HUNTER VALLEY 99ERS USERS GROUP
HOME COMPUTER NEWSLETTER
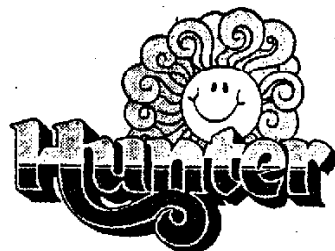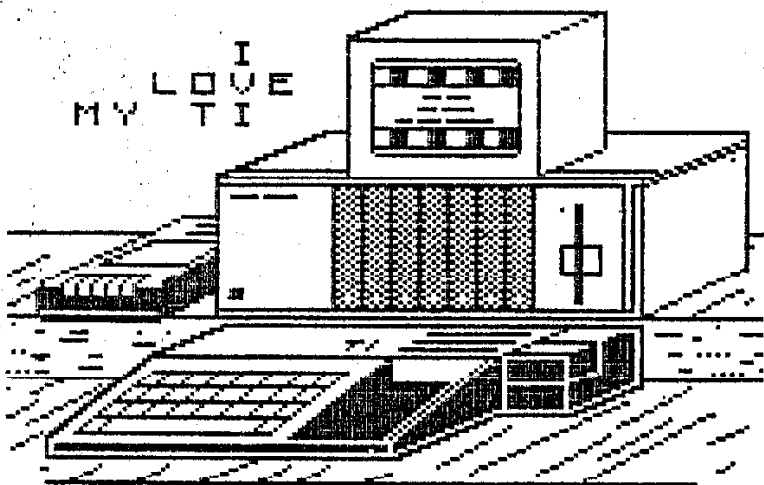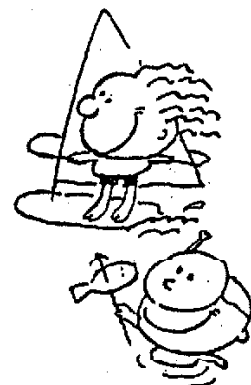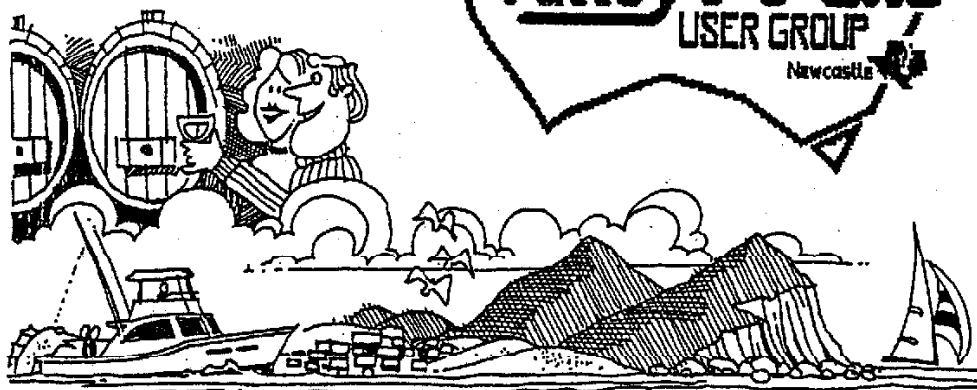
I LOVE MY TI

Hunter

AUGUST 1988

Hunter Valley '99ERS USER GROUP Newcastle

YOUR COMMITTEE

**all telephone numbers
are STD area code 049**

PRESIDENT
Alan Lawrence
35 Bayview St.,
WARNERS BAY 2282
Ph. 486509

SECRETARY
Albert Anderson
6 Arcot Close,
TARRO 2322
Ph. 662602
Viatel 496626020

TREASURER
Peter Smith
8 Glebe St.,
EAST MAITLAND 2322
Ph. 336164
Viatel 493361640

SOFTWARE LIBRARIAN
John Paton
1 Parlen Close,
RUTHERFORD 2320
Ph. 326014
Viatel 493260140

PUBLICATIONS LIBRARIAN
Allen (Joe) Wright
77 Andrew Rd.,
VALENTINE 2280
Ph. 468120

EDITOR
Brian Woods
9 Thirlmere Pde.,
TARRO 2322
Ph. 662307
Viatel 496623070

PURCHASING CO-ORDINATOR
Alan Franks
822 Pacific Highway
MARKS POINT 2280
Ph. 459170

COMMITTEE MEMBERS
Noel Cavanagh
378 Morpeth Road
MORPETH 2321
Ph. 333764

Rodney Gainsford
56 Sedgewick Ave.
EDGEWORTH 2285
Ph. 583515

CONTRIBUTIONS

Members and non members are invited to contribute articles for publication in HV99 NEWS.

Any copy intended for publication may be typed, hand written, or submitted on tape/disc media as files suitable for use with TI Writer (ie. DIS/FIX 80 or DIS/VAR 80). A suitable Public Domain word processor program will be supplied if required by the club librarian.

Please include along with your article sufficient information to enable the file to be read by the Editor eg. File Name etc. The preferred format is 35 columns and page length 66 lines, right justified.

All articles printed in HV99 NEWS (unless notified otherwise) are considered to be Public Domain. Other user groups wishing to reproduce material from HV99 NEWS may feel free to do so as long as the source and author are recognised.

Articles for publication can be submitted to the Editor, ALL other club related correspondence should be addressed to The Secretary.

DISCLAIMER

S

SECRETARYS REPORT

FROM ALBERT ANDERSON

This month of August brings us some sad news from our friend in Tasmania, Rex Shephard. Rex has let us know that Bill Knecht from the Houston Users Group in Texas passed away on the 9th July. Bill provided TI users with a host of software and was well known for his writings, particularly in the field of music for the 4A, which appeared in many newsletters around the the USA. A sad loss indeed... we at HV99 would like to express our thanks to Bill and pass on our sympathy to Bill Knecht's family and friends.

At the office membership renewals continue to come in and thank you to the devotees that have continued into 1989 with us. Those that as yet have not renewed will find a notice on their newsletter informing them that this will be their LAST ISSUE !! If you fall into this category and wish to renew for '89 please attend to it SOON.

This month sees the general meeting shift location for the first time in about 2 years. This has come about in response to members suggestions and we hope that the Argenton venue will be more suited to our needs. Again we need feedback on it. Please tell us what you think or put forward alternate ideas as to a venue. While you're still thinking you can also consider the need for HV99 to become an incorporated body. If you know anything of the details in the process of incorporation

could you please let the committee know as we must look at this very soon.

Would the local members like to do a little easy advertising for the group? Does anybody go to the local supermarkets to shop every now and again? Ever noticed the community notice stands that people hang around while they are waiting for one thing or another? Well how about you grab a pen and fill in one of the community activity notices and let the inquisitive know that HV99 supports TI-99/4a users.... great idea!!! thanks Noel Cavanagh for this one.

The Australian scene this month has been rather quiet except for the good news that the TIUP group from Western Australia is back in the swing and has produced its first newsletter for quite a while. Good luck for the future to our friends in the west.

From the Chicago TI-User group in the USA. comes a long awaited response on HV99s newsletter exchange from Nick Iacovelli Jnr. Thanks to the Chicago group and we look forward to more contact from you in the near future.

There is more news on the overseas scene from Joe Wright elsewhere in this issue but as for me well that all I've got... back next month.

Albert Anderson
(4a4me)
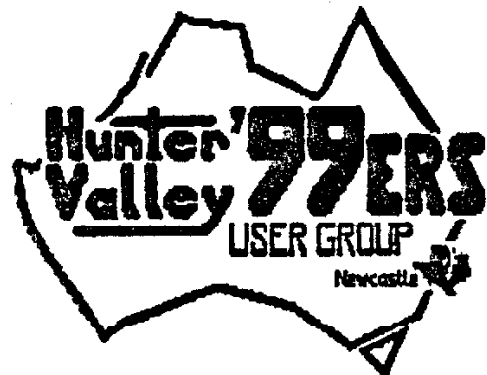
IN THE NEWS

A POT POURRI OF LOCAL
AND INTERNATIONAL NEWS
COMPILED BY

joe wright

Ah! August! it signals the beginning
of the end of winter. Although, in
all fairness, it must be said that
this winter has thus far been
extremely mild.

Now for the quotation for this
month;

The rule in carving holds good as to
criticism - never cut with a knife
what you can cut with a spoon.

Charles Buxton.

MEMORY WITH VARIED OPTIONS.
****
Bob Carmany a HV99'ers member and
prolific writer has an article in
the June Micropendium in which he
reviews a stand alone 32k expansion.

Bob had a 32k expansion roasted when
a console crashed, the 32k
replacement he purchased is the
subject of that review. I have
extracted sections of the review
below.

"Once the initial panic subsided, I
began to search for a replacement.
In my wanderings through volumes of
advertising literature. I came

cross an outfit called "The
Captain's Wheel". Among other
things they advertised was a 32k
standalone with some intriguing
options.

Basically, three options were
offered.
1)Duplicate any 8k block of memory,
2)Add up to three additional blocks
counting the software loader as a
block.
3)Order a loader to allow the
transfer of code from disk to tape
or visa versa.

Each option was US$10. So, for a
32k with all three options (two
banks of memory and the loader
software) you would only have to
come up with US$79. That compares
quite favorable with the CorComp 32k
standalone for US$99.

I didn't order the fully expanded
32k although I now wish I had
ordered another bank of memory. I
did however, order the 32k with an
optional bank of memory at >6000- a
32k and 8k"supercart" all in one
unit!."

"The standalone 32k was everything
that I had hoped for! In fact, with
the extra bank at >6000 (the GROM
port), it exceeded my expectations
by a good measure."

"The real eye opener came when I
decided to exercise the optional
bank of memory. The first thing I
tried it on was Funnelweb."

"The bank at >6000 and Funnelweb
made an amazing pair."

"The 32k simply plugs into the
console. It has to be immediately
next to it in line, because it draws
it's power from the console."

"Despite the sometimes brief
documentation, the over-all
performance and ease of use of the
Captain's Wheel 32k makes it a real
winner! With the requisite hardware
and the "Fairware" Funnelweb
package, you can load programmes
that, quite frankly, I had never
seen before."

"The only drawback I have found with
the optional memory bank at >6000 is
that it leads you to want more
programmes you can use with the
Captain's Wheel 32k."

