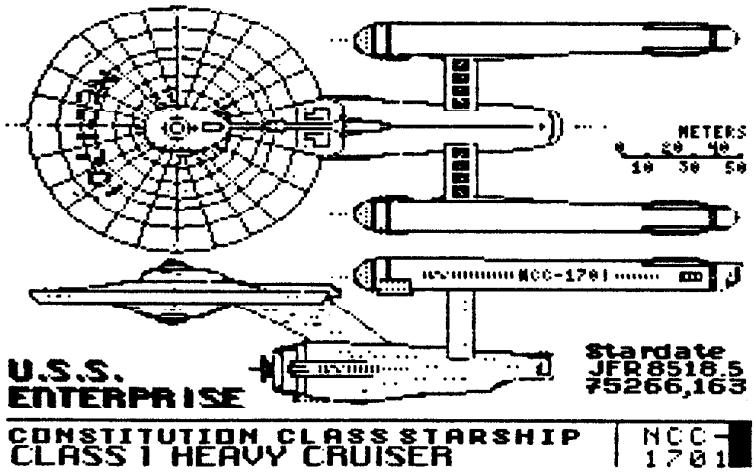
However, the Formatter can handle all character values. It is possible to use the Transliterate function to replace one or more normal characters with ones in the upper range. This is extremely useful for those printers (eg: DEC or IBM standards) which use the upper range for a series of multi-national and graphics characters. For example, on my Brother EP44 typewriter the e acute character is produced by byte 130, so I can transliterate the $ sign to é using: .TL 36:130.

Once you've got that figured out, it is quite possible to transliterate a whole series of characters and get the Formatter to build its own Tab record for the Editor to reread!

[Thank goodness it's the Editor in the TIW and <u>not</u> the Editor of the Newsletter! Sorry, I couldn't resist the temptation... jo]

>>>>> E N D <<<<<

'88 MODELS NOW READY FOR TEST DRIVE
URBAN CYCLE - 3 PARSECS PER
DI-LITHIUM CRYSTAL!
LEAVE THE KLINGONS AT THE LIGHTS!



(Courtesy of Derek Allen)

FOR SALE FOR SALE

ITEM	£ £ £
TI-99/4A NTSC Console	10.00
TI-99/4A PAL Console w/modulator	10.00
Peripheral Expansion Box	50.00
TI Disk Controller Card w/Disk Manager II	50.00
Double-Sided Disk Drive	-SOLD
32K Memory Expansion Card	40.00
RS232 Interface Card	-SOLD
Milton Bradley MBX System	15.00
Sound Track Trolley Module	2.25
Space Bandits Module	2.25
SewerMania Module	2.25
SuperFly Module	2.25
Championship Baseball Module	2.25
I'm Hiding Module	2.25
TI LOGO /Complete	-SOLD
TI Editor/Assembler /Complete	-SOLD
TI Writer Word Processor /Complete	-SOLD
TI Terminal Emulator /Complete	10.00
TI Speech Editor	5.00
TI Extended Basic Module	15.00
TI Speech Synthesizer	15.00
TI Mini-Memory Module /Complete	15.00
MicroSurgeon	2.25
M*A*S*H	2.25
Slymoids	2.25
TI Invaders	2.25
BurgerTime	2.25
Buck Rogers	2.25
Parsec	2.25
Star-Trek	2.25
Fathom	2.25
MunchMobile	2.25
Super Demon Attack	2.25
Video Games 1	2.25
Pac Man	2.25
Ms Pac Man	2.25
Donkey Kong	2.25
Pole Position	2.25
Shamus	2.25
Defender	2.25

Contact: Dennis Hemmings tel: 0626-56869 (evenings)
17 Bowden Hill
Newton Abbot
Devon TQ12 1BH

Dual Cassette Leads * 2.00
* Offers to: EAR 99'ers User's Group

TREASURY REPORT:

MONTHLY BEGINNING BALANCE.....		$ 383.80+
Due Fairware Authors	$ 40.97	
ASSETS (INCOME):		
Library Tapes and Disks	$ 93.15	
Subscriptions Income	$ 34.60	
Modem Account	$ 89.22	
Joysticks/Cassette Leads	$ <27.00>	
Assets Sub-Total.....		$ 189.97
LIABILITIES (EXPENSES):		
Payment to Arto Heino (Picasso's Publisher)	$ 40.97	
Postage/Stamps	$ 43.51	
Stationery Supplies Expense	$ 12.39	
Bank Service Charges	$ 4.00	
Liabilities Sub-Total.....		$ 100.87
ENDING MONTHLY BALANCE.....		$ 431.93+

Exchange Rate at $1.85/$1.90=£1.00

From the Pres:

THANKS TO EVERYONE WHO CONTRIBUTED ARTICLES TO THE NEWSLETTER THIS MONTH! In fact, the participation was so good JoAnn had to move some articles to next month's issue!

We hope everyone who attended Bloxwich had a good time and we hope to see a review of it in the next newsletter.

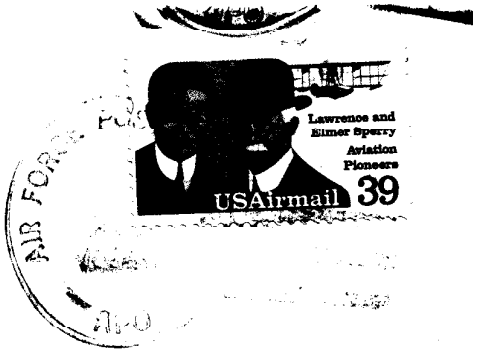
We'd like to have a ONE-YEAR-OLD BIRTHDAY PARTY in MAY! We'd like your thoughts on party ideas as soon as possible, and possibly hold it along the usual lines of 'bring your own recipe' as a contribution idea.

Our Modem is up and running and we've been enjoying 'playing' with it. The electronic mail idea is great and for the cost of a local call you can leave messages to people who would normally be a long-distance call. Our Prestel E-Mail Number is 219999131 for anyone interested. If you wish to obtain your own Modem free of charge contact me for a demo and/or information on how to join Prestel and obtain the free Modem!

NEXT MEETING: 16 APRIL 1988 at 2:00 PM
NEXT SIG GROUP: 30 APRIL 1988 at 2:30 PM
Hope to see you there!

Buffer Full . . . Scott Copeland, President

SSGT DONALD S. COPELAND
% EAST ANGLIA REGION 99'ers USER'S GROUP
1979CS/LGG
PCS BOX 5927
APO NEW YORK 09179-5379



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