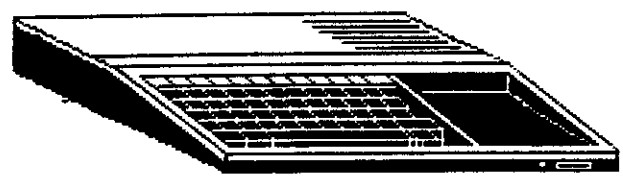
A publication of the TI 99/4 and 9640
Ogden users group inc.

OGDENS USERS GROUP

TI 99-4A



THE CLUB THAT REFUSES TO SAY GOODBYE

JULY 1989

Portions of this Newsletter has been done in Page Pro 99. So if you like what you see come to user Group Meetings and see what you think of this neat Prog

SEE YOU ALL THERE MEL

TI-BASE - From INSCEBOT
TUTORIAL 9.1.1 By Martin Smoley
NorthCoast 99'ers - April 15, 1989

COMPLICATED DATABASE OPPERATIONS AND GRAPHICS

It's time we started to do more complicated chores with TI-Base. With this tutorial I am starting a mini-series. In this mini-series we will create a set of CFs and DBs for a computer club disk librarian who handles mail order requests from members who live in other states. The system should rapidly locate a members name in the club DataBase (Db), print out the three labels you see below (To:, From:, and CAUTION) and keep track of who received disks and the club disk inventory. The system will keep us informed of the disk inventory and start telling us to order more disks at a pre-set quantity. At this point we will only keep the quantity of disks shipped to a person and the date. This will not include information on the disk name or type of disk. I want to save some information and demo how it's done, but I don't want to turn the operator into a data entry person either. I have the system working at this time, but I may make some changes as we go or by next months tutorial. I will cover as much as possible each month without stretching my page limatations too much.

To:

Exp. Date 88/09
Raymond (Slim) B. Whitman
2574 East 254th.
Eastlake OH. 44094

From:

Martin A. Smoley
6149 Bryson Drive
Mentor, OH 44060

** CAUTION **

COMPUTER DISK
DO NOT BEND OR FOLD
DO NOT X-RAY

```
* Copyright Martin A. Smoley 1989
*                                DSKSHP1
CLOSE ALL
     DO DSK2.PREP1
     DO DSK2.DSKSCR1
LOCAL SEL2 N 5 0
LOCAL MORE C 1
LOCAL TEMP1 C 60
LOCAL ANS N 3 0
  SELECT 5
   USE DSK2.GRF1
  SELECT 4
   USE DSK2.MSRET
  SELECT 3
   USE DSK2.DSKINV
  SELECT 2
   USE DSK2.SLSREC
  BOTTOM
  SELECT 1
   USE DSK2.TNAMES
REPLACE MORE WITH "Y"
WHILE (MORE = "Y")
  TOP
 WRITE 23,6,"ENTER NM "
 READ 23,17,SEL2
  IF SEL2 = 0
    CLOSE ALL
    RETURN
   ENDIF
     CLEAR
   FIND SEL2
  IF (NM = SEL2)
    DO DSK2.DSKNAP1
    DO DSK2.INVUPDT
  ELSE
    WRITE 23,4 "Number Not Found"
    WAIT 3
  ENDIF
 WRITE 23,4,"FIND another Y/N"
 READSTRING 23,23,MORE
 CLEAR
ENDWHILE
CLOSE ALL
 DO DSK5.SETUP
RETURN
*
* DSKSHP1    Save as DSKSHP1/C
* *******    Ver. 2.01 03/31/89
* Find NM  using "FIND"
* Print a label and save record
*****************************
```

Continued Next Page.

TNAMES

REC	NM	LN	FN	MI	SA
0002	1	Aardvark	Grant	E.	9995 State Rt. 84
0003	2	Aardvark	Willard	J.	No Newsletter
0005	3	Jones	Quincy	W.	37285 Burgandy Lane
0000	4	Smoley	Martin	A.	6149 Bryson Drive
0004	5	Vivannovitch	Elexxie	I.	111 E. 98th. St.
0001	6	Whitman	Raymond (Slim)	B.	2574 East 254th.

CT	ST	ZP	PH	XP	GP	ID
Geneva	OH	44014	1-465-9876	89/02	NOCO	0717851
	OH		1-465-7689	89/09	NOCO	0717852
Mentor-on-the-Lake	OH	44060	257-1029	89/08	NOCO	0820871
Mentor	OH	44060	216-257-1661	90/02	NOCO	0713831
Cleveland	OH	91023	541-5415	89/05	NOCO	0712881
Eastlake	OH	44094	951-2345	89/09	NOCO	0921861

TI—BASE - From INSCEBUT
TUTORIAL 9.1.2 By Martin Smoley
NorthCoast 99'ers - April 15, 1989

In this series I will attempt to include all the data you will need to complete this task. This will require some redundency that I hope will not boar the more advanced TIB users. I am printing three labels in each print cycle because I always seem to need return address and CAUTION labels. OK, let's get started.

