TI*MES

No 56 Spring 1997

This is a holding copy containing the few pages I have been able to scan. Although the issue ran to 84 pages, it was full of random images placed over scattered text, and black text on black images is not worth scanning. Even the white pages were largely black.

The magazine was edited entirely by Richard Twyning.

The Chairman reported building a PC to run the "Group BBS" due to failure of the Myarc HFDC.

The editor had also purchased a PC

The BBS PC was purchased with group funds and catered only for TI consoles with 80 column cards.

To demonstrate the readability of many pages I will include a best efforts scan of the original Page 31 at the end of this short document..

However, if you think you could do the job then let me know. YOU GET AN EDITORS PEB, with the job!

Advertising will also start in the New Year at full pelt. This will be pushed at various sources. So watch this space. I will be sorting out a letter to a load of our old members who gave up the group and we will see if they might return.

Well I have done with the chit chat so now it is:

PROGRAMMING TIME

Part 3 of the Line-By-Line Tutor
We now move on into the depths of the LBL (Line-By-Line) assembler. We will now show you how we can give you the BIG programming power.

Instructions in Assembly Language (Both LBL and Editor/Assembler) describe very closely the way the processor operates at its lowest level. This can be an advantage or a disadvantage. This makes the lists in the program VERY long, for doing just a small action on the computer, but it makes any program you write VERY fast. So to bring you into the feel of programming the following programs will not give the Mini-Memory LBL Memory locations. This is so you can get used to entering code from other places, like this Mag, and see how much you have to write to make something simple work.

When we use a computer we do all our action via ADDING and SUBTRACTING or Increasing or Decreasing. So I will show you how this works.

Increasing and decreasing a Value.

This is done with provided subroutines that are stored in memory locations. These are referenced by the four instructions INC INCT DEC and DECT.

INC (INCrement) this increases your value in a memory address or register by one, adds one to the memory there.
i.e. 7D00 INC R3 (adds one to the value in R3)

DEC (DECrement) decreases the value of a register or memory location by one. Opposite of INC
i.e. 7D00 DEC R3 (takes away one from value in R3)

Remember if you reference a Hexadecimal or decimal memory location directly, it must be proceeded by the @ symbol, except when using JUMP instructions.
i.e. 7D00 DEC @>7F00

INCT (INCrement by Two) this adds two to your value in a register or memory location.

DECT (DECrement by Two) the same as DEC but decrements by two.

The AI (Add Immediate) instruction is different in that it adds a fixed value to a register, and can ONLY BE USED WITH REGISTERS. This fixed value must be specified along with your source code, and is always stored with the program. The A (Add) instruction adds dynamic values that are contained in registers, or memory locations.

ADDING and SUBTRACTING

AI (Add Immediate) This lets you add or subtract from a value stored in a register. To cause it to subtract, you simply need to stick a minus sign in front of the value you specify as the operand. If you say wanted to add 32 to the value in R4 you would put the following:

7D08 AI   R4,32 (the value of R4 is increased by 32)

       To add >312 to say R12

7D08 AI   R12,>312 (Value now increased by >312)

To subtract all you do is add the minus sign to reverse the process.

7D08 AI   R4,-32 (Subtracts the value 32 from R4)

7D08 AIR12,->312 (Value now decreased by >312)

The same instruction can be used to subtract the Value from the contents register. (Subtract immediate does not exist!). Now just add the Negative value of the number you wish to subtract.

i.e.: 7D00 AI R7,-712 (Subs 712 from R7)

 To subtract >24 from the value stored in R15

i.e.: 7D04 AI R15,->24 (Subs >24 from R15)

The result of addition or subtraction by the AI instruction is placed in the same register where the initial value was stored. In the previous example, for instance, the value after subtracting >24 from the value in R15 is placed back into R15.

The A and S instruction.

A and S are used in many cases where you want to add and subtract the values of two registers, or memory locations, or a combination of a Register and a memory location. The A (add words) and the S (subtracts words). These instructions are VERY useful indeed. Being WORD instructions these use the full FOUR DIGIT hex numbers, in a register or an address.

The A (add Words) instruction adds the word value in the first operand to the word value in the second operand. It then places the addition in the second operand. Assuming that R3 is loaded with, say >1202 and R1 with >1362 then the following line would be thus:

7D00 A R3,R1

This adds >1201 and >1362 (total >2563) and places the answer in R3. The first operand remains unchanged by the operation. If you want the answer in R1 then just turn the operands round.

