SASKATOON TEXAS INSTRUMENTS COMPUTER CLUB

NEWSLETTER NOVEMBER 1984
NEXT GENERAL MEETING AT:
SASKATOON REGIONAL COMMUNITY
COLLEGE, ROOM 114
145 1st Ave. North
7:00 p.m. Nov. 5th, 1984
December 3rd, 1984

TENTATIVE DATES (1985):

JANUARY 7th, 1985
February 4th, 1985
March 4th, 1985
April 1st, 1985
May 6th, 1985

NEWSLETTER COMMITTEE

KEN READ 3713 DIEFENBAKER DR. 384-3844
MIKE BENDEL 146 STONE TERRACE 382-5993
FRANCIS X. GASTON 407-2010 22nd st. w. 978-0182

TEXAS INSTRUMENTS CANADIAN TOLL FREE NUMBER: 1-800-268-3866

TI REPAIRS:

TEXAS INSTRUMENTS
SHELLY ROAD, RICHMOND HILL, ONTARIO
L4C 5G4 postage prepaid 10 days

NEWSLETTER COMITTEE EDITORIAL

As we progress, it becomes a little more difficult to always think of something new for the newsletter. Therefore submissions are greatly needed; So if you have any new products, or have short programs, or would like to review even old ones, please submit them and this should help us out immensely.

Newsletters will be distributed at the meetings in order to conserve costs. Therefore submissions are due at least five days before every meeting. Any outstanding newletters will then be mailed out to those who did not attend. Newsletters will be distributed to fully paid members only. Our current memberships now stands as follows:

Raymond Casavant, 221 Campion Cr., S7H 3T8, 374-3366

Gordin Millham, 221 Acadia Pl.,S7Y 3W3, 374-3387

Rich Lacroix, 1811 Wilson Cr., S7J 2N8, 373-1522

Mike Boyachek, 124 110 St., S7N 1S2, 374-4580

Dave Brunnen, 3673 J.A. Macdonald Rd., S7L 4X7, 384-2175

Harry Caruk, 417 Dickey Cr., S7L 5P2, 244-8379

Francis X. Gaston, 407-2010 22nd St. W., S7N 0V1, 978-0182

Randy Gaw, 106 Wark Pl., S7M 4E1, 384-8066

Doug Gibbens, 1030 Ave. J. N., S7L 2L4

Dean Jamieson, 1524 Shannon Cr., S7H 2T7, 373-3548

Randy Mortenson, 947 Coppermine, S7K 5H8

David Oconner, 305 Barber Crt., S7M 1L3

Balram Rajput, 234 Thain Way, S7K 5T2, 934-5896

Ken Read, 3713 Diefenbaker Dr., S7H 3W4, 374-2823

Upcoming in future issues: Reviews on Microsurgeon, Pole Position, Miner 2049er, and the continuation on the Disk Drive articles. Hardware reviews will also be provided on printers. So if you wish to do a review , now is the time to do one on your printer. If you feel reluctant, please bring a copy of a printout by your printer. Your recommendations will also be of interest. Of interest to all members is that Election of Officers will be upon us soon. So please make an attempt to attend the next meeting in December.

OCTOBER MEETING "MINUTES"

Well, at least 12 members attended the October meeting.
Two of the members brought complete systems at the meeting.
The following occurred: (events are not necessarily in order)

1. Membership fees were collected by Doug Gibbens and receipts
provided.

2. Two newer members joined the club. They were Rick Lacroix
and Dennis Hancock. (Dennis brought his whole system with him).
Also other new members were Gordin Millham and Raymond
Casavant from the previous meeting.

3. Harry Caruk had brought along the programming language, FORTH.
Also a thick manual accompanied it. No doubt a demo in a future
meeting since everyone was interested.

4. Demo of Winging-it, a flight simulator, was shown by one
of the members.

5. Printouts from a Mannesmann Tally printer provided by Rick
Lacroix. It provides a comparison with other printers.

6. Harry Caruk's Demo on an MBX voice recognition system
resulted in rave reviews. Everyone loved it. "I see you!!"

7. Mike Boyachek brought copies of a basic and Extended basic
compiler. A copy is donated by Mike to the "Hard copy library".