Manufacturer:

THE CAPTAIN'S WHEEL
17295 Chippendale Ave.,
Farmington,
MN 55024
U.S.A.

Bob also mentions that an excellent source of programmes to use with the bank of memory at >6000 is;

David R. Rommer
P.O. Box 554,
Walbridge,
OH 43465
U.S.A.

GENEALOGY.
The amazing thing about Genealogy is where and how information can spring from. Mr. Paul and Mrs Beryl Thomson of Canberra are a good case in point. The Thomsons contacted me to obtain a copy of my Genealogy Record Keeper simply to "see" what it was like. Our librarian sent Genealogy and also FUNL PLUS as requested. I have since had many telephone conversations with the Thomsons. They are using a Genealogy Record keeper written in extended basic by a member of the Canberra group Mr. Don Gilchrist. The programme has been developed over a four year period by Gilchrist making modifications as the Thomsons needed additional features. The programme is used to store large numbers of records mainly from the Yorkshire Region of England. It has many search and sorting features which make it ideal for large amounts of data to be stored. A review of this programme was featured in the Canberra newsletter. A copy of the review is on it's way to me and I hope to have it in the september HV News. I also understand that the programme "Newfile" has the ability to create files for transfer via modem.

The Thomsons maintain these large records at their own cost. They maybe able to help if you have hit a brick wall for the Yorkshire County. They are very kind people and have since helped me with where to obtain information and how to get it. They have given me permission to mention them and how they assisted me in this column. If you do write to them PLEASE do so only if you are stuck and as a minimum, enclose a self addressed and stamped envelope for return mail. Make it large enough to hold at least several sheets of paper.

Mr. Mrs. P. Thomson
114 Learmonth Dve.,
KANBAH   2902
A.C.T.
AUSTRALIA
Phone 062-317-338

SECOND HAND P.E. BOXES.
If you read the article in the HV99 Newsletter last month regarding expansion of your system here is a bit of additional information. I mentioned that the cost of a second hand P.E. BOX should be around $600 to $800. I also mentioned that you should speak to our wheeler and dealer before making a purchase. You could be surprised at what sort of deal you might be able to dig out by looking around. That is, those prices are not set in cast iron.

So before you dive into the deep end of the pool see the wheeler and dealer in second hand equipment. (A wink is as good as a nod to a blind horse!)

LINE HUNTER.
The latest release of Funnelweb has Wills McGovern's LINE HUNTER on it. This programme is a must for the assembly language programmer. Especially if you are like me and have been executed many time using the assembler off Funnelweb. For those who don't know what I am talking about, it is a programme that will find a particular line or label in source code. Makes finding one's programming slips very easy. It is particularly useful where the COPY directive is used. On top of all that it operates very quickly. It brings the offending code to screen, indicating the filename and line number within that file. It also makes an interesting demo!!

MODULES.
TEX COMP are still running advertisements in Micropendium for modules, cassette and disk based software. Some prices are given below, these are by no means all those available.

```
MODULES........
     T.I. INVADERS.....$4.95
     MUNCHMAN..........$4.95
     PARSEC............$4.95
     HANGMAN...........$6.95
     MOONSWEEPER.......$9.95
     BUCK ROGERS......$15.95
     ADVENTURE MODULE +
        PIRATE DISK.....$6.95
     EARLY LEARN FUN...$4.95
     MUSIC MAKER.......$9.95
     TI LOGO II.......$19.95
     READING FUN.......$9.95
     READING ON........$9.95
     READING FLIGHT....$9.95
     ADDIT & SUBTRACT..$9.95
     MULTIPLICATION....$9.95
     DIVISION..........$9.95
     MILLIKEN SUBTR....$9.95
     MILLIKEN MULTI....$9.95
     MILLIKEN EQUAT....$4.95
     FACE MAKER........$9.95
     STORY MACHINE.....$9.95
     TERMINAL EMULATOR.$9.95
     PERS REPORT GEN..$10.95
     MULTIPLAN........$24.95


DISKETTE SOFTWARE.....
     GRAPHING PACKAGE..$4.95
     ELECTR ENGINEER...$4.95
     MUSIC MAKER DEMO..$4.95
     CASH MANAGEMENT...$4.95
```

TEX COMP also have their 1988 catalogue available for $2:00

```
     TEX COMP
     P.O. BOX 33064
     Granada Hills,
     CA 91344.
     U.S.A.
```

All the above prices are in US dollars. Visit to library corner at the meetings to check out these and other goodies advertised in Micropendium.

DESK TOP PUBLISHER.
This cartridge is reviewed in the June Micropendium by Ron Prewett.

"Desk Top Publisher is a cartridge programme produced by DataBioTics that allows the user to create a graphic picture and then include the picture in text. The text can be printed in one to three columns with an Epson compatible printer.

The cartridge can be used with just a console and cassette recorder. Expanded memory is not required nor

are other peripherals except for an RS232 interface and printer."

"The programme consists of three major sections that are selected from the main menu. These are PICTURE MAKER, WORD MAKER and PRINT PAGE."

"PICTURE MAKER is a graphics or drawing programme that has many of the drawing functions of other graphics programmes like TI-Artist and Graphx."

"WORD MAKER is the text input programme. You will first be asked to choose 1,2 or 3 columns for inputting your text. Choosing 1,2 or 3 columns will allow you 78,39 or 26 characters per line respectively."

"PRINT PAGE section is pretty straightforward. it allows input of printer device (the default is "PIO.CR") and whether to include the picture in the printed output."

"Value is greater for those with an unexpanded system. It is a minimal text processor that allows you to prepare your text in one, two or three columns."

```
          DataBioTics,
          P.O. Box 1194,
          Palos Verdes Estate,
          CA 90274
          U.S.A.

Price.........US$69.95.
```


---RUMOURS---RUMOURS---RUMOURS---

The rumours department has been very quite of late. One Myarc hard disk controller has been ordered by one of our members and I am certain that another will not be far behind.

Also heard that the software librarian and his gang have been burning the mid night oil getting the completed library contents onto a well known data base.

Neil Quigg's Ramdisk board is just awaiting orders to come in for an initial order of P.C. boards to be finalised. Me thinks that the board ready to accept memory chips should come out at about $60/70. me also thinks that is without sockets for the memory chips.

RANDOM BYTES

from
BOB CARMANY

Before we get started with some "goodies", a bit of a warning. Information from the Ottawa UG from the newsletter:

"Most importantly of all this month, we would like everyone to know that the Ottawa Users' Group recommends that anyone using DM1000 3.6 or higher go back to 3.5. It is the most stable version at the moment, and later versions contain potentially serious bugs. We apologize to anyone who may have experienced them, and hope that they will enjoy using 3.5 until our next release. Version 4.0, for example, is an escaped beta copy and should not be used under ANY circumstances. We hope to have a new version (thoroughly beta-tested) in the fall.

Summer's end here in our hemisphere and the beginning of spring "down under". As usual, there are gardening chores left undone and other "honey do's" that should have been done. But, the TI beckons and you know what is going to get done first!

Over the past few months, I have been sending the latest terminal programs for inclusion in your UG library. There are several but notably MASSTRANSFER 4.3 and TELCO 2.1. Right now, though, there is simply no way to use them on VIATEL. So, what can you do with a program that is geared for a 300/1200 baud system? We will explore that in just a minute but first a word about terminal programs in general.

What does all of this baud rate business mean anyway? How may characters are being transmitted? Let's compare it to something that almost everyone should be familiar with --- your printer. Printers are rated by the number of characters

that they can print per second. For example, my NP-10 prints 100 CPS (characters per second) in draft mode. Baud rate, on the other hand is BITS per second. Without going into the details of data transmission, each character takes a total of 10 bits to send. There are the actual character bits, a stop bit and a parity bit sent for each character. That means that VIATEL receives at a sluggish 7.5 CPS (75 baud) and transmits at 120 CPS (1200 baud). So, there are some printers that can actually keep up with a 1200 baud modem transmission!

Okay, that is out of the way. Now, what can you do with one of these terminal programs that I have sent for inclusion into our UG library. Even though VIATEL operates at 75/1200 baud, there are other uses for an unmodified version of MASSTRANSFER or TELCO. You can exchange files with any other UG member who has a modem by using one of these programs. With an auto-dial and auto-answer pair of modems, you could ring up Larry Reid in Queensland, for example, and swap programs back and forth at 1200 baud. Something to think about!

I ran across something the other day that I knew about (that's why I bought it) but I don't know how many other people do. When TI started making the 99/4A they started with the black and silver consoles that would run just about anything. Third-party software packages like CENTIPEDE, PAC-MAN, etc. came out and ran quite well on it. In 1983, they came out with Vn 2.2 of the operating system. These consoles are the familiar beige jobs and display "VN 2.2" when the title screen comes up. They had the advantage of having a higher capacity (and cooler running) power supply and improved heat sinks to dissipate the "coffee warmer" heat that the old black and silver consoles produced. But with Vn 2.2 came a "surprise". The system had been altered so that these third-party software packages would no longer run. You had to buy an adapter from the likes of TEX-COMP to run CENTIPEDE and PAC-MAN.

However, between the time that the black and silver console was discontinued and the Vn 2.2 was

produced, there was a "transition" console produced -- a beige NON-Vn 2.2 console. This console has the high capacity power supply and the "guts" of the old black and silver console. I happened to run across one and immediately added it to my inventory! It runs cooler and will run any program for the TI!

Oh yes, if you want to know when your console was produced, turn it over and look at the "LATA" number on the bottom. The first two digits are the week and the last two are the year. So, LATA 4283 becomes the 42nd week of 1983 -- that's when my beige NON-2.2 TI was produced. Incidently, the black and silver that I was using was produced the week before in 1983.

Oh yes, if any of you haven't gotten around to ordering a QED cartridge, you are missing "the best thing since sliced bread". I have had one for several months and it has replaced XB as my most-used cartridge. With F'WEB 4.1 installed, there isn't much that you can't do with it. Ron Kleinschafer has done an outstanding job with his version 3.0 of the QED program.