IE 7D00 A   R1,R3   (ADDS THE VALUES & PLACES THE ADDITION
                             IN R1 LEAVING R3 UNCHANGED)

Easy? I told you it was!!!!

The S (Subtract words) works the same way only subtracting instead of adding. The value again of the First operand is subtracted from the second operand and places the answer in the second. If say R5 is loaded with 2 and R7 with 5 then it looks like this:

7D00 S   R5,R7   This would subtract R5 (2) from R7 (5) and places the
                          answer (3) in R7. The value of R5 remains the same.

Some other examples would be :

7D00 S   @>7FC2,@>7100   (subtracts the word at location >7FC2 from >7100 leaving
                             result in >7100)

7D00 S   @>7F00,R14   (subtracts the word value in >7F00 from the word value
                             found at R14 and leaves the result in R14)

Now we go to the AB and SB instructions which area similar to the A and S but the AB and SB add and subtract BYTES. They do the same as the above word examples but they only operate on the left (most significant byte) of the word. So if say the Register R4 was

loaded with >0492 and R5 was loaded with >1067, the result of a AB on this would look like this:

7D00 AB R4,R5   The left Byte of R4 = >04 (>0492)

The left byte of R5 = >10 (>1067)

So the answer (>14) would be placed in the left byte of R5, so R5 would be >1467 and R4 remains unchanged.

SB is exactly the same, but subtracts the value found in the left BYTE. If R2 was loaded with >0127 and R1 with >0256 then this would happen.

7D00 SB R2,R1

Left Byte of R2 = >01 (>0127)

Left Byte of R1 = >02 (>0256)

This subtracts >01 (R2) from >02 (R1) which now makes the R1 contain >0156 while R2 remains unchanged.

INSTRUCTION FORMATS

Each instruction is classified into one of nine formats. For example, all instructions which use two operands in the operand field, separated by a comma, and where the operands are general addresses (such as a memory address or a workspace register) are considered FORMAT I instructions.
These are also called TWO GENERAL ADDRESS INSTRUCTIONS.

IE 7D00 A @>837C,R6

The next format are FORMAT II instructions which are JUMP instructions, which transfer control of location or label indicating a memory location.

IE 7D00 JMP LP

FORMAT III These are logical expressions which contain a general address as first operand, separated by a comma from the second operand, which is a WORKSPACE POINTER

FORMAT VI This is a single address instruction that requires only a general address.
i.e. INC INCT DEC DECT

FORMAT VIII Immediate instructions, which require a register as the first operand, followed by a comma and a numeric expression in the operand field.

IE 7D00 LI R5,3

ALSO INCLUDED would be:

7D00 LWPI   >70B8

FORMAT IX   EXTENDED operation instructions. This format includes the extended operation
instructions and the multiplication and division instructions.

Do not worry about understanding the formatting of these instructions yet. As you start to work with the LBL/EA you will get used to what instruction to use and where. When ever you come across an instruction which gives you some doubt about what kind of operands it works with, you can refer to this article later in the series which will have a list of these operands and instructions.

OK. I think that is enough for the old GREY matter to have a go at.

Try out with the LBL some of the above examples.

All you need to do is Load Immediate the values I have given you, then you can try out the results.

IE 7D00 LI R2,>7100

YOUR PROGRAM CODE

Then look at the location with the LBL reference indicator by AORG to > YOUR LOCATION and you will see the change.

Hope to see you at the AGM. Have fun with your programming. I know some of you out there do love to program as I do. That is when I have time.

Also Richard Twyning will also be at the AGM with programming help. He is good at C99. If Mark WILLS comes he is good at most languages.

MEMORY VERY FULL (1K) FCTN/QUIT ....   Trevor Stevens

RAMBLES by Stephen Shaw

December 1996

Oh dear... I really did submit copy for the January issue of TI*MES... but alas in the confusion of the editorial take-over my copy missed out. Sorry...

Greetings once more. Not too much to say again, as no one is asking questions or writing to me, and no responses to the puzzles set in the last two issues.

In days of old there was much to discover about the TI-99/4A, much new to review, and tons of material arriving from user groups all over the world. Not so now! I am doing nothing new with my TI and there are no new products of general interest.

Back when the computer press took an interest in the TI, I had a couple of articles purchased by "Personal Computer News", no longer with us alas. Within that magazine there were regular puzzles set by J J Clessa, who then also wrote for Personal Computer World (still with us, but no longer with the same quality of puzzle).