You should recognize the database TNAMES. I have updated it and filled in some blanks. I use it because it is a direct substitute for the NorthCoast DB, but smaller. (NOTE: The only real name is mine.) Two items of importance are NM and ID. NM is an N type field with a width of 5 and 0 decimal places, while ID is a C type field with a width of 7. In order to get the desired end product I SORTed TNAMES ON LN,FN before I placed the numbers in the NM field. I then entered the NM field and SORTed ON NM. This was because I wanted the names in LN,FN order, but they had to be sorted on NM to allow me to use the FIND function on the NM field. This was all covered previously, I believe it was around Nov./Dec. 1988. FYI: A field type can be changed at will using MODIFY STRUCTURE. You can change a C type to an N type or vice versa, but don't change any field lengths or you'll lose the data. Also, I do not recommend changing a C)haracter field that contains names to a N)umeric field. The field should already contain numbers before it is changed. If you want to experiment with this idea, use a database you can afford to lose.

```
*                                   PREP1
CLEAR
CLOSE ALL
SET PRINTER=RS232.CR.LF.DA=8
SET HEADING OFF
SET RECNUM OFF
COLOR WHITE DARK-BLUE
SET TALK OFF
RETURN
*
*   Pre-Program Preparation
*
* PREP1  Save as  PREP1/C
* *****


CLEAR
WRITE 3,9,"This section Locates a record"
WRITE 5,9,"using the NM field in the"
WRITE 7,9,"TNAMES Database. It then"
WRITE 9,9,"displays the name and address"
WRITE 11,9,"and asks how many disks to"
WRITE 13,9,"be shipped.  It also keeps a"
WRITE 15,9,"running inventory of disks"
WRITE 17,9,"in stock and shipments."
RETURN
*
*  DSKSCR1      Save as DSKSCR1/C
*  *******  Info Screen     12/1/88
```

DSKSHP1 is the main CF. To start this small system you would place the disk containing all the needed files in disk drive #2 and type DO DSK2.DSKSHP1 <E>. The TIB processor will find DSKSHP1 and start executing the commands one line at a time. The first thing TIB will do is CLOSE ALL Dbs. The next command will DO the PREP1 CF on DSK2. The PREP1 CF is my general purpose preparation file. It resets the defaults for a particular set of CFs. The only line in PREP1 that I feel is important is SET PRINTER=RS232.CR.LF.DA=8. I am using one of the original TI 99/4 Impact Printers to do the labels. This printer runs off the RS232 port and not PIO. It was also necessary to place Dip switch #1 of the SW2 set to the OFF position for graphics mode (see your printer manual), and last, it is also necessary to send the command DA=8 in the PRINTER statement. This was all necessary to set up my printer for the graphics I intend to print on the second and third label.

> The ability to print graphics, which can now be done with Version 2.01, will create an exciting new area of capability for the TI-Base user.

DO DSK2.DSKSCR1 puts up the information screen in the lower left hand corner of this page. It can say anything you want, or you can leave it out of the CF altogether. The next four lines in DSKSHP1 are not important, but then again they are important. Sounds like I'm confused, "HUH". Well, I'll explain. As far as I can tell the local variables are limited to about seven names with a size that is currently about 256 bytes. You can increase the size somewhat with the SET command but the point is that this space is a precious commodity. It seems like I run out of local space every few minutes. So, in the future I will try to use one or more databases to temporarily store information that TIB needs. As you will see in this tutorial the information is as readily available in a Db as is would be in LOCALs and I can store as many as I need.

CRTD 04/03/89 CHNGD 04/05/89

FIELD	DSCR	TYPE	WIDTH	DEC
1	FR	C	006	
2	TO	C	003	
3	NAME	C	030	
4	STREET	C	030	
5	CTSTZP	C	030	
6	CTN	C	020	
7	CD	C	030	
8	DNB	C	030	
9	DNI	C	030	
10	MS1	C	010	
11	MS2	C	010	
12	MS3	C	010	
13	MS4	C	010	
14	CD1	X	020	
15	CD2	X	020	
16	CD3	X	020	
17	CD4	X	020	

000 1 MSRET 00000/00001

CRTD 04/06/89 CHNGD 04/06/89

FIELD	DSCR	TYPE	WIDTH	DEC
1	GRFNM	C	010	
2	GR1	X	080	
3	GR2	X	080	
4	GR3	X	080	
5	GR4	X	080	
6	GR5	X	080	
7	GR6	X	080	

000 1 GRF1 00000/00001

CRTD 03/31/89 CHNGD 04/06/89

FIELD	DSCR	TYPE	WIDTH	DEC
1	RTOT	N	005	00
2	LDT	D	008	
3	PDT	D	008	

000 1 DSKINV 00000/00001

Continued Next Page.