8. Gordin Millham provided the munchies: selling chocolates
for a good cause - School for the Deaf. Many members contributed
and gained, both in good will and calories.

9. A copy of Unisource Electronics new catalogue was shown.
Everyone were so interested, that a bulk order by eight of our
members was issued for more catalogues.

10. Demo of word processors using extended basic was provided.

11. Demos of entertainment modules were also provided.

12. Current newsletters were distributed. Maybe this will be
the best way since cost can be kept at a minimum.

13. Mike Bendel had copies at the meeting of his system. Prices
enclosed in this newsletter. He is trying to sell the system
all at once.

14. Everyone tried to get Ken Read's printer to work. Even with
everyone's help, we still did not succeed. It appears
that no one can remember the correct syntax or did not have
their manuals handy. Next time!!!!

Meeting adjourned.

Drive for Diskettes

In Part One of this article we looked at the hardware you need to run TI's Disk Operating System (DOS). In Part Two we'll talk about DOS software- the computer code itself- how it works and how you can use it for sequential and relative file housekeeping.

THE DISK OPERATING SYSTEM   The DOS is the computer code you need to handle both disk and file housekeeping. It is the programming that makes soft sectoring work and manages files as part of your BASIC programs. Before we get into the details, let's go over how disk usage is organized. First of all, one major difference between tape and disk is that each disk has a permanent name.   Even though you can reference as disk drive as DSK1 (similar to CS1), you can also reference a particular disk by its name. To illustrate, let's look at a disk called BASIC01. We know that it is broken up into tracks and sectors, but that is how the disk is hardware configured. What we need to find out now is how it is configured in terms of software or by the DOS.
     Just as you can have more than one program stored on a single magnetic tape, you can store many programs (or files) on BASIC01. The DOS reserves the first two sectors of the disk for a directory. This is similar to a book's table of contents. It keeps track of all the data located on the disk.   There is a directory entry for each program or file located on the disk, with enough room for 128 different entries. Suppose you want the computer to get a program called DOOMCASTLE (assuming that my disk called BASIC01 is located in disk drive 1). You can do it by entering OLD DSK.BASIC01.DOOMCASTLE
(which means that I am asking the DOS to look for BASIC01 on any disk drive).

     Once the DOS locates the proper disk, it searches the directory for the program. The directory entry points to the exact track and sector where the program can be found. The read/write head moves directly to that track, and as soon as it gets to the proper sector in the track, it starts reading. This is one reason why disk access is so much faster than tape cassettes. The disk does not need human intervention to turn it on or rewind it, and it doesn't have to read through a lot of unnecessary data before it gets to the place where the information you want is stored. In addition, a disk drive is capable of reading this data and passing it on to the controller (and from there into the computer) about 100 times faster than a tape recorder can.
     If you want to write out a sector of information to BASIC01, the DOS follows the same procedure: First, it locates the disk; then it finds the first available track on the disk (also from the directory). Finally, the read/write head moves to that area, and the disk drive starts writing out the data. And, after the data is written out, it is automatically reread and immediately verified. That's a handy feature of writing data to a disk: no need to rewind and do a separate verification as in tape!

FULL SECTOR BLOCKING
     In order to minimize disk reads and disk writes, the DOS takes 2K of RAM away from your 99/4A and uses them for buffers (one for each file). When you write data to a part of a sector, it is really written to a buffer in RAM. Only when the information for an entire sector is ready, does the DOS actually write out to the disk. Likewise, a buffer in RAM will hold the entire contents of a sector when you want to read data from it. Then,

if you want to read more data from the same sector, the DOS gives you the
data from the buffer without having to reread from the disk. This technique
is called full sector blocking, meaning that the operating system always
processes programs or data one sector at a time.

If the data uses more than 255 bytes (as does our example program,
DOOMCASTLE), it must be read or written a sector at a time. If the information
uses less than a full sector, you must be sure to CLOSE your data file or
type BYE before leaving the program. Then the unused bytes in the sector
will be filled with zero bits, and the DOS will understand that it is to
write out the data to the disk even if the buffer is not filled a full sector's
worth. No other data file will be able to use that sector.