To keep your minds active in the New Year, some puzzles by J J Clessa...

a. Do you know what 121 and 484 have in common? They are palindromes (the same backwards as forwards). They are both perfect squares. They both have three digits, an odd number. Now for the puzzle -- what is the smallest palindromic number which is a perfect square AND has an even number of digits. (From Personal Computer World, January 1980). I will give an answer if someone asks for it!

b. The sum of consecutive cubes, starting at 1, is always a perfect square.
      For example 1^3 plus $2^3 = 9 = 3^2$.
   However, you cannot start at one for this one... looking for consecutive cubes which add to a perfect square, but the first cube is not to be one! The perfect square is to be the lowest number possible matching these requirements.
This puzzle from August 1982 Personal Computer World.

c. Who was J J Clessa? The name was false, the actual authors initials were L.S.C. -- and no, I don't have an answer to that one!

NEW ADDITIONS TO THE DISK LIBRARY:

LOADMASTER from Bruce Harrison is now into version two, and REQUIRES a two drive system to operate. You leave Loadmaster in drive one, and it will catalogue drive two disks with option to print disk jacket. Each file will be identified as far as possible e.g. whether it is Basic, requires Editor Assembler Option 3, or is a TI Artist picture etc etc. Using LoadMaster you may also load many forms of program without the Editor Assembler module.

On the personal front, my "new" job (from July 1996) is rewarding and time consuming! My father becomes ever frailer and requires more care. My son has been given a word processor by the local education authority to assist him with his schoolwork (he has DCD and there is no English name for it! A report in the FT suggests it is a dysfunctional transubstantia negra, like a Computer with a constantly varying clock speed) and he has now settled very well into secondary school and helps them with their computer problems! Mrs Shaw has found a local part time job as a WP operator.

NET news...

Charles Good (Lima Ohio group): good.6@ose.edu or cgood@brightnet
Paul Urbanus (author of Parsec etc): urb@onramp.net or urb@urbonix.com
Hank Mishkoff (TI software author): hank@WebFeats.com
MICROPENDIUM: MICROPENDIUM@delphi.com

Interesting sites:
Newsgroup: comp.sys.ti
TI-99/4A Shrine: http://www.geocities.cpm/Athens/7374/ti.html

TI modules to download to run on PC with emulator:
ftp://ftp.premierweb.com/pub/systems/ti99/uploads/
(files are archived with PKZIP)

TI software to download in TI format- then needs to be transferred to your TI after downloading to your PC:
Western Horizon Technology: ssonyx.com/wht/
Gary Bowser: io.org/~opanit/

Page 18

TI*MES - Issue 56   TI-User Group U.K.

* caveat emptor- do not under any circumstances give your credit card number or send any funds to Gary Bowser. A number of TI owners including the author have had funds fraudulently obtained and retained by this gentleman (NOT).   *

TI MANUALS ON DISK – alas, on PC disk in a format which requires Windows, either 3.1 or Win95. ALMOST all manuals- including some for unreleased modules such as Crossfire- are now available in Adobe Acrobat format. This format allows for text and graphics to appear on your system exactly as the originals.
The "Acrobat Reader" you need to view the files is FREE, and unlike so many Windows programs the reader only takes up some 2 megabytes of hard disk (Win 3.1 version).
These manuals were scanned and cleaned up, and are "official" in that they are issued with the consent of TI – and with payment to TI. Therefore please respect the copyright as the folks responsible have put a lot of work into them!

An advantage of the Acrobat reader is that is has a fine word search facility. and can also be used to print. It uses .PDF files (Portable Document Format).

The cost is US$2 for the smaller manuals and US$3 for the larger manuals. A single PC High Density 3.5" disk can hold approx. 1 large manual or 3 smaller manuals. Quoted price excludes the cost of disks or postage which must be added. Need a clean copy of the Editor Assembler manual? TI Writer manual? Etc.
etc. Contact is:

CaDD Electronics, 45 Centerville Drive, SALEM, NH, 03079-2674, USA.
Telephone (USA) 603-895-0119 or 603-893-1450.

For a complete (to date) set of manuals for 125 modules C'.) send 5C) PC disks (3.5" HD) with return postage, and indicate version of Windows ! Cost is US$256 plus disks plus return postage.

ADDENDUM- Delivery of my order was prompt, however one disk contained two bad clusters- at the start of a single data file this rendered the disk contents unusable. Also it would appear my Version 2 Acrobat Reader is missing some element used in the TI Manuals as I could not read any of the disks, although my
Acrobat Reader is quite happy reading PCN PDF files. Old Windows users will know this problem proliferates on the PC... 1 now await additional bits of Acrobat Reader from CaDD together with a replacement disk.