As you can see, this month I took a bit of a break in the technical (and non-technical) TI information for a bit of a rambling dissertation. Next month, we will get back to some "goodies".
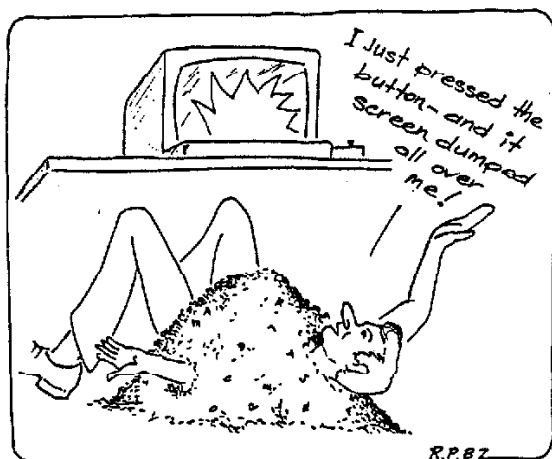
I just pressed the button- and the screen dumped all over me!
R.P.82

RAFFLE

The recently drawn MULTIPLAN RAFFLE was won by none other than our own 'Soldering Iron' Bob MacClure. Congratulations Bob, and thanks to all those who purchased tickets in the draw. Thanks again also to Steve Taylor for so graciously donating the MULTIPLAN package to the group for the raffle...your gesture was very much appreciated by the club.

SOFTWARE

Don't forget the big SOFTWARE OPEN NIGHT on Tuesday, 16th August at Warners Bay High School. Come along and browse through the software library and get copies of those programs you haven't got in your library. All it costs is $1 per disk, so BE THERE! Depending on the results of this night our Software Librarian, John, will conduct these special nights on a regular basis in future.

FUNNELWEB
FARM
PRESENTS
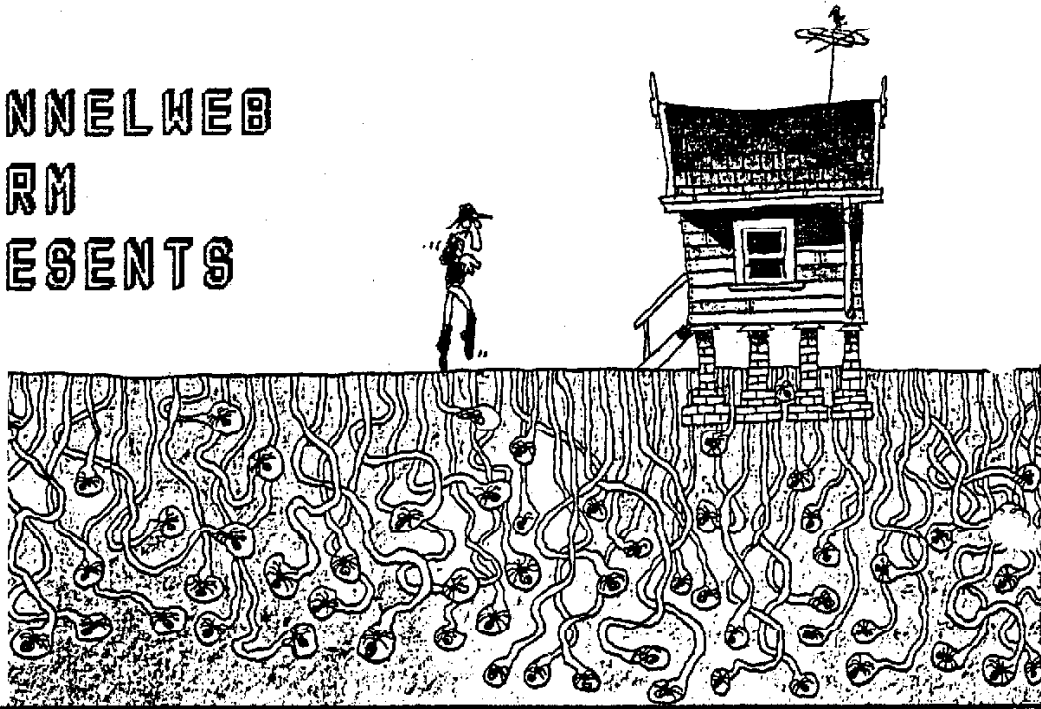
ASSEMBLY
SQUEEZING
part 2
TONY McGOVERN

Let's continue our look at squeezing bytes out of assembly code by looking at *moving a block of words* of data in CPU RAM. Assume that this block is of known length and is to be copied from one fixed address to another. In other words all of these can be represented by symbols or expressions in your assembly source code. The immediately obvious way to code this is

```
      LI   RØ,BLOK1
      LI   R1,BLOK2
      LI   R2,BLLEN
LOOP  MOV  *RØ+,*R1+
      DECT R2
      JGT  LOOP
```

Registers Ø-2 are loaded with the the *relevant values,* and then the word shift is counted out. Using JGT rather than JNE is defensive coding in the same spirit as defensive driving. This takes 9 words (18 bytes) and uses 3 registers. Let's try a first little squeeze.

```
      LI   RØ,BLOK1
      LI   R1,BLOK1
LOOP  MOV  *RØ+,*R1+
      CI   RØ,BLOK1+BLLEN
      JL   LOOP
```

Now it takes one word less and needs only 2 registers. In complex code the register economy can be more important than the 2 bytes saved. Notice that a *logical* comparison is used since it is addresses that are being compared rather than a count being checked. Let's do better still!

```
      LI   R1,BLLEN
LOOP  MOV  @BLOK1-2(R1),@BLOK2-2(R1)
      DECT R1
      JGT  LOOP
```

This now takes only 7 words of code and uses only 1 register. Note that RØ cannot be used here. It is different from the previous code segments in that it starts at the highest word and works downward. If the blocks don't overlap it doesn't matter. If you have been a keen *student of the E/A manual you will* already have known this form of code from reading p1Ø1. If the blocks do overlap then you may be forced to use one form or the other.

A DIFFERENT APPROACH TO SPEECH

by
KEVIN COX

The program opposite does not look much like the normal 'speech program', as it consists mainly of CALL LOADS, but this method works much quicker than the usual CALL SAY method.

When the computer encounters a CALL SAY(" "), it stops execution of the program until it has completed the CALL SAY subprogram, while in the CALL LOAD(" ") method the computer continues on with the program, not waiting for the subprogram to be completed.

The phrases are listed in the Editor/Assembler manual on page 422. The 2 bytes following the phrase are noted and the digits reversed and 64 is added to each digit. After inserting the numbers it must finish with 64, and then 80 is needed at the end to tell the computer to speak that line.

The first program will do all that for you. All you have to do is insert the numbers as they appear in the manual. This program runs in either Extended BASIC or in BASIC with the Mini-Memory module.

```
10 REM *********************
20 REM *SPEECH CONVERSION*
30 REM *    NUMBERS       *
40 REM *  by kevin Cox    *
50 REM *  USING THE E/A   *
60 REM *     MANUAL        *
70 REM *  9th July 1988   *
80 REM *********************
90 CALL CLEAR
100 PRINT "INPUT 4 HEX NUMBERS"
110 INPUT "SEPARATE BY
    COMMAS -";A$,B$,C$,D$
120 IF A$="A" THEN A$="10"
130 IF A$="B" THEN A$="11"
140 IF A$="C" THEN A$="12"
150 IF A$="D" THEN A$="13"
160 IF A$="E" THEN A$="14"
170 IF A$="F" THEN A$="15"
180 I=VAL(A$)
190 I=I+64
200 IF B$="A" THEN B$="10"
210 IF B$="B" THEN B$="11"
220 IF B$="C" THEN B$="12"
230 IF B$="D" THEN B$="13"
240 IF B$="E" THEN B$="14"
250 IF B$="F" THEN B$="15"
260 H=VAL(B$)
270 H=H+64
280 IF C$="A" THEN C$="10"
290 IF C$="B" THEN C$="11"
300 IF C$="C" THEN C$="12"
310 IF C$="D" THEN C$="13"
320 IF C$="E" THEN C$="14"
330 IF C$="F" THEN C$="15"
340 J=VAL(C$)
350 J=J+64
360 IF D$="A" THEN D$="10"
370 IF D$="B" THEN D$="11"
380 IF D$="C" THEN D$="12"
390 IF D$="D" THEN D$="13"
400 IF D$="E" THEN D$="14"
410 IF D$="F" THEN D$="15"
420 K=VAL(D$)
430 K=K+64
440 PRINT K;J;H;I;64;80
450 OPEN #1:"PIO"
460 PRINT #1:K;J;H;I;64;80
470 CLOSE #1
475 PRINT
480 PRINT "ANOTHER SET OF
    NUMBERS (Y/N)"
490 CALL KEY(0,K,S)::
    IF S<1 THEN 490 ELSE
    IF K=89 THEN 100 ELSE END
```

```
90 CALL CLEAR
100 CALL INIT
110 S=-27648
112 PRINT "        READY TO START"
115 CALL LOAD(S,67,"",S,75,"",S,70,"",S,69,"",S,64,"",S,80)
118 FOR I=1 TO 500 :: NEXT I
120 CALL LOAD(S,74,"",S,65,"",S,69,"",S,67,"",S,64,"",S,80)
125 FOR I=1 TO 500 :: NEXT I
130 CALL LOAD(S,68,"",S,71,"",S,73,"",S,70,"",S,64,"",S,80)
140 CALL LOAD(S,64,"",S,76,"",S,71,"",S,68,"",S,64,"",S,80)
150 CALL LOAD(S,66,"",S,67,"",S,74,"",S,67,"",S,64,"",S,80)
160 CALL LOAD(S,70,"",S,73,"",S,70,"",S,70,"",S,64,"",S,80)
165 FOR I=1 TO 500 :: NEXT I
170 CALL LOAD(S,67,"",S,73,"",S,71,"",S,67,"",S,64,"",S,80)
180 CALL LOAD(S,74,"",S,68,"",S,68,"",S,67,"",S,64,"",S,80)
190 CALL LOAD(S,78,"",S,67,"",S,76,"",S,66,"",S,64,"",S,80)
200 CALL LOAD(S,71,"",S,67,"",S,70,"",S,65,"",S,64,"",S,80)
210 CALL LOAD(S,74,"",S,72,"",S,75,"",S,67,"",S,64,"",S,80)
220 CALL LOAD(S,69,"",S,64,"",S,68,"",S,68,"",S,64,"",S,80)
225 FOR I=1 TO 500 :: NEXT I
230 CALL LOAD(S,67,"",S,73,"",S,71,"",S,67,"",S,64,"",S,80)
240 CALL LOAD(S,73,"",S,77,"",S,76,"",S,65,"",S,64,"",S,80)
250 CALL LOAD(S,67,"",S,64,"",S,68,"",S,71,"",S,64,"",S,80)
260 CALL LOAD(S,64,"",S,65,"",S,77,"",S,65,"",S,64,"",S,80)
270 CALL LOAD(S,76,"",S,77,"",S,76,"",S,68,"",S,64,"",S,80)
280 CALL LOAD(S,77,"",S,66,"",S,68,"",S,66,"",S,64,"",S,80)
283 FOR I=1 TO 500 :: NEXT I
285 CALL LOAD(S,78,"",S,75,"",S,65,"",S,71,"",S,64,"",S,80)
290 CALL LOAD(S,73,"",S,77,"",S,76,"",S,65,"",S,64,"",S,80)
300 CALL LOAD(S,69,"",S,78,"",S,79,"",S,68,"",S,64,"",S,80)
305 CALL LOAD(S,72,"",S,66,"",S,78,"",S,66,"",S,64,"",S,80)
307 CALL LOAD(S,75,"",S,70,"",S,74,"",S,71,"",S,64,"",S,80)
310 CALL LOAD(S,77,"",S,71,"",S,67,"",S,68,"",S,64,"",S,80)
320 CALL LOAD(S,65,"",S,68,"",S,76,"",S,68,"",S,64,"",S,80)
330 CALL LOAD(S,68,"",S,71,"",S,73,"",S,70,"",S,64,"",S,80)
340 CALL LOAD(S,73,"",S,78,"",S,75,"",S,67,"",S,64,"",S,80)
350 CALL LOAD(S,76,"",S,77,"",S,76,"",S,68,"",S,64,"",S,80)
360 CALL LOAD(S,77,"",S,78,"",S,74,"",S,67,"",S,64,"",S,80)
365 FOR I=1 TO 500 :: NEXT I
370 CALL LOAD(S,73,"",S,78,"",S,71,"",S,71,"",S,64,"",S,80)
375 FOR I=1 TO 500 :: NEXT I
380 CALL LOAD(S,67,"",S,73,"",S,71,"",S,67,"",S,64,"",S,80)
390 CALL LOAD(S,71,"",S,70,"",S,66,"",S,68,"",S,64,"",S,80)
400 CALL LOAD(S,68,"",S,78,"",S,70,"",S,65,"",S,64,"",S,80)
410 CALL LOAD(S,76,"",S,67,"",S,66,"",S,66,"",S,64,"",S,80)
420 CALL LOAD(S,79,"",S,78,"",S,72,"",S,66,"",S,64,"",S,80)
425 FOR I=1 TO 500 :: NEXT I
430 CALL LOAD(S,72,"",S,68,"",S,65,"",S,67,"",S,64,"",S,80)
440 PRINT "         GOODBYE"
```