TI—BASE - From INSCEBOT
TUTORIAL 9.1.3 By Martin Smoley
NorthCoast 99'ers — April 17 1989

The statement SELECT 5 means select the number 5 work area, or as I like to call it, "slot 5". We then open the GRF1 database, which is located on disk drive #2 in slot 5, with the statement USE DSK2.GRF1. We then SELECT slots 4,3,2 and 1, and open MSRET, DSKINV, SLSREC and my old favorite TNAMES respectively. I SELECTed 1 last because that is the slot I want to be in when TIB is ready to start performing its tasks. I'll get into it more later, but for now the DBs look like this. GRF1 is the DB where I will keep my graphic data to make up the OHIO and DISK graphics. MSRET is a DB that contains Martin Smoley's RETurn address and a bunch of other tricks which I will explain as we go. DSKINV, of coarse, is our DiSK INVentory, and SLSREC holds the SaLeS RECords. TNAMES holds the name and address list and must be sorted on the NM field. I have listed the STRUCTUREs for three of these Dbs in the lower right corner of sheet 9.1.2 to help you understand and enter the information needed. I have squashed them slightly to make them fit in the corner, but all the information is there. Let's look at GRF1, my graphic Db, first. It contains 7 fields. Field #1, GRFNM, is a C)haracter field with a maximum width of 10. This field will hold our graphic name, such as OHIO or DISK. This Db must be SORTed ON GRFNM to allow TIB to FIND a graphic by its name. Fields 2 through 7 are X-type fields, and will contain our graphic and other interesting stuff. I have listed the data from GRF1 at the bottom of this page. It was quite confusing to me at first, so don't expect to grasp the idea immediately. Special Note: (04/11/89), I told Dennis Faherty about these new discoveries in graphics last night and he was amazed. Until then he had not grasped the unbelievable potential of the X-type field even though he created it. The X-type field is very special. You type in normal ASCII character like 123ABC. The character must be entered in a very special pattern, but the actual entry is easy. The character must be entered in pairs because TIB will interpret them as Hexidecimal numbers. For example, the pair of characters 1B would be interprited by TIB as Hex for the number 27 and 27 to your printer means ESC, or escape. We now have the basics for all printer control codes.

I am going to explain some of the data in GRF1 i condensed form to give you a quick look at what is going Note: The ")" is generally used to denote a Hex number. line GR1 (under DISK) goes like this. 1B =)1B = 27 = ESC, =)33 = 51 = 3, 18 =)18 = 24, or (ESC)"3"CHR$(24), 24 be 24/216", right out of the printer users manual. To my prin that phrase means, set line spacing to 24/216ths inch. This needed for graphics. Next, 1B =)1B = 27 = ESC, 4B =)4B = = K, 1F =)1F = 31, 00 =)00 = 0, or (ESC)"K"CHR$(31)CHR$(To my printer this says, print normal-density graphics, bytes of graphic data will follow. The printer will t consider the next 31 bytes of data to be part of the grap itself. As you can see we are taking the characters two a time, 1B, 33, 18, 1B, 4B, etc. Each of these small units one byte. This means that the printer will actually need characters from this data field to make a total of 31 byt "I hope this is helping and not confusing you." So the FF, D5, etc. are used to print the first line of the DISK graph At this point I have designated 6 lines for a graphic in Db, but the biggest graphic is OHIO which takes only 5 lin I have temporarily placed the characters 0D0A at the beginn of lines GR6. 0D = ASCII code 13, or CR, and 0A = ASCII c 10, or LF. If printed, GR6 will produce a Carriage Return a Line Feed. The reasoning is this. Printing GR1 will force the printer into action. Apparently a LF is needed cause the data in the printer buffer to be printed on paper. GR6 contains both a CR and an LF, so printing out complete disk graphic would go like this: PRINT GR1,GR6,G GR6,GR3,GR6,GR4,GR6. In DISK, GR5 graphically prints a bl Remember GR6 contains CR,LF. We also have the ability to the TIB system CR, LF. They are entered like this: PR GR1,(CR),(LF),GR2, (CR),(LF),GR3,etc. This is too much typ for me, but the functions can be used together to interesting advantage. The line PRINT GR1,(CR),GR1,GR6,G (CR),GR2,GR6,etc. will produce a double struck graphic. will be printed, the (CR) will return the carriage to beginning of that line without a line feed, the second GR1 then be printed for a double strike affect, and the GR6 then send a CR, LF to set up for the next graphic line. A progress in this area I don't see any reason why it wouldn' possible to produce sales reports or purchase orders. T printout sheets can incorporate letterheads with graphics could be anything from disks, as I used here, to intri company logos. Once the graphic database is created you use it anywhere.

Next Page.