The Internet TI Mailing list has a number of German subscribers unfortunately many UK net users have e-mail addresses with USA service providers addresses which makes it difficult to identify UK subscribers- but I don't see any familiar names. I count nine Canadians, six Germans, two Australians, one each
from Italy, Holland, -is .ZA South Africa? -and one from .BE wherever that is! Total on the list was 110 subscribers.

Do you use the Internet - I don't. Some interesting reports in the press of rogue Web sites which you view - but in the process programs or data are without your knowledge loaded into your

computer. No technical details, but it would appear that Microsoft products are involved, and a competitor, Netscape Is apparently immune...

Interesting report of one imaginative user clicks on a link to a really hot site, which turns out to be quite boring, so moves on to other sites by clicking on hot links on the hot site or on pages which follow... unknown, the hot link has taken over control, invisibly, and all links are now made via a different server- using an overseas premium rate telephone number. Whew! Guess who gets the bill...

And on German TV, a demo was given of using this method to install Instructions into a popular software package, Quicken, which together with a link to your Bank account, invisibly transferred funds to an overseas Bank account, without your knowledge or consent...

The papers then indicated that Microsoft were working overtime to remove a security "hole" in their Web access software - whilst competitor Netscape apparently was unaffected. Customers of Barclays Bank who use Quicken are forced to use the Microsoft access route - customers of the Royal Bank of Scotland can use Netscape if they wish.

Of course, no technical details, and nobody is saying that W 3.11 is OK while W95 is not, or that MS Internet Explorer is questionable whilst Netscape is not - not in so many words...

==================

Thanks to John Murphy of DORTIG for submissions to previous puzzles-
PUZZLE THREE-

I gave a listing for this one in Issue 54, which took 50 seconds. John submitted an entry which took 4 minutes 50 seconds.

....

PUZZLE FIVE: TIMES issue 54, page 5.
Find the factorial of 40.

A factorial is the result when you multiply a number by every number less than itself, and a factorial is expressed by using an exclamation mark, so that 5! means FACTORIAL 5, or in long hand,
5 * 4 * 3 * 2 * I where the * means multiply in common computer and calculator parlance. 5! is 120.

This challenge means you have to work out how to do high precision math, and the usual 13 digits the TI works with are not enough. Try the program to accept an INPUT for the number you wish to obtain a factorial of. What are the limiting factors? (If your system balks at 40! try 20!).

Here is John's submission...

```
100 ! factorial puzzle
110 DISPLAY AT(15,3) ERASE ALL: "OUTPUT TO PRINTER? Y/N  "
:: ACCEPT
AT(17,5)VALIDATE("YN"):~PR$
120 IF PR$="V" THEN OPEN #1:"PIO"
130 DISPLAY AT(15,3) ERASE ALL: "PRINT ALL FACTORIALS?
Y/N"   :: ACCEPT
AT(17,5)VALIDATE("YN"):PA$
140 INPUT  "FACTORIAL NUMBER" : NU :: TF,X,Z=1 ::
C1$,C$,B1$="" :: AD=O
150 IF NU=0 THEN 160 ELSE 180
160 IF PR$="Y" THEN CLOSE #1
170 STOP
180 IF NU>14 THEN  X=15 :: Z=2 :: A$="87178291200"
190 FOR N=X TO NU
200 IF N>13 THEN 250
210 TF=TF*N :: A$=STR$(TF)
220 IF PA$="Y" THEN PRINT "FACTORIAL !";N;"=";TF
230 IF PA$="Y" AND PR$="Y" THEN PRINT #1:"FACTORIAL
!";N;"=";TF
240 IF N<14 THEN 390
250 L=LEN(A$)
260 FOR T=L TO 1 STEP -1
270 C$=STR$(VAL(SEG$(A$,T,1))*N)
260 C$=STR$ (VAL (C$) +AD)
290 AD1=AD :: AD=0
300 LC=LEN(C$)
310 IF LC>1 THEN AD=VAL(SEG$(C$,1,LC-1)) :: AD1=AD
320 IF LC>1 THEN C$=SEG$(C$,LC,1)
330 IF AD1>0 THEN AD2$=STR$(AD1)
340 B1$=C$&B1$
350 NEXT T
360 A$=AD2$&B1$ :: B2$=A$ :: B1S="" :: AD,AD1=0
370 IF PA$="Y" THEN PRINT "FACTORIAL !";N;"=";B2$
380 IF PA$="Y" AND PR$="Y" THEN PRINT #1:"FACTORIAL
!";N;"=";B2$
390 NEXT N
400 ON Z GOTO 410,430
410 PRINT "FACTORIAL !";NU;"   ";TF  ::IF PR$="Y" THEN
PRINT #1:"FACTORIAL !";NU;" ";TF
420 GOTO 440
430 PRINT "FACTORIAL !";NU;"   ";B2$ :: IF PR$="Y" THEN
PRINT #1: "FACTORIAL
!";NU;"   ";B2$
```

NOTE: This text has been manually retyped as the original could not be legibly scanned.