GOOD OLD DAYS ARE COMING BACK!

Remember those good old days of 1982-84 when there were loads of books and magazines for us TIers? When we could buy text materials for our computer faster than we could read and use them?

Methinks these days may be returning. At least a little.

At recent fairs and meeting I've attended there has been lots of talk about the disappearing textware in the TI World Community. No more books. No more magazines. No more...

Actually, there is more good, intelligent, useful TI stuff being written for textware today than there was four or five years ago. It's just that most of it is being written by users in user groups, not by some writer in an ivory tower somewhere. Most of these new writings are free (in newsletters with membership or in exchanges with other groups) and readily available.

Someday, I'd love to publish a list of my favorite dozen newsletters worldwide, but then the IMPACT column may be reduced to only 12 publications.

So much for that idea.

Joining a user group, even by mail if there aren't any nearby, may be the most sensible investment a 99er could make.

However, there are still lots of other sources for textware in our fellowship.

COMPUTER SHOPPER ($21 for 12 monthly issues) P.O. Box 1419, Titusville, FL 32781-9988. This 10 X 14 500-plus page giant is probably the best single source for inexpensive disks, drives, cables, monitors, printers, what-have-you. It also has a listing of user groups (Check to see if yours is there.) and a TI classified section. But the reason most of us subscribe is the TI FORUM written by Ron Albright and John Zittrain. These two guys have a folksy approach to the realm of TI computing. Their monthly articles are full of wit and information and lots of heady reviews and predictions.

MICROpendium ($20 for 12 monthly issues) P.O. Box 1343, Round Rock, TX 78680. This is the best magazine source for all TI owners, from super techie all the way down to the ladder to people like myself. The 40-page newsprint periodical is totally devoted to the TI world and is as up-to-date as one can get. If you can only

afford one TI magazine, this is probably the one. (John Zittrain, Ron Albright, Barry Traver, Chris Bobbitt, and Bruce Forbes are probably going to kill me for saying that.)

ASGARD NEWS ($6 for 4 quarterly issues) P.O. Box 10306, Rockville, MD 20850. Though this magazine is fairly new, the editor - Chris Bobbitt - has been programming and writing steadily for the TI community since his teenage years. So it's a natural step up to see a full-fledged magazine come out. Most of the first issue was devoted to previews of coming attractions and reviews of many of Asgard's products. Each issue, though, has an extended article (the 9640 vs the 99/8, for example) and editorials and regular columns and corners. Lots of small, helpful hints and comments are given on a variety of TI subjects.

Genial TRAVelER ($36 for a one-year, six-disk subscription; $65 for a two-year subscription. These subscriptions also have bonuses.) 835 Green Valley Drive, Philadelphia, PA 19128. In a way Barry Traver's diskazines are not strictly a hardcopy magazine, but the DV/80 files dumped will give you the equivalent of a novel. To be a 99er with a disk drive and not have this service is to live a deprived life. These are packed! Really packed disks of an incredible variety of things for the TI: music, utilities, games, wordprocessing, everything else. Far less than a penny a sector, these disks may be ordered as Volume One (the set already complete), Volume Two (still some to come), or both. I would recommend Volume One first.

All of the above you may already own or at least know about. But there is another text/disk setup out there that is really worth discovering. It's been around since September of '87, and it is another large value for we 99ers, though it just hasn't gotten the attention many other TI efforts have gotten.

TId Bits is a bimonthly magazine for only $8 (6 issues) for hardcopy or $12 (6 issues) for disk (SSSD flippy). The disk contains lots of additional PD and Fairware stuff to extend to two sides. The magazine contains photos and graphics.

Mail to Bruce Forbes, editor; Route 2, Box 412; Sumerduck, VA 22742.

It may be hard to make a decision about which version to get - disk or text. Although the disk contains the exact text of the hardcopy (and a lot of other surprise goodies besides), I personally am partial to the hardcopy for a number of reasons.

TId Bits pilot issue contained some screen photos, graphics of an envelope cataloguer, and a screen graphic to go along with a program printed out.

The next issues, however, contained some of the best photographs I have ever seen of the internal structure of our computer. The text talks about chip replacements and

gives some very precise, easy-to-follow photographs and enlargements that make the process very logical, very clear, and - even to a non-techie like myself - very unfrighteningly practical.

It's one thing to see drawings (which are also in the magazine), but it's quite another to have step-by-step photographs of the entire process.

These magazines are jam-packed with all kinds of stuff that you don't find in MICROpendium, ASGARD NEWS, or - because of the nature of the zine - Genial TRAVeLER.

Bruce supplies the reader with all kinds of inside information on a large variety of things for us 99ers. Lots of prices of things. Lots of addresses. Lots of reviews. Lots of programs.

It's a one-man operation at this point, and each issue gets better and better.

Bruce has promised to continue this magazine through at least one year's subscription. And hardcopies of all past issues will be available through 1988 with each subscription. The disk versions will continue to be available, I assume, for a long time to come. Newsletter editors may be interested in getting this version for an excellent source of some interesting material.

Bruce says of his excellent efforts, "A lot of thought went into the pricing of TId Bits. We were keeping in mind that the TI community has been shafted too many times. Therefore, we set a break-even factor projected to December 1988. If we should pick up a sponsor or advertiser throughout the year, those funds would be used to lure a TI Brain to produce unique articles for us." [I'm available, Bruce.]

"We are committed to 1988 and are cautiously optimistically looking to 1989. If the world doesn't end or the TI community completely collapse we project some kind of survival into 1989 and beyond."

Hooray!

Writers and programmers and techies and non-techies, take note! Here is a good source for you to get those articles and designs and comic strips published.

Readers, here is another great TI source that needs

all our support. For only $8 we could be supporting our own futures with our fabulous machines.

Good luck, Bruce!

HELPFUL HINT OF THE MONTH'

This has to do with TI WRITER (or any version: FUNNELWEB, BA WRITER, whatever).

Did you know you can type anything you want after a carriage return ON THE SAME LINE and it won't print out. But it'll SAVE. This is great for text notes for screen reading. I've used this a lot, particularly when I'm writing articles or poems. Very handy.

Also handy is the space use in LF and SF.

When you LF and there is already a DSK2.HEMENWAY up there on auto and you want to load DSK2.KONG, all you have to do is type the KONG and press the Spacebar and Enter, even though it'll say DSK2.KONG WAY on the screen. Actually, you can even SAVE a file with a long (up to 80-col) note after a filename and space. This will disk SAVE and screen read but not print out. Lots of handy reasons for this on working disks.

Another handy-dandy is letter saving. If you write as many letters as I do (between 30 and 40 a week), it's a good idea to let your FUNNELWEB do the sorting and hard work for you. Just name the letter by date and name in the following way. If I write to Jim Peterson on May 17 I would SAVE the file as DSK2.517/JIMPETE. Always month first (5) followed by day (17) and the name as far as it'll go. So if I wrote to Jim on November 27 it would be 1127JIMPET, but if I wrote on April Fool's Day it would be 41JIMPETER. This is also handy if you write a lot of letters on the same day, as TI WRITER will sort them for cataloguing first by number then by letter. So they will be numerically AND alphabetically sorted in the catalog.

Have any handy-dandys of your own (or ones you've come across)? Send them along. I'll be happy to IMPACT them for you, with full credit, of course.