0000 **Database: GRF1**

GRFNM DISK
GR1 1B33181B4B1F00FFAAD5AAD5AAD5AAD5AAD5AAD5AAD5AAD5AAD5AAD5AAD5AAD5AADFABF80000
GR2 1B33181B4B1F00FFAA55AA55AA55AA55AA57AF5CB858B858B85CAF57AA55AA55AA55AAD5AAFF0000
GR3 1B33181B4B1F00FFAA55AA55AA55AA55AAD5FA351A1F1B1F1A35EAD5AE5FAE55AA55AA55AAFF0000
GR4 1B33181B4B1F00FFAB55AB55AB55AB55AB54AB55ABFDFFFDAB55AB54AB55AB55AB55AB55ABFF0000
GR5 1B33181B4B1F0000000000000000000000000000000000000000000000000000000000000000100
GR6 0D0A0000000000000000000000000000000000000000000000000000000000000000000000000000

0001
GRFNM OHIO
GR1 BB33181B4B2100070404040404040404040402020101000000010101020202040404080810 1F00
GR2 1B33181B4B2100FF0000000000000000000000000080808000000000000000000000000000FF00
GR3 1B33181B4B2100FF00000000FF818181FF00FF101010FF00FF00FF818181FF00000000000106F800
GR4 1B33181B4B2000FC060202010100000000000000000000000000000302060810202040800001 0000
GR5 1B33181B4B200000000000000080808040402040408070080830C0000000000000000000000010000
GR6 0D0A0000000000000000000000000000000000000000000000000000000000000000000000000000

TI-BASE - From INSCEBOT
TUTORIAL 9.1.4 By Martin Smoley
NorthCoast 99'ers - April 15, 1989

The next Db is MSRET. This is a good example of how to use a Db to store information that would have previously taken up LOCAL variable space. I have used all 17 fields. Even though I don't need them at this time, I will probably use them later. You will notice that I have mixed C)haracter fields with X-type fields in a normal database. This is a very important development. If you need to do a very special printout or you have paper size restriction etc., special control codes could be saved with individual data records to automatically change the printer settings for special fields. A simple example of this would be a Db containing 100 names, which must be printed weekly for inventory. 90 of those names have 80 characters or less, and the other 10 names have from 80 to 110 characters. The standard form you print on has 85 spaces to print the names. You can include special control codes with certain names to change the print pitch, or micro-justification if you have a very expensive printer, and the names will fill the space perfectly every time. I am probably confusing you with ideas, so let's get back to the subject. MSRET is self explanatory. As you will see late, I use these fields to print labels and screen messages. There is only one record in this Db, the one listed below. A more complicated system could use more records for different labels and different messages.

Database: MSRET

1	FR	From:
2	TO	To:
3	NAME ...	Martin A. Smoley
4	STREET .	6149 Bryson Drive
5	CTSTZP .	Mentor, OH 44060
6	CTN	** CAUTION **
7	CD	COMPUTER DISK
8	DNB	DO NOT BEND OR FOLD
9	DNX	DO NOT X-RAY
10	MS1	*******
11	MS2	* ORDER *
12	MS3	* MORE *
13	MS4	* DISKS *
14	CD1	1B331800000000000000
15	CD2	1B332400000000000000
16	CD3	00000000000000000000
17	CD4	00000000000000000000

The screen below is the structure of SLSREC. It will be used to save 3 items. The ID number of the member requesting disks, the date the disks were shipped and the quantity of disks that were shipped to that ID number, or member.

CREATED 04/01/89 CHANGED 04/05/89

FIELD	DESCRIPTOR	TYPE	WIDTH	DEC
1	ID	C	007	
2	SDT	D	008	
3	QTS	N	004	00

000 1 SLSREC 00000/00063

The SLSREC Db should be created, but left empty. The CF will fill in the data automatically each time disks are shipped. TNAMES is opened last, and I hope , needs no further explanation. The statement REPLACE MORE WITH "Y" will get us into the WHILE loop. "I hope that most of this standard stuff is familiar to you by now." You are then asked to enter a number for the NM field you wish to FIND. If you enter a zero, all databases will be closed and the CF will be ended. That's a quick way out that I may modify later. If you enter a good number it will be found by TIB, the statement IF (NM = SEL2) will be true and DO DSK2.DSKNAP1 will be executed. DSHNAP1 is the Command File I created to display the name, if found, on the screen so you can decide if it is the name you want. However, I am going to leave that until next months tutorial.

TI-Artist Instances To TI-Base

Wes Richardson has said that he would attempt to write a program to convert TI-Artist Instances into a format that can be imported into a TI-Base Database using the Convert function. Knowing Wes' past record I would eliminate the word attempt from that statement. The creation of this type of program will open up a new world to the TI-Base user. There are currently large quantities of graphics available to everyone. There are also many program arrund to change those graphics to TI-Artist Instances. A program to change Artist Instances to data in a TI-Base database would give TI-Base users more tools in this area than had ever been imagined before. If everything goes well, the conversion program will be published in the NorthCoast Newsletter in the next couple months. Hopefully that timing will bring the program out at the end of my mini series on graphics and exactly when you are ready for it.