TWO PARTS OF THE DOS
The TI-99/4A's DOS has two major components.: the Disk Manager Command
Cartridge (which comes free with each disk controller) and the code inside
your TI-99/4A console. The Disk Manager, which contains the program code
for all of the housekepping functions not needed to run a program, performs
several jobs. It will initialize a disk by establishing the sectors on each
track and filling them with an initialized pattern (there's that "soft-sectorin
g"
again!). It can also copy files between disks for backup, name or rename
a disk or file, delete (remove) files or programs from a disk, and print
the contents of the disk directory. In addition, it gives you the means
to set a deletion protect switch which prevents you from accidentally deleting
or replacing a file or program. This is a neat function seldom found on
microcomputers. The functions and operations of the Disk Manager are spelled
out in detail in the manual that comes with the disk controller. The manual
also outlines how to set up the controller and your TI disk drives, initialize
disks and name them, and how the various housekeeping functions operate.
It also provides a good background on file- and disk-naming conventions.
I won't repeat that information here, but I will say that as in most TI
manuals, the information is straightforward and it includes enough examples
to give you a good idea of how things work.

The second major DOS component, the code, allows you to load and save
programs and read and write data files as part of BASIC programs. I'm not
going to go into all the features of this part of the DOS. Instead, I will
spend the rest of the article describing how the disk files work.

FILES AND DATA
A file is a grouping of data stored outside the computer. It is composed
of one or more individual groups of data called records. The easiest way
to explain this is to give an example. Suppose that I am the secretary of
a bowling league and want to maintain a scoring history for every bowler
in the league. A file would be made up of records which hold information
about each individual bowler. If there were 60 bowlers in the league, I
would need 60 records in the one file. Each file has a name, and we will
call this file BOWLERS.

File data used in BASIC programs can be stored in either DISPLAY or
INTERNAL format. DISPLAY stores data exactly the way it would look on a
monitor or in data statements. All letters and numbers are stored as individual
characters. In order for our program to perform operations with this data,
the computer will first have to convert it to INTERNAL format. For this
reason, we are better off saving all data in INTERNAL format to begin with.
All numbers in INTERNAL format will take exactly 8 bytes no matter how large
they are. This may help us estimate how big each record that holds the data

will be. In all my programs, I use only INTERNAL format data.

HAving made that decision, we tell the DOS in the OPEN statement what kind of file we are going to use and what option we'd like for processing the files. Depending on how we want to store and retrieve our records, we will tell the DOS to file our bowlers' information in either SEQUENTIAL or RELATIVE data files.

Considering the length of this article, the last part will be in next months newsletter. This final part will cover both SEQUENTIAL and RELATIVE file processing.   Ken Read