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

EVOLUTION OF THE 4A

by

BOB CARMANY

"Now that Texas Instruments has stopped producing the 99/4A, it is doubtful if the computer will survive more than four months."

This "epitaph" was issued in 1984 not once but several times! What is it about this computer that commands such a dedicated following? Why is it still going strong long after the VIC-20 and IBM PC Jr have come and gone? Let's look at "what was and what is" to see if we can find the answer to why our "orphan" is still very much alive.

Back in early 1984 -- after TI "pulled the plug", there really wasn't much available in the way of hardware and software support for the 99/4A. You could expand to 128K with one of the Foundation cards and you could go DSSD but that was just about the limit of expansion. There were even some glimmers of what was to be as Will and Tony started on Funnelwriter.

TI did some very stupid things over the years that they were marketing the 99/4A, to be sure. They allowed themselves to be drawn into the "bloodbath" of price cuts that eventually led to loosing 1/4 BILLION dollars in one quarter year. That's right, 225 MILLION dollars. Despite that enormous loss, the company was held up by their calculator division which was making equally enormous profits for the company. They also adopted the misbegotten practice of "giving away the razor to sell the blades". By discounting the computer console heavily, they hoped to recoup the loss by selling the software at "more than marginal" profit. These marketing decisions led to the untimely cessation of production of the 99/4A.

On the other hand, they did some things extremely well. Not the least of which was to engineer a durable and reliable product. The TI-99/4A and its peripherals remain the most durable and trouble-free computer around. Texas Instruments used the technology developed as a defense contractor to design the PE-Box to specifications that make it almost "bomb proof". Most repairs are simple and can now be done by the user at home.

Another thing that TI did is to leave a "hook" for future expansion of the system. Although the limits were once 128k and DSSD, there was no reason why they had to remain at that level. In fact, it wasn't long before CorComp, Myarc and others began pushing beyond those limits. One of the first developments was the ability to go DSDD with both the CorComp and Myarc PEB cards and standalone peripherals. It wasn't long before news leaked of a DSQD modification done by some ex-TI employees in Lubbock, Texas. Disk storage went from a maximum of 3 SSSD drives to 4 DSDD drives (or more). There is even a hard drive controller available at the present.

The next step in the evolutionary cycle was the upgrade of storage capacity to include RAMDISKS. The start was a humble 192K Horizon and things completely got out of hand from there! It wasn't long before 256K, 512K, and full 1 Meg models made there appearance. In fact, there is even a 1 Meg standalone model available!

Software development was equally a spectacular. Funnelwriter evolved from a loader for TI-Writer to a complete software shell for the TI that will load just about anything imaginable. PRBASE made its appearance as did FAST-TERM, MASSTRANSFER, and TELCO. The TI went from "I wish I had" to "I've got just about what I need". ARCHIVER 2.4 made the TI on par with the "top of the line" IBM.

Today, some four years after the epitaph for the 99/4A was written, a "fully-expanded" system takes on a completely new meaning. A console with a PE-Box with 3 Horizon 1 Meg RAMDISKs, an AVPC 80-column card, a Myarc hard drive/

floppy controller, a 30 Meg hard drive and a couple of DSDD floppies do an admirable job. Of course, a print spooler and clock card are installed as well (the speech synthesizer has also been installed).

Let's see, what was it they said, 48K and 3 SSSD drives were the maximum configuration . . .

Most of these improvements and enhancements don't come from large companies concerned with the IBM world. They are being produced by small companies, fairware authors, and users group hardware hackers. The greatest growth by far has come SINCE TI quit making the 99/4A. These are the true "halcyon days" of the TI. There is so much available now that was never dreamt of four years ago. Support the TI and, for goodness sake, support your Users Group!!

Notes on the use of

THE
QED
UTILITIES
LOADER

Version 3.1
RON KLEINSCHAFER

The update of the QED utilities loader from Version 3.0 is mainly cosmetic, with some alterations to the screen display, a couple of minor bugs removed and the addition of file or drive error handling. The latter enables quicker loading if a mistake is made in drive choice or the wrong data files disk is put in the drive. As well there has been the addition of ARCHIVER 2.4 to Option 3.

To load the QED files to the module, insert the QED disk in Drive 1, select Option 5 from EDITOR/ASSEMBLER then press ENTER - the QED loader file is named UTIL1 and will auto run. When loaded select the option required - to FULLY load the module takes approx 25 seconds.

OPTION 1:- FUNNELWEB, DM1000, SBUG/6
Note:- This Option is suitable only for F'WEB Vn 4.10 and version 3.1 of the utilities loader.

F'WEB Vn 4.10 as issued loads and gives instant access to QD (Directory). Normally to make use of the User List the UL file is loaded from disk. To make use of the options of :- 9 <CRT ROM> 0 -: of the User List for quick access to Cart Rom files the User List program must be installed permanently and then QD is then loaded from disk for use. This does not cause much delay in having a Directory Option because DM1000 is instantly available, the only drawback being that files cannot be "marked" for use, but it only takes a few seconds to load QD if required. To install User List print out and follow the directions in the file PRINT-ME.

F'WEB 4.1's User List Option :- 9 <CRT ROM> 0 -: has a very useful feature for users of the 32K QED Module. That particular Option looks for and executes each of the first two (if available) program files from the address space >6000 thru >7FFF by searching for the program link addresses then executing the program located there. The QED utility loader Vn 3.1 has the files arranged so that selection of 9 from User List gives instant access to DM1000 while the selection of 0 similarly gives instant access to SUPERBUG version S/BUG6.

DM1000 has been changed considerably to suit the environment of the QED 32K Module so that the module is controlled by the software as options are selected. Because the program is now so fractured and the module is full with the three program options, the saving of printer codes back to disk has been deleted but the codes can still be "set" and used, in fact once a disk has been catalogued this way the printer will stay initialised with those codes until other codes are sent or the printer is turned off.

S/BUG6 has been slightly modified so that returns to F'WEB or PAGE 0 are available from that programs header after quitting by pressing "Q" (or SHIFT Q) then Enter.

OPTION 2:- Unfilled as yet, selection will only load the first

option. It is anticipated that a future update will include extra (perhaps) useful utilities.

OPTION 3:- VIATEL, MASS TRANSFER, ARCHIVER 2.4
The program file VIATEL has been modified so that screen dumps are available ONLY IF YOU HAVE A TI RS232 CARD AND USE THE PARALLEL PORT. The program makes use of direct access to the PIO port and is not compatable with other controllers. To dump a screen to the printer press FUNCTION =. To exit VIATEL turn off the printer press FUNCTION = then press FUNCTION 4, this will return you to Page Ø (title screen).

MASS TRANSFER is used substantially unchanged and quitting will return you to the title screen but at Page Ø ready for your next selection.

ARCHIVER has been installed as the third option. Its use is becoming increasingly used for the archiving of multiple files into one file for transfer by modem, and personal testing of that method indicates that it makes for much faster transfer than using just multiple file transfer. Option 6 on Archivers main menu screen has been changed to :- 6 M/TRANSFER -: selection of that key will instantly load MASS TRANSFER.

It has been anticipated the once the modem files have been used then reloading of the module may be required so the QED Loader program has been installed as another option on the header. It will load instantly and your selection can be made.

OPTION 4:- TEST MODULE.
This option is to test the module. If you are experiencing difficulty in loading the module this test will check each page and if there are any faults it will display roughly where the problem is. There are two error reports and these are FAULT IN REGISTER indicating that Page switching is not functioning and FAULT IN PAGE xx, xx being the Page in the module that has a fault in the main 32K chip. This test totally destroys any data in the module.

Although the loader is name UTIL1 it

can be altered to suit the individual. The utility files could also be put on a seperate disk and used in another drive. The Loader (UTIL1) can be edited with a sector editor so that the loader will look for the files on the drive chosen. Using ASCII with the editor look near the end of the program for DSK1.PAGE and the drive number can be changed. Do the same in the file PAGE32. The utility files themselves CANNOT be renamed as the loaders PAB uses that name, and it is also set with the page option numbers by the key number when chosen.

On the Distribution Disk is a file named UTILFLIP. This is a special version of the loader for owners of Single Sided or 35 track drives. When Option 3 is selected there is a reminder to flip the disk then press enter to load those files, (Pages30 thru 33). It can be renamed UTIL1 for auto loading from E/A Option 5.

On each header there is an option to EXIT MODULE. It is strongly recommended that this option be used before turning off the console. If you accidentally chose this option merely switch off the console then switch back on again. With selection of this option the module is "parked" at a convenient page so that when the console is powered down the first byte of the main header in Page Ø is not corrupted. After power down the module can be removed and used at any time as required. Insert the module in the console with the power off then when switched on the module is reset at Page Ø ready for use. If at any time the console "locks up" because of a program or any other fault the header may seem to be "lost", before reloading the module again from disk try this --> Select Basic, Type CALL LOAD(24576,170,Ø), then quit and the header should be available. If not then reloading from disk is required.

All program files used as options are FAIRWARE and you are reminded to honor that principal.

I would appreciate any suggestions for improvements or any bugs encountered, just write !!

Ron Kleinschafer - HV99ers
30/6/88.

```
SCR #10
  0  RHT28Jly88
  1  --------------------------------------------------------
  2  -
  3  -
  4  -    STRUGGLING FORTH
  5  -
  6  -    HU99ERS  AUGUST ARTICLE 1988
  7  -
  8  -    :Analysing the Ti-Editor
  9  -    :An any size editor
 10  -
 11  -
 12  -
 13  -
 14  -
 15  --------------------------------------------------------
```
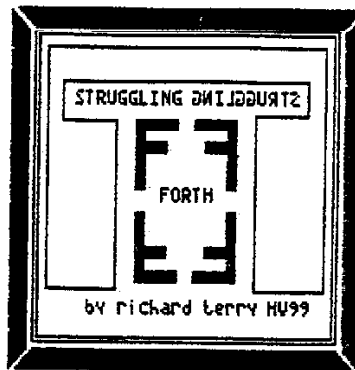
Spurred on by a visit by Joe who confessed he still thought that Forth was the "only way to go", and that come the end of the year he would be getting into it (forth), I decided to write this months article!