REMAIND/80

```
50 ! TI*MES ISS:53 P7 PUZZLE
3 BY S SHAW. SOLUTION BY
JM.DORTIG. TO 6TH DECIMAL
PLACE
90 A=1731 :: B=5363 :: C=717
9 :: D=9903
100 FOR LN-1730 TO 1 STEP -1
110 A1$=STR$(A/LN):: B1$=STR
$(B/LN):: C1$=STR$(C/LN):: D
1$=STR$(D/LN)
112 W=POS(A1$,".",1):: X=POS
(B1$,".",1):: Y=POS(C1$,".",
1):: Z=POS(D1$,".",1)
130 A2$=SEG$(A1$,W,8)
140 B2$=SEG$(B1$,X,8)
150 C2$=SEG$(C1$,Y,8)
160 D2$=SEG$(D1$,Z,8)
170 IF A2$=B2$ AND C2$=B2$ T
HEN 180 ELSE 200
180 IF D2$=C2$ THEN 210 ELSE
 200
200 NEXT LN
210 PRINT "FOUND NUM: IS ";L
N: :"REMAINDERS ARE ";A2$;"
";B2$:C2$;" ";D2$
220 END
```

```
440 GOTO 140
450 END
```

This complex little program found the answer to Factorial 40 (40!) in just 2 minutes 51 seconds. Not bad for a slow computer eh! Count the digits in the answer   now   is our old computer limited to.13 digit math or not?

However John's program uses string handling   a useful way to handle numbers to be sure, but not the fastest. Instead of storing values in strings it is faster to use arrays   but unlike strings we need to dimension our arrays, and thus need to have a rough idea how many digits we are going to need! Factorial 40 uses 48 digits (not bad precision !!) but lets leave room for higher number factorials...

try this one...

```
100 DIM F(100) :: L=100 :: INPUT  "NUMBER":N
110 FOR X=2 TO L :: F(X)=0 :: NEXT X :: P,F(1)=1
120 FOR M=1 TO N :: C=O :: FOR X=1 TO P
130 F(X)=F(X)*M+C :: C=INT(F(X)/10)  :: F(X)=F(X)-10*C ::
NEXT X
140 IF C=0 THEN 160
150 P=P+1 :: F(P)=C :: C=INT(F(P)/10)  :: F(P)=F(P)-10*C ::
GOTO 140
160 NEXT M
170 S=0 :: FOR X=P TO 1 STEP -1 :: IF F(X)>0 THEN 190
160 IF S=0 THEN 210
190 PRINT STR$(F(X));
200 S=I
210 NEXT X
220 END
```

This time the factorial of 40 is produced in around 45 seconds. Very creditable math for such an old computer - and this does serve to demonstrate that you are NOT limited to any level of mathematical accuracy when using computers.

Letters to the Editor
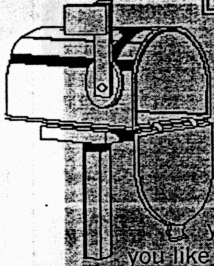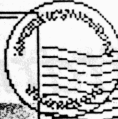
Welcome to TWO fixtures in your new and improved TI*MES newsletter. "Letters to the Editor" is, as it says, a place were your letters will be published so you can voice your opinions on the group, and tell us if you like it or if you don't like it!

We don't want this section to be a blank page, and in fact, we don't mind if it turns out to be many pages, because we want it to contain your letters, and if you don't send us any, then you'll have a nice big picture of a postage stamp to look at!!!

What do you want to see in the magazine? We already know you like to have plenty of listings to type in, so you will hopefully find enough in this issue to keep you going.

My fellow co-editor and I have many ideas for improving the magazine, and the way things are run. If you have any questions, then don't hesitate to contact either myself or Ian.

The BBS is running like a dream at the moment, and we are always looking for new callers.

Hopefully my car will be much lighter when it returns from Derby, since I've got a boot full of 2400 baud modems to give away! Watch this space, or write a letter which will fill it!!!