SHORT PROGRAMS

The following programs were obtained from old issues of the 99er Home Computer Magazine. Each is an individual program so they can run independantly of one another.

```
100 REM BEE
110 N=1
120 CALL SOUND(-99,RND*8+110,RND*10)
130 N=N+1
140 IF N=75 THEN 150 ELSE 120
150 REM END .  —
300 REM DOOR CHIMES
305 FOR X=1 TO 3
310 FOR A=0 TO 30 STEP 5
320 CALL SOUND(-99,698,A,1924,A)
330 NEXT A
340 FOR A=0 TO 30 STEP 5
350 CALL SOUND(-99,554,A,1527,A)
360 NEXT A
370 B=RND*1000
390 FOR A=1 TO B
400 NEXT A
410 NEXT X
500 REM ENGINE
510 FOR N=1 TO 8
520 CALL SOUND(60,220,8,-5,0)
530 CALL SOUND(60,220,8,-5,5)
540 NEXT N
550 CALL SOUND(80,220,8,-5,0)
  0 FOR F=1000 TO 5000 STEP 20
570 CALL SOUND(-99,111,30,111,30,F,30,-8,0)
580 NEXT F
590 FOR F=4000 TO 800 STEP -50
600 CALL SOUND(-99,111,30,111,30,F,30,-8,0)
610 NEXT F
700 REM FOOTSTEPS
710 N=1
720 X=INT(RND*5)
730 IF X=2 THEN 790
740 CALL SOUND(5,-3,5)
750 CALL SOUND(30,-7,20)
760 CALL SOUND(500,-7,30)
770 N=N+1
780 IF N=30 THEN 810 ELSE 720
790 CALL SOUND(60,-7,20)
800 GOTO 760
810 REM END
900 REM SHORTWAVE RECEIVER
910 N=1
920 F=RND*15000+110
930 A=RND*30
940 CALL SOUND(-99,111,30,111,30,F,A,-8,30-A)
950 N=N+1
960 IF N=100 THEN 970 ELSE 920
  0 REM END
```

```
1100 REM SIREN
1110 N=1
1120 FOR F=700 TO 900 STEP 5
  30 CALL SOUND(-99,F,0)
1140 NEXT F
1150 FOR F=900 TO 700 STEP -8
1160 CALL SOUND(-99,F,0)
1170 NEXT F
1180 N=N+1
1190 IF N=4 THEN 1200 ELSE 1120
1200 REM END
1300 REM SWORD FIGHT
1310 N=1
1320 FOR A=0 TO 30 STEP 15
1330 CALL SOUND(-99,1000,A,3250,A,6750,A)
1340 NEXT A
1350 FOR D=1 TO RND*200
1360 NEXT D
1370 N=N+1
1380 IF N=30 THEN 1390 ELSE 1320
1390 REM END
1500 REM RADIO TELEPRINTER
1510 N=1
1520 CALL SOUND(22,2975,0)
1530 FOR D=1 TO 5
1540 S=850*INT(RND*2)
1550 CALL SOUND(22,2125+S,0)
1560 NEXT D
  70 CALL SOUND(31,2125,0)
 580 N=N+1
1590 IF N=30 THEN 1600 ELSE 1520
1600 REM END
```

MILLER GRAPHICS
1475 W. CYPRESS AVE.
SAN DIMAS, CA. 91773

I have just recently received a brochure and a sample newsletter from Miller Graphics. They distribute and develope various books and software for the TI99/4A.

Miller Graphics have developed at least six entertainment graphics programs for the Extended Basic module. They have also developed three books, namely "The Fundamentals of Microcomputer Design","Software Development", and "Smart Programming Guide for Sprites". They are also well known for their outstanding newsletter called "The Smart Programmer". It provides a deeper insight into the TI and has an extensive growing list of memory mappings. This alone is worth it.

The price of the newsletter is $16.00 (us), while the programs are $14.95 on cassette and $17.95 on disk. All in U.S. currency.

Francis X. Gaston

ADVERTISEMENT

- Many products are still for sale at Eatons. Prices vary greatly, so rummage.
  - TI monitor cables are now available from Supervalu for $10.95.
  - Head Cleaning Kits are available for $5.95 also at SuperValu.
  - Also available are Disk Drive controllers. No price.
  - Mike Bendel, 683-0999, is still selling his unit at the lower price of $1500 for the whole system. They are as follows:

Item:	Cost:
TI 99/4A Keyboard	100.00
Expansion box - drive	600.00
32K memory expansion	200.00
RS232 interface	150.00
TI Colour Monitor	500.00
Extended Basic	150.00
Multiplan	150.00
TI Writer	150.00
Personal Real Estate	20.00
Parsec	20.00
Music Maker	20.00
Hunt the Wumpus	15.00
Othello	15.00
IUG Data Base 300	20.00
20 - 5¼ Diskettes	80.00
Speech Synthesizer	80.00
Cassette Cable	20.00
Last Word on the 99	18.00
Editor - Assembler	15.00
Mastering Multiplan	15.00
Home Comp. Program	12.00
99/4A Reference Guide	14.00
Assorted Magazines	30.00
Assorted Hardcopies	30.00
Desk with typing wing	250.00

TI 99/4A Key Board - 100.00
- Francis -

- Thermal Printer - Ken Read

Edmonton 99'er Comp. U.S.
P.O Box 11983
Edmonton, Alt
Canada T5J 3L1
ATTN: S. Livingston