One of the most valuable exercises I've ever undertaken in my programming was the bit by bit dissection of the Ti-Forth Editor. I initially undertook this task just to be able to understand its workings, but it soon dawned on me what a powerful piece of code this was. Wouldn't it be great if I could make the code general, ie be able to have an editor of any size, from postage stamp to full screen size.

It was early days in my Forth career, and I've got to admit it took several attempts over some months to get the hang of it. Once I understood it, I then replaced the fixed numerical values with variables or constants, to make it totally general. Whereas I have used constants for the changeable sizes here, when using multi-editors, variables would be more appropriate, so you can change them accordingly

THE ANY-SIZE-EDITOR.

After typing in the code, and compiling it, its dead easy to use. For example a full screen editor is obtained by:

0 0 40 24 EDITOR

ie a 40 column by 24 lines starting at row 0, col 0

10,10,20,5 EDITOR

gives an editing area, framed, 20 wide by 5 high starting at row 10 column 10

HOW IT WORKS.

You will note the supplied code is fully commented, perusal of this will show how the editor works.

A couple of additional comments:

The SBOX box drawing routine has no validation, if you exceed the logical screen limits the computer will lock up. Similarly the word EDITOR-FRAME will also lock up. This word checks to see if the start row or column are zero and if so will not draw a surround frame. However it also does not allow for idiot input!

The RKEY routine, comes from Superforth. Permission for its use was originally given to me by letter from Edgar Dohmann, and it has been printed in the magazine before. I never figured out how it worked, nor why one needs the BOX routine. If any bright spark would like to decipher its workings and present it in the magazine, feel free.

VED.

This word controls the whole of the editor workings. It simply awaits a key press and first checks to see if the user has inputted a function or control key which has been allocated a special function, such as insert, delete etc. If no such special key

has been pressed it then checks to see if the input is a printable ASCII code, between 16 an 127. NOTE: if you do not program in Superforth you must change the lower value to the hex equivalent of 31, as these lower values are used for graphics characters.

```
DUP 1F > OVER 7F < AND
```

If it is printable it prints it to the screen, and stores the duplicate to the buffer area where the screen display data is kept. If it is not printable, and is not a control/function key, it honks the user with a 7 emit.

The construct of VED illustrates many interesting points. Failure to understand the complete workings of the BEGIN ----- UNTIL, and the CASE ---ENDCASE construct can cause you no end of frustration when you try to develop an application based on the above code, so I will go into it in some detail.

BEGIN....UNTIL REVISITED.

To refresh beginners memories, the construction is:

```
BEGIN ..  guts of code ...UNTIL
```

Once the program flow gets you past BEGIN, the guts of the code within the loop is executed. UNTIL then tests the TOP OF THE STACK. If non-zero UNTIL consumes the top value and the loop is exited. If the top of the stack is ZERO it is a signal to execute once again the guts of the code. If you fluff up and don't leave a non-zero value at the appropriate time, you will never get out of the loop!!!!!

Examine SCR# 29 & 30. Note the only time we want to jump out of the editor in this case is when Function 9 (keycode of F) is pressed. This line is followed by a 1 which is left on the stack, all the other control/function key lines have a zero(0) after them.

Seems simple eh!!!!

CASE....OF...ENDOF....ENDCASE.

As reviewed before in previous articles, this is an invaluable

routine. It allows you to test the top stack value against a series of options, and channel the program flow down particular directions before reaching ENDOF and ENDCASE.

However, problems with your stack orders can arise as a result of what happens between the last ENDOF and ENDCASE. When the value is tested against your options such as in:

```
CASE
1 OF .....  ENDOF
```

The actual value being tested is not destroyed, as it would be if tested against IF etc. After the last ENDOF the value is left on top of the stack to the consumed by ENDCASE, whether or not it correlated with one of the OF options.

Now, if one has tested all the case options as in our example, and wants to do something in between the last ENDOF and the ENDCASE, such as we do here to test for printable characters etc, whatever you do must leave a value on top of the stack to be consumed by ENDCASE. Failure to do so will lead to erratic results which if what you are doing here is complicated, can be hard to track down, so keeping absolute track of your stack is essential.

Additionally in our definition, a value must also be left on the stack for UNTIL to consume - here a zero(0). (see the IF...THEN..ELSE statement on SCR# 30)

IMPORTANT NOTE.

If you type in this code to use, note that it will dump the editor data onto the screen number in the variable SCR, here set to 8 by the definition MEDIT. You can change this to suit yourself, but be aware if you keep it as 8 it will overwrite the contents of screen 8! Note also the 0(zero) in MEDIT, it actually sets the VDP position the cursor will appear at when you first enter the editor.

Another point of interest is in the definition EDITOR. Try changing the value of SCRN_END back to its default value of 960, (which you

should anyway after using this definition). Then reload screen 32 and try the full screen editor again. When one reaches the last screen position the display will be vertically scrolled off the screen. Adjusting the screen-end position to one greater than necessary prevents this occurring.

APPLICATIONS OF ANY-SIZE-EDITOR.

These are limited only by your imagination. I have used it in a program I use to design all the screen displays for my programming, in others to be able to edit just part of the screen, yet again as a character generator in a program to design and build tailor made pattern descriptor tables, and in another it is used in a data-base type program. The uses are endless. You can even have more than one full screen editor operating on different size windows at the same or different times within a single program, or let the user adjust the editor window to suit himself. If I can find time I'll put the code the the character pattern generator in a subsequent issue.

ADDRESS FOR CORRESPONDENCE

RICHARD TERRY
141 DUDLEY RD
WHITEBRIDGE 2290
NEWCASTLE NSW
AUSTRALIA

049 436861/292450

SCREENS FOR RICHARDS APPLICATION BEGIN OVER THE PAGE

FOR SALE

NOTICEBOARD

Have you got something to sell? After some hardware, software or modules? Don't forget, members of our Group can advertise free in this Newsletter

As ever, we are always on the lookout for articles for the newsletter. If you've got something to say, or discovered something new about our computer, share it with the world. Contributions covering anything regarding the TI are always welcome, especially articles on BASIC & Extended BASIC. Written a program or even a simple routine? Pass it on to the rest of us.

If you have an idea for an article but don't feel you could write it, let the Editor know so he can make arrangements for someone to do it for you.

The group offers various classes for members & their friends. Why not join one? You can currently attend tutorials on Extended BASIC & an introduction to Genealogy, both classes being held at Warners Bay.

If you haven't come along to one of our monthly meetings recently, why not make the effort and join us at our new meeting place, the Argenton Community Centre - see the map on the inside back cover. The new venue will allow us to offer more variety at the meetings, due to the increased size of the hall.

Program:ANY-SIZE-EDITOR   Date:28July88

Screen #11

```
\ Any size Editor          28Jly88

BASE DECIMAL
    0 VARIABLE ER           \ ER->RLOG variables for autorepeat
    0 VARIABLE EC
    0 VARIABLE BLINK
    0 VARIABLE OKEY
    0 VARIABLE CC
    0 VARIABLE KC
   10 CONSTANT RL
   50 CONSTANT RH
   RH VARIABLE RLOG
    0 CONSTANT START-COL     \ start column of editor
    0 CONSTANT START-ROW     \ start row of editor
    0 CONSTANT C/LN          \ no of characters per line
    0 CONSTANT CHAR/SCRN     \ no characters per editing screen
    0 CONSTANT L/SCRN        \ no of lines per editing screen
```

Screen #14

```
\ Any size editor          28Jly88

\ location to hold editing cursor position
: CUR R# ;
\ store cursor
: !CUR                        \ expects a 0 to max pos
            0 MAX             \ keep larger position
            CHAR/SCRN MIN     \ clip to max pos if greater
            CUR !             \ otherwise store as curpos
            ;                 \ leaves nothing on stack
\ increment cursor position by n positions
: +CUR                        \ expects n to increment by
            CUR @ +           \ add to current curpos
            !CUR              \ re-store in variable
            ;                 \ leaves nothing on stack
```

Screen #12

```
\ Any size Editor          28Jly88

\ same as PICK ie place copy of the nth number on top of stack
: P    2 * SP@ + @ ;                       \ expects n

\ routine to draw a box/rectangle -do not exceed screen size
: SBOX         \ expects Leftcol/up row/Rghtcol/lower row
     4 P  4 P             1  17 HCHAR   \ upper left cnr
     4 P  OVER            1  19 HCHAR   \ lower left  cnr
     OVER 4 P            1  18 HCHAR   \ upper right cnr
     OVER OVER            1  20 HCHAR   \ lower right cnr
     4 P 1+ OVER 4 P 1- 7 P -  21 HCHAR  \ lower horizontal
     4 P  4 P 1+ 3 P 1- 6 P -  22 VCHAR  \ left  vertical
     OVER 4 P 1+ 3 P 1- 6 P -  22 VCHAR  \ right vertical
     4 P 1+ 4 P  4 P 1- 7 P -  21 HCHAR  \ upper horizontal
                          3DROP DROP ; \ leaves nothing
```

Screen #15

```
\ Any size editor          28Jly88

\ increment cursor by n lines
: +LIN                        \ expects n of lines to increm
            CUR @ C/LN        \ cursor pos,charact per line
            /                 \ = line number currently on
            +                 \ add n want to increment by
            C/LN *            \ this is start of new line
            !CUR              \ store to cursor variable
            ;                 \ leaves nothing on stack
\ print shape of cursor to the screen
: CURSOR                      \ expects ascii cursor shape
            EC @ ER @ 40 * +  \ give vdp cursor position
            VSBW              \ print image of cursor
            ;                 \ leaves nothing on stack
```

Screen #13

```
\ Any size editor          28Jly88

\ draws frame around area to be edited expect if full screen
: EDITOR-FRAME                 \ expects nothing
            START-COL          \ starting column
            START-ROW MIN      \ starting row, leave smallest
     IF     START-COL 1-       \ frame starting column
            START-ROW 1-       \ row starting column
            OVER               \ col,row,col
            C/LN    + 1+       \ col,row,col+c/ln +1=width
            OVER               \ col,row,width,row
            L/SCRN + 1+        \ col,row,width,row+l/sc+1=hgh
            SBOX               \ col,row,width,height->box
     THEN                      \ unless row/col out of bounds
            ;                  \ leaves nothing on stack
\S note there is minimal validation on this routine, if you put
most out of bound values program will lock up
```