HORIZON RAMDISKS

I must throw in a plug for Bud Mills. I forced myself to do this graphic series on a standard disk drive to experience the speed of the system. I must say that it is too slow, and too noisy. If you have a real need for a database system like TI-Base then you will probably put your TI through some heavy use. In that case a Horizon Ramdisk is the only way to go. Bud Mills has been unbelievably helpful and supportive to me for the whole time I have known him, and I hear the same story from other people. His Ramdisks are a top quality item, they are super fast compared to a normal disk drive and they don't make a sound. If you're interested, I recomend that you call Bud at (419) 385-5946 and get further information.

THE MULTI USERS GROUP CONFERENCE

If everything works out I hope to see many of you at the Lima, Multi Users Group Conference on May 20. I am looking forward to the possibility of meeting many of the users I know by mail or by phone, but haven't met face to face yet. I am currently shuffling through ideas for my scheduled demonstration of TI-Base for the conference. As the NorthCoast members will tell you, the biggest problem is getting me to shut up. Quite a few NorthCoast members are planning to attend. I think we will all have a great time and make a lot of new friends. Try to make it if you can.

Continued Next Month.

TI-WRITER TIPS #2
- by Bob Saddon -

DUPLICATE KEYBOARD COMMANDS

COMMAND/ESC (also)	CTRL c, FCTN 9 FCTN =
DEL. CHAR	CTRL f, FCTN 1
DEL. LINE	CTRL n, FCTN 3
INSERT LINE	CTRL o, FCTN 8
DOWN ARROW	CTRL x, FCTN x
LEFT ARROW	CTRL s, FCTN s
RIGHT ARROW	CTRL d, FCTN d
UP ARROW	CTRL e, FCTN e
INSERT	CTRL g, FCTN 2
REFORMAT	CTRL r, CTRL 2
NEW PAGE	CTRL p, CTRL 9
NEW PARA	CTRL m, CTRL 8
LAST PARA	CTRL 8, CTRL h
NEXT PARA	CTRL 4, CTRL j
ROLL DOWN	CTRL a, FCTN 4
ROLL UP	CTRL b, FCTN 8
TAB:	CTRL i, FCTN 7
WORD TAB	CTRL w, CTRL 7
OOPS!	CTRL z, CTRL 1

The above chart lists preferred alternatives first, on the left. You have a choice with duplicates and in some cases the choice is very, very clear. In some cases both alternatives are bad; in some, both are okay, or at least, acceptable. Some are situational, i.e., the strokes preceding or following will determine which is better; in other words, in a few special cases the right-hand alternative is better. I offer a couple of rules to help with the selection process:

(1) Anytime you can avoid using two hands you are better off, unless the one-handed alternative is awkward.
(2) If you do have to use two hands, avoid any move that makes use of the top row of keys (keys with numbers).
(3) Paired moves ought to complement each other. By "paired" I mean Roll up/ then down, Last paragraph/ next paragraph, etc.
(4) If no other insight is available, a move may be considered better if its mnemonic.
(5) Some keystrokes precede or follow others in a sequence of moves, moves which tend toward fluidity.

COMMAND/ESC (also)	CTRL c, FCTN 9 FCTN =

First of all, I object to the name selected by TI-Writer. It is confusing to use the term "escape" for anything other than Printer Control Code 27 (the escape command). Nor do I like the term "command". There are Direct commands (such as CTRL 3 for color) and there are Prompted commands which make use of a MASTER PROMPT LINE. This, in my opinion, ought to be called PROMPTS.

This is the only command in TI-Writer which can be achieved in three different ways. The 3 commands do the same thing, whatever you choose to call them, but only if you are currently Editing.

CTRL c is the best choice for every occasion because FCTN 9 is a hard reach and because FCTN = can accidently erase a file: if you have a file on-screen and call up the Disk Directory while in that file you can make a mistake when leaving the disk directory if you are in the habit of calling up PROMPTS with FCTN =.

The accident works like this: when you want to leave the disk directory and go back to your file you must key ENTER. Habit, though, causes you to call up the MASTER PROMPT LINE instead. If you habitually use CTRL c or FCTN 9 to call up the MASTER PROMPT LINE there is no problem, because the program will BEEP (telling you that you made a mistake) and do nothing; now, alerted, you can key the correct response, ENTER.

If, on the other hand, you are in the habit of using FCTN = to call up the MASTER PROMPT LINE you will reap disasterous results. After reviewing the Disk Directory you are supposed to key ENTER to return to your file. If you key FCTN = you will Purge the file and return to the title screen.

The point is: you will never accidently Purge your file if you mistakenly key FCTN 9 or CTRL c instead of ENTER, but you will accidently Purge your file if you mistakenly key FCTN = instead of ENTER. The solution is obvious: never, ever use FCTN = to call up the MASTER PROMPT LINE, because you may accidently Purge your file. Never use FCTN 9 because it is awkward. Always use CTRL c; it is an easy reach.

DEL. CHAR	CTRL f, FCTN 1
DEL. LINE	CTRL n, FCTN 3
INSERT LINE	CTRL o, FCTN 8

We can reference Rule (2) for all three and Rule (1) for both DELETES. INSERT LINE would seem to be a one hander (FCTN 8), but many people are probably in the habit of keying 8 with the right hand; therefore, CTRL o is better. For some reason, TI did not print INSERT LINE on the Quick Reference Card which comes with TI-Writer.

CTRL f is a convenient reach: left thumb and forefinger. Also, after Deleting you can Insert (CTRL g) with a very small finger movement: (f and g are adjacent keys.)

THE ARROW KEYS

DOWN ARROW	CTRL x, FCTN x
LEFT ARROW	CTRL s, FCTN s
RIGHT ARROW	CTRL d, FCTN d
UP ARROW	CTRL e, FCTN e

Since most PROMPT LINE commands are CTRL key commands I strongly recommend CTRL over FCTN for the arrow keys. We also cite Rule (1).

INSERT	CTRL g, FCTN 2
REFORMAT	CTRL r, CTRL 2

There is a special case for using FCTN 2 in the non-Wordwrap mode, discussed several paragraphs below. Most of the time, though, you are in Wordwrap. When Wordwrap is on CTRL g is the better choice. We cite Rule (1) and (not so obviously) we also invoke Rule (5) with some complex logic.

INSERT is dependant not just on the immediately following move (typing in new text) nor even the second move (ARROW UP) but rather on the last move (REFORMAT). Rarely does REFORMAT not follow INSERT. The sequence of moves is usually:

(1) INSERT (2) Type in text
(3) ARROW UP (4) REFORMAT

insert	arrow	reformat
FCTN 2,	CTRL e,	CTRL r
FCTN 2,	CTRL e,	CTRL 2
FCTN 2,	FCTN e,	CTRL r
FCTN 2,	FCTN e,	CTRL 2
* CTRL g,	CTRL e,	CTRL r *
CTRL g,	CTRL e,	CTRL 2
CTRL g,	FCTN e,	CTRL r
CTRL g,	FCTN e,	CTRL 2

Please notice that there is a fluid sequence of moves for only one of the eight possibilities. This single instance uses only CTRL keys. We can cite Rules (1), (2), & (5). I recommend using CTRL g instead of FCTN 2 when breaking text. The deciding factor in selecting CTRL g over FCTN2 is the second move (REFORMAT), not the first move (ARROW UP). CTRL r is faster than CTRL 2.

As promised in the introductory paragraph on INSERT, there is a unique situation which calls for preferred use of FCTN 2 over CTRL g. Let us say you wish to amend a chart with Wordwrap off (CTRL O; hollow cursor) such that you wish to INSERT a caret (SHFT 6) in the same column on every line. All text right of each caret on each line moves right one space. The fastest way to do this is to position the cursor on the first line, on that column. With Wordwrap off, you do a little dance:

(1) FCTN 2 [R thumb, L index]
(2) SHFT 6 [R index, L index]
(3) FCTN x [R thumb, L index]
(4) FCTN s [R thumb, L index]
(5) FCTN 2 [R thumb, L index]
repeat 2-5, 2-5, 2-5, etc.

Keep in mind the device of the Little Dance. It shows up in other forms again and again in Word Processing and is a very useful concept.

NEW PAGE	CTRL p, CTRL 9
NEW PARA	CTRL m, CTRL 0

Rule (2) applies to both. We can cite Rule (4) if you can remember p stands for Page, not Paragraph.

LAST PARA	CTRL 6, CTRL h
NEXT PARA	CTRL 4, CTRL j

Last and Next Paragraph are used in pairs. There are a lot of pros and cons here. Keys h and j are next to each other, making them easy to remember, but requiring you to use two hands. A case FOR using h and j is your ability to think in terms of h being "before" j; h "Heads for Home". While Roll Up/Down (FCTN 6/4) are two handers, CTRL 6/4 are not, even if (properly speaking) 6 is a right hand key which you probably will press with your left hand. 4 and 6 are not adjacent, are on the top row, and most commands are via CTRL key anyway.

I can think of one occasion when it would be better to use CTRL 6 and CTRL 4 in preference to the FCTN key counterparts. When reformatting a whole file to new margins it is best to begin at the top and do a little dance: CTRL r, CTRL 4, CTRL r, etc. For this reason, I break Rule (2) and use CTRL 6/4 in all cases.

ROLL DOWN	CTRL a, FCTN 4
ROLL UP	CTRL b, FCTN 6

Roll Up/Roll Down are used as pairs. Because they are one handers, CTRL a and CTRL b are the best choices. You can poise your hand like a spider: thumb on CTRL, little finger on a, and index finger on b. It is fairly natural to dance your digits back and forth as you read Up and Down. b is an easy mnemonic for "back". You may be tempted to think backwards that A comes "before" B.

The case against FCTN 4 and FCTN 6 is about the same as with LAST/NEXT PARA. The case for 4 and 6 is now bolstered by the fact that Rolling Up/ Down and Paragraphing Up/Down are very similar operations, one pair via CTRL, the other pair via FCTN. Furthermore, the mnemonics for the CTRL key alternates are inconsistent, with b (the second letter) taking you "back" and h (the first letter) taking you "home". With both Paragraphing and Rolling the number logic is backwards: 6 takes you Back; 4 takes you forward. I still favor using 4 and 6 for Paragraphing, a and b for Rolling.

TAB	CTRL i, FCTN 7
BACK TAB	CTRL t
WORD TAB	CTRL w, CTRL 7

Back Tab is not a duplicate command, but I included it because it is used in conjunction with Tab, which is. The best way to use TAB/BACK TAB is CTRL t (left hand forefinger on the t) and CTRL i (right middle finger on the i). We can cite Rules (1) & (2).

OOPS!	CTRL z, CTRL 1

Properly speaking, z is little finger key, but I would imagine most people who use the combination CTRL z put their forefinger on z. CTRL 1 is no better: 1 is also a little finger key but, again, I bet most people use their forefinger here. CTRL z seems a little more fluid.

SUMMARY

When in Wordwrap, use CTRL keys at all times for all keyboard commands which have FCTN key alternatives. Do not use the top row keys at all except for LAST/NEXT PARA (CTRL 6/4) and non-Wordwrap REFORMAT (FCTN 2).

Please share your ideas if you think of good cases to justify the alternative keystrokes I chose not to recommend. The arguments presented here do not cover every case and some good points may be derived from further analysis.

Happy Birthday

This singalong program for the TI-99/4A will play Happy Birthday whilst showing the cake and candles on screen.

You might choose to use it as a birthday surprise and ask those celebrating to blow out the electronic candles on screen.

After a few blows and when everyone is going blue in the face, a press of any key will do the job for you.

```
6 REM  THIS PROGRAM IS FOR THE TI99/4A
7 REM . BY JAMES, D, WRIGHT
8 REM     "HAPPY BIRTHDAY".
10 CALL CLEAR
20 CALL SCREEN(2)
30 CALL CHAR(64,"FFFFFFFFFFFFFFFFFFFF")
40 CALL COLOR(5,5,2)
50 CALL VCHAR(5,16,64,5)
60 CALL VCHAR(5,13,64,5)
70 CALL VCHAR(5,19,64,5)
80 CALL CHAR(112,"18183C3C3C3C7E7E7E7E")
90 CALL COLOR(11,11,2)
100 CALL HCHAR(4,16,112)
110 CALL HCHAR(5,16,113)
120 CALL CHAR(113,"FFFFFF7E7E3C3C1818")
130 CALL HCHAR(4,13,112)
140 CALL HCHAR(5,13,113)
150 CALL HCHAR(4,19,112)
160 CALL HCHAR(5,19,113)
170 CALL CHAR(152,"FFFFFFFFFFFFFFFFFFFF")
180 CALL COLOR(16,16,2)
190 CALL HCHAR(10,11,152,11)
200 FOR I=1 TO 8
210 CALL COLOR(I,9,1)
220 NEXT I
230 CALL HCHAR(18,4,104)
240 CALL HCHAR(11,10,152,13)
250 CALL HCHAR(12,9,152,15)
260 CALL HCHAR(13,9,152,15)
270 CALL HCHAR(14,9,152,15)
280 CALL HCHAR(15,9,152,15)
290 CALL CHAR(40,"FFFFFFFFFFFFFFFFFFFF")
300 CALL HCHAR(16,8,40,17)
310 CALL HCHAR(17,9,40,15)
320 CALL COLOR(2,5,2)
330 CALL CHAR(41,"FE7E3E1E0F070301")
340 CALL HCHAR(17,8,41)
350 CALL HCHAR(16,7,41)
360 CALL CHAR(42,"FFFEFCF8F0E0C080F")
370 CALL HCHAR(17,24,42)
380 CALL HCHAR(16,25,42)
390 CALL CHAR(153,"0103070F1F3F7FFF")
400 CALL HCHAR(10,10,153)
410 CALL HCHAR(11,9,153)
420 CALL CHAR(154,"80C0E0F0F8FCFEFFF")
430 CALL HCHAR(10,22,154)
440 CALL HCHAR(11,23,154)
441 CALL CHAR(56,"8142241818244281")
442 CALL COLOR(4,5,16)
443 CALL HCHAR(18,1,56,224)
450 CALL SOUND(300,262,0)
460 CALL SOUND(300,262,0)
470 CALL SOUND(600,294,0)
480 CALL SOUND(600,262,0)
490 CALL SOUND(600,349,0)
500 CALL SOUND(1200,330,0)
520 CALL SOUND(300,262,0)
530 CALL SOUND(300,262,0)
540 CALL SOUND(600,294,0)
550 CALL SOUND(600,262,0)
560 CALL SOUND(600,392,0)
570 CALL SOUND(1200,349,0)
580 CALL SOUND(300,262,0)
590 CALL SOUND(300,262,0)
600 CALL SOUND(600,262,0)
610 CALL SOUND(600,440,0)
620 CALL SOUND(600,349,0)
630 CALL SOUND(600,330,0)
640 CALL SOUND(600,294,0)
650 CALL SOUND(300,494,0)
660 CALL SOUND(300,494,0)
670 CALL SOUND(600,440,0)
680 CALL SOUND(600,349,0)
690 CALL SOUND(600,392,0)
700 CALL SOUND(1200,349,0)
720 CALL SOUND(300,262,0)
730 CALL SOUND(300,262,0)
740 CALL SOUND(600,294,0)
750 CALL SOUND(600,262,0)
760 CALL SOUND(600,349,0)
770 CALL SOUND(1200,330,0)
790 CALL SOUND(300,262,0)
800 CALL SOUND(300,262,0)
810 CALL SOUND(600,294,0)
820 CALL SOUND(600,262,0)
830 CALL SOUND(600,392,0)
840 CALL SOUND(1200,349,0)
860 CALL SOUND(300,262,0)
870 CALL SOUND(300,262,0)
880 CALL SOUND(600,262,0)
890 CALL SOUND(600,440,0)
900 CALL SOUND(600,349,0)
910 CALL SOUND(600,330,0)
920 CALL SOUND(600,294,0)
930 CALL SOUND(300,494,0)
940 CALL SOUND(300,494,0)
950 CALL SOUND(600,440,0)
960 CALL SOUND(600,349,0)
970 CALL SOUND(600,330,0)
980 CALL SOUND(1200,349,0)
981 CALL KEY(0,K,S)
982 IF S=0 THEN 981
983 IF K=30 THEN 984
984 CALL COLOR(11,2,2)
985 CALL COLOR(11,11,2)
986 CALL COLOR(11,2,2)
987 CALL COLOR(11,11,2)
988 CALL COLOR(11,2,2)
990 FOR DELAY=1 TO 900000000000000
1000 NEXT DELAY
```

DESIGNING CHARACTERS MADE EASY
(OR AT LEAST UNDERSTOOD)

by: PAUL E. SCHEIDEMANTLE

As the title states I have set out to make designing of characters for both fonts and graphics easier to understand the cryptic way in which T.I. explains every aspect of thei computer is best shown in how they explain the designing of characters. They show you a chart similar to FIGURE 1 below and expect you to memorize it or have it in front of you always. Thus making it tedious if not tiresome.

Well lets analyze the chart in FIGURE 1 below. First of all it has been enhansed to include both the numeric values of each dot (pixel) and the decimal equivalent. Now lets look closely and understand why each set not only has a different CODE(HEX CODE).

HEXIDECIMAL is a numbering system that uses base (16) (0 1,2,3,4,5,6,7,8,9,A,B,C,D,E,F). In our case it is short han for those numbers that exceed single digits. But back to th subject. Notice that each colomn has a value above it (8,4, ,1). These and the fact that 10=A will help you design and code your characters much quicker.

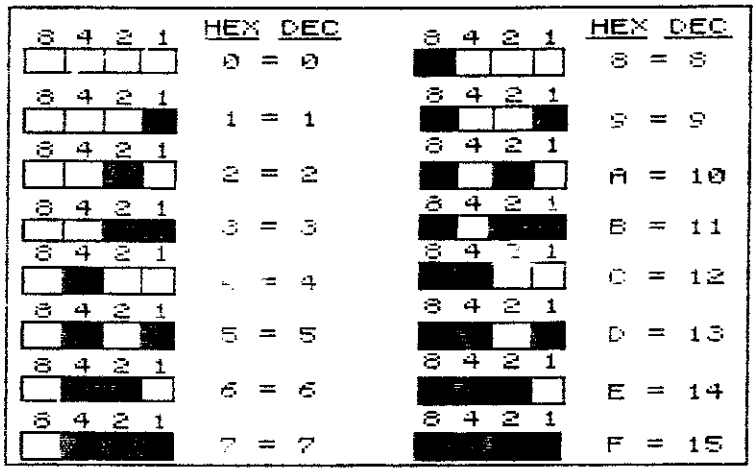


FIGURE 1

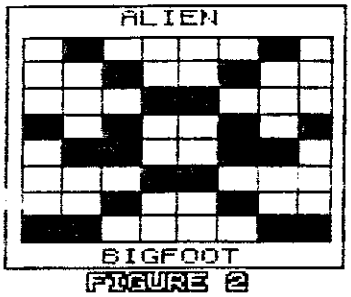


FIGURE 2

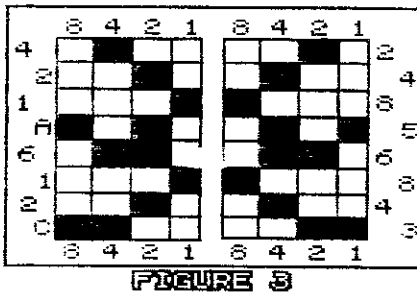