Screen #16

```
\ Any size editor          28Jly88

\ write a single line of editing screen to viewing monitor
: (LINE)                       \ expects index/scr#
            BLOCK  SWAP        \ start of block adr/ind
            C/LN * + ;         \ block adr + offset
                               \ expects from to
: LINE.
            DO
            I SCR @            \ I = line # or SCR
            (LINE)             \ leave buffer addr,
            START-ROW 40 *
            I 40 * +
            START-COL +        \ start vdpadr of this row
            C/LN               \ no of characters to type
            VMBW               \ write these to the monitor
            LOOP ;             \ leaves nothing on stack
```

*** Forth screen print utility -fairware- by Richard Terry Hv99ers ***

Program:ANY-SIZE-EDITOR   Date:28July88

Screen #17

```
\ Any size editor

\ autorepeat for editor
: RKEY BEGIN ?KEY -DUP 4 BLINK +! BLINK @ DUP 60 < IF 30 CURSOR
ELSE CC @ CURSOR ENDIF 120 = IF 0 BLINK ! ENDIF IF KC @ 1 KC +!
0 BLINK ! IF RLOG @ KC @ < IF RL RLOG ! 1 KC ! 1 ELSE OKEY @
OVER = IF DROP 0 ELSE 1 DUP KC ! ENDIF ENDIF ELSE 1 ENDIF ELSE
RH RLOG ! 0 KC ! 0 ENDIF UNTIL DUP OKEY ! DUP 31 > OVER 127 <
AND IF DUP CURSOR ELSE CC @ CURSOR ENDIF ;

HEX : BOX 8F7 8F1 DO 94 I VSBW LOOP ;  DEC
```

Screen #20

```
\ Any size editor                 28Jly88

\ delete all characters to right cusor and transfer to PAD
: DELHALF                         \ expects nothing on stack
            PAD C/LN BLANKS       \ fill PAD with blanks
            PTR                   \ adr in block corresponding
            PAD                   \ adr of PAD
            C/L                   \ characters per line
            R/C DROP              \ leave column
            -                     \ = no of char to delete
            CMOVE                 \ place these in PAD
            ;                     \ leaves nothing on stack
```

Screen #18

```
\ Any size editor                 29Jly88

\ calculate the addr to place character under cursor
: PTR
            CUR @                 \ current cursor position
            SCR @                 \ current screen number
            BLOCK                 \ address of data block
            +                     \ add= adr to put character
            ;                     \ leaves adr of char in block
```

Screen #21

```
\ Any size editor                 28Jly88

\ print editing cursor to the screen
: .CUR                            \ expects nothing
            CUR @ C/LN /MOD       \ curpos,chr/line /mod=col,row
            START-ROW + SWAP      \ adjust starting row
            START-COL + SWAP      \ adjust starting column
            2DUP ER ! EC !        \ store copy for RKEY
            AT                    \ spot to display cursor at
            EC @ ER @ 40 * +      \ =vdp posit under cursor
            VSBR CC ! ;           \ store this byte in VAR CC
\ increment and display cursor by n
: +.CUR                           \ expects n
            +CUR                  \ inc cursor position by n
            .CUR                  \ display it
            ;                     \ leaves nothing on stack
```

Screen #19

```
\ Any size editor                 28Jly88

\ leaves current column row on stack
: R/C                             \ expects nothing
            CUR @ C/LN            \ cursor pos,character/line
            /MOD                  \ leave remainder,quotient
            ;                     \ leaves column, row
\ leaves column, char per line on stack
: R-C
            R/C                   \ leaves column ,row
            DROP                  \ drop row
            C/LN                  \ characters per line
            SWAP ;                \ leaves column,char/line
```

Screen #22

```
\ any size editor                 28Jly88

\ delete a whole line from editing screen
: DELLIN                          \ expects nothing
            R/C SWAP              \ leave row,column
            MINUS                 \ column = - column
            +CUR                  \ gives curpos at start line
            PTR PAD C/LN CMOVE    \ move whole line to PAD
            DUP                   \ duplicate row number
            L/SCRN SWAP           \ stck row, last row, to row
            DO                    \ move rest of screen up 1line
                PTR               \ start of deleted line
                1 +LIN            \ increment curpos to next lne
                PTR               \ addr of start of next line
                SWAP C/LN CMOVE   \ move line up one
            LOOP                  \ do this for whole screen
            --)
```

Program:ANY-SIZE-EDITOR   Date:28July88

Screen #23

```
\ Any size editor          28Jly88

        0 +LIN              \ replace cursor at start line
        PTR C/LN BL FILL    \ blank out the line
        C/LN !
        !CUR               \ store cursor position
        ;
\ insert line held at PAD
: INSLIN                    \ expects nothing
        R/C SWAP            \ leaves row,column
        MINUS              \ makes column -column
        +CUR DUP           \ decr curpos to start line
        L/SCRN             \ lines per editing screen
        SWAP               \ stack,row/l/scr,row
        -                  \ number of rows below current
        --)
```

Screen #24

```
\ Any size editor          28Jly88

        1- DUP             \ current row -1
        +LIN               \ increment cursor to here
        L/SCRN SWAP -
        L/SCRN
        DO
          PTR              \ current char posit in buffer
          -1 +LIN PTR
          SWAP  C/LN CMOVE
          -1
        +LOOP
        PAD PTR C/LN CMOVE
        C/LN !
        !CUR ;
```

Screen #25

```
\ Any size editor          28Jly88

\ write blanks from curpos to end of the line
: BLNKS                    \ expects nil,leaves nil
        PTR                \ adr in block
        R-C -              \ # char to end of row
        BL FILL ;          \ fill to end row with blanks
\ delete a single character from the row
: DEL                      \ expects nil,leaves nil
        PTR DUP 1+ SWAP    \ adr,adr+1
        R-C -              \ no of char to end of line
        CMOVE              \ move these back a space
        BL PTR             \ ascii blank,current posit
        R/C DROP -         \ adr of start of the row
        C/LN + 1 -         \ adr of end row -1
        c! ;               \ store a blank here
```

Screen #26

```
\ Any size editor          28Jly88

\ print row
: RELINE
        R/C SWAP DROP DUP  \ 2 copies of current row
        13 EMIT
        LINE.              \ print this line
        UPDATE             \ mark buffer updated
        .CUR ;             \ print cursor
\ store the ascii character to block buffer
: !BLK                     \ expects ascii of a char
        PTR                \ adr in block to store to
        C!                 \ store byte
        UPDATE             \ mark buffer updated
        1 +.CUR            \ increment cursor
        ;
```

Screen #27

```
\ Any size editor          28Jly88

\ insert a single character
: INS                      \ expects nothing on stack
        BL PTR DUP         \ ascii 32, addr under cur #2
        R-C -              \ = no of char to right curs
        +                  \ = adr of end of this line
        SWAP               \ stk,adr at end,adr under cur
        DO I C@ LOOP DROP  \ place these bytes on stack
        PTR DUP            \ 2 copies of adr under cursor
        R-C - + 1-         \ adr at end of line - 1
        SWAP 1-            \ adr under cursor   -1
        SWAP               \ stack to adr,from adr
        DO I C! -1 +LOOP   \ resave blank and the chars
        ;                  \ leaves nothing on stack
```

Screen #28

```
\ Any size editor          28Jly88

\ redraw the entire editing screen
: REDRAW                   \ expects nil,leaves nil
        L/SCRN 0 LINE.     \ display this screen
        UPDATE             \ mark as updated
        .CUR ;             \ reprint cursor
\ insert a blank line on editing screen
: INS-BLANKLINE            \ expects nil,leaves nil
        INSLIN             \ insert a line
        BLNKS              \ fill with blanks
        REDRAW ;           \ redraw editing screen
```

*** Forth screen print utility -fairware- by Richard Terry Hv99ers ***

Program:ANY-SIZE-EDITOR   Date:28July88

Screen #29   ## Screen #32

```
\ Any size screen editor      28Jly88      \ Any size editor          28Jly88

HEX                      \ base hexadecimal   \ Used as eg 0 0 40 24 EDITOR= full screen editor
\ Control of entire editor actions            : EDITOR                 \ exp col/row,c/ln/#lines
: VED                    \ expect a = spot to start cur       2DUP       \ duplicate top 2 items
    BOX !CUR .CUR        \ display cursor at vdp a spot        # 1-       \ col,row,char per screen-1
  BEGIN RKEY             \ detect key presses                 DUP 959 =  \ if want full screen editor
        CASE             \ was the key pressed:                IF 961 SCRN_END ! \ this stops scrolling
\ note the zeros before endof are left for UNTIL               THEN
    3 OF DEL RELINE   0 ENDOF \ F1=delete single character     ' CHAR/SCRN  !  \ save chars per screen
    4 OF INS RELINE   0 ENDOF \ F2=insert a blank space        ' L/SCRN     !  \ lines per screen
    6 OF INS-BLANKLINE 0 ENDOF \ F8=ins blank line above curs  ' C/LN       !  \ characters per line
    7 OF DELLIN REDRAW 0 ENDOF \ F3=delete line under cursor   ' START-ROW  !  \ starting row
    8 OF -1 +.CUR     0 ENDOF \ F'n S=left arrow key           ' START-COL  !  \ starting column
    9 OF 1 +.CUR      0 ENDOF \ F'n D=right arrow key          MEDIT         \ and jump into editor!
    A OF C/LN +.CUR   0 ENDOF \ F'n X=down arrow key           ;             \ leaves nothing on stack
    -->                       \ keep compiling

                                              DECIMAL
```

Screen #30   ## Screen #33

```
\ Any size screen editor      28Jly88       ( Any size Editor )       28Jly88

    B OF C/LN MINUS +.CUR 0 ENDOF \ F'n E=up arrow key
    B OF 1 +LIN .CUR    0 ENDOF \ Enter=to start of new line  : \   IN @ 64 / 1+ 64 # IN ! ; IMMEDIATE ( ignore rest line )
    F OF 0 0 AT QUIT    1 ENDOF \ F9=quit editor mode         : THRU 1+ SWAP DO I LOAD I . LOOP ;
    1E OF INSLIN REDRAW 0 ENDOF \ Cntl 8= insert line
\ if no function key press is detected check if valid character  11 22 THRU
                                                               25 29 THRU
        DUP 10 > OVER 7F < AND  \ is character printable       31 32 THRU
        IF DUP EMIT DUP !BLK    \ yes, emit,save to buffer
           0 SWAP               \ stack 0,ascii for endcase
        ELSE 7 EMIT 0 SWAP      \ not printable honk user
        THEN
          ENDCASE               \ this consumes top stack item
      UNTIL                     \ quit only if 1 on top
        ;                       \ leaves nothing on the stack
DECIMAL                         \ fix up base !!!
```

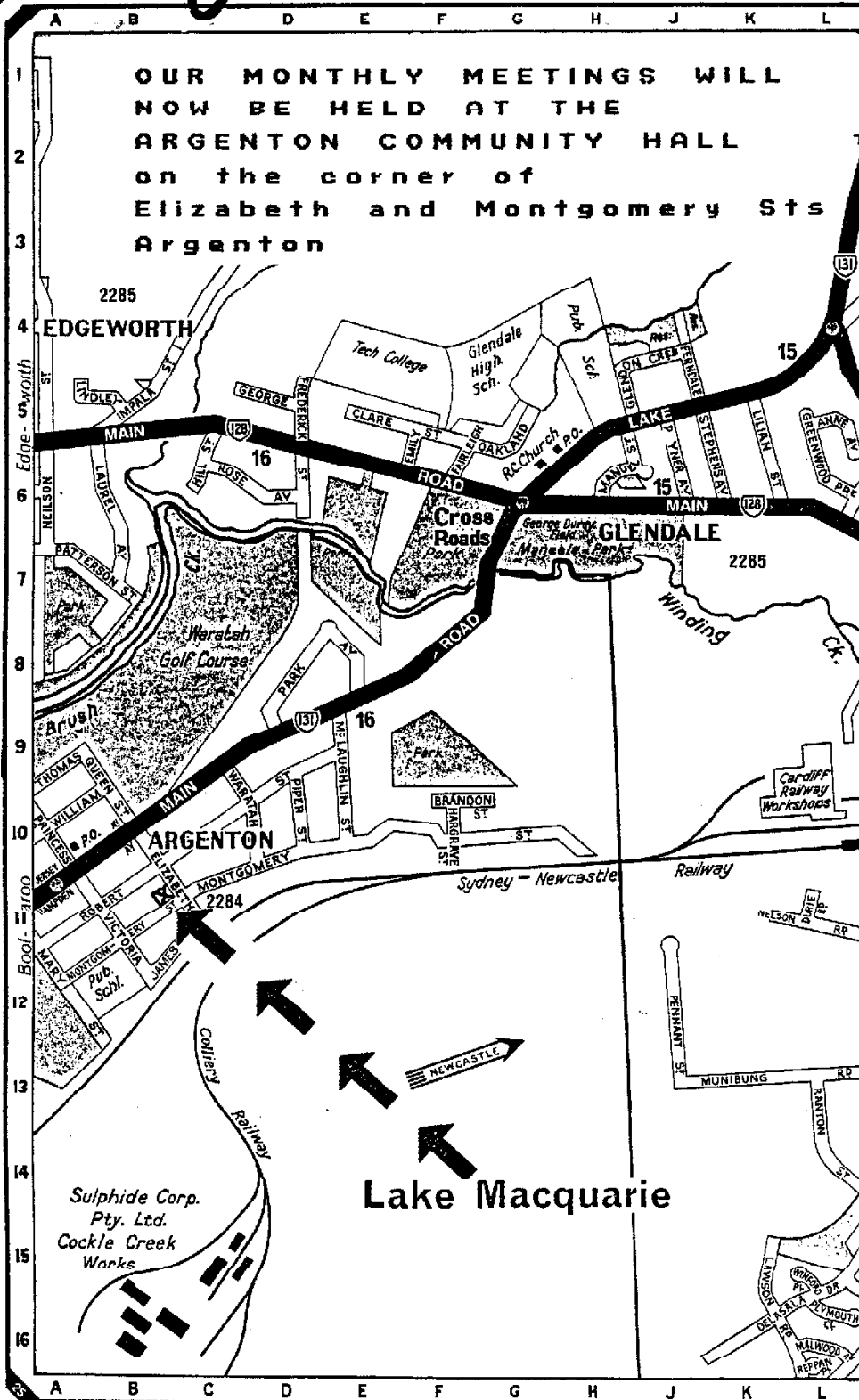## Screen #31

```
\ Any size editor      28Jly88



\ draws frame,enter editor, display screen contents
: MEDIT
        8 SCR !          \ put editor screen on 8
        EDITOR-FRAME     \ draw the frame
        L/SCRN 0 LINE.   \ fill editor with content of8
        0                \ used by !CUR to zero to top
        VED              \ enter editor
        ;                \ leaves nothing on stack
```

*** Forth screen print utility -fairware- by Richard Terry HV99ers ***

Take note of this

OUR MONTHLY MEETINGS WILL
NOW BE HELD AT THE
ARGENTON COMMUNITY HALL
on the corner of
Elizabeth and Montgomery Sts
Argenton

Lake Macquarie

ARRAYS

In this column I often do strange things that you normally would not do in your program. I do this to show you what certain commands will do. For instance in line 210 of the program below, I used a GOSUB 390 to clear the screen instead of CALL CLEAR. This is just to show you that there are many ways to arrive at the same result.

Our tips this month is on ARRAYS and how they work. An array is like a pigeon hole where you can store information for later use. If X=1, what you put in Row(X) will be available to use at any time as Row(1). Try this:

```
100 CALL CLEAR
110 NAME$(9)="MY NAME"
120 NAME$(4)="YOUR NAME"
130 PRINT NAME$(9);" IS ";NAME$(4)
140 PRINT NAME$(4);" IS ";NAME$(9)
```

By now you should know that if you use a string variable you need the dollar sign. If you are going to use a number larger than 10 you need to DIMension your array before you use it. To see what it will do leave out line 110 in the program below. Your DIM(x) can be larger than needed but not smaller. However the DIM(x) statement reserves space in memory so it's best to only use what you need. Enter the program below and run it while looking at the printout and I am sure you won't have any trouble finding out what it's doing. If you don't understand something ASK!

```
100 CALL CLEAR
110 DIM ROW(24),COL(32)
120 PRINT "READING DATA INTO TWO
    ARRAYS":::
130 FOR X=1 TO 24
140 READ ROW(X)
150 READ COL(X)
160 PRINT "ROW(X";STR$(X);")
    =";STR$(ROW(X)),"COL(X";
    STR$(X);")=";STR$(COL(X))
```

```
170 NEXT X
180 PRINT :: "DATA HAS BEEN READ
    AND IS" :: "NOW IN TWO ARRAYS:"
    :: "ROW(X) COL(X)" :::: "PRESS
    ANY KEY TO USE DATA"
190 CALL KEY(0,K,S)
200 IF S=0 THEN 190
210 GOSUB 390
220 FOR A=1 TO 24
230 CALL HCHAR(ROW(A),COL(A),42)
240 NEXT A
250 GOSUB 370
260 FOR B=24 TO 1 STEP -1
270 CALL HCHAR(ROW(B),COL(B),42)
280 NEXT B
290 GOSUB 370
300 FOR R=1 TO 20 STEP 4
310 FOR I=1 TO 24
320 CALL HCHAR(COL(R),COL(I),42)
330 NEXT I
340 NEXT R
350 GOSUB 370
360 STOP
370 FOR DELAY=1 TO 500
380 NEXT DELAY
390 CALL CLEAR
400 RETURN
410 DATA 4,5,2,6,5,7,4,8,6,9,6,10,
    6,11,8,12,9,13,10,14,11,15,12,
    16,13,17,14,18
420 DATA 15,19,16,20,17,21,18,22,
    19,23,20,24,24,25,23,26,22,27,
    21,28
```

HAPPY PROGRAMMING!!!

TEXAS INSTRUMENTS
HOME COMPUTER
READY-PRESS ANY KEY TO BEGIN
© 1981 TEXAS INSTRUMENTS
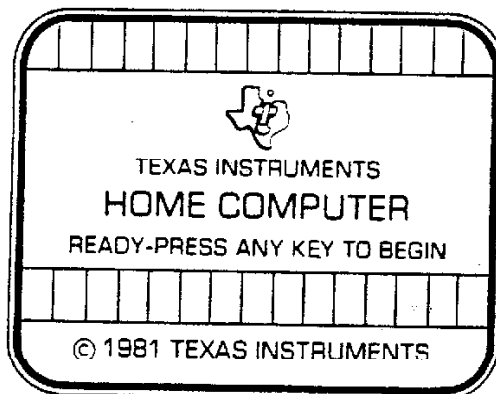
THE INFORMATION PAGE

IN YOUR NEWSLETTER THIS MONTH

In the News - a round-up of TI happenings	A. Wright
Random Bytes	B. Carmany
Assembly Squeezing -Part 2	T. McGovern
A Different Approach to Speech	K. Cox
Impact 99	J. Sughrue
Evolution of the 4A	B. Carmany
QED Utilities Loader V3.1	R. Kleinschafer
Any Size Editor in FORTH	R. Terry

PLUS MUCH MUCH MORE!!!!!

COMING EVENTS

Software Library Access Night:Tuesday 16th August, 1988
(come along to Warners Bay and get copies of progs from library)
Next Committee Meeting: Tuesday 6th September, 1988
General Meeting: Tuesday 13th September, 1988

AGENDA FOR SEPTEMBER MEETING

Demonstration of software for Modems
Super Duper Games Competition

CLASSES AVAILABLE FOR MEMBERS

XB Class Tuesday 16th August, 1988 at Warners Bay
Genealogy -TYFA- Thursday 18th August - contact Joe Wright for location

ANNUAL SUBSCRIPTIONS

Subscriptions to the Group cover the period 1 July to 30 June
following year. Membership enquiries are welcome; please address
all enquiries to the Secretary.

The annual subscription is:
Australian Residents...$25
Overseas Residents.....$45 (airmail)

Back issues of our Newsletter are available for $1 plus postage

C#97   OK

Mr.R. CARMANY
1504 Larson St.
GREENSBORO NC.27407
U.S.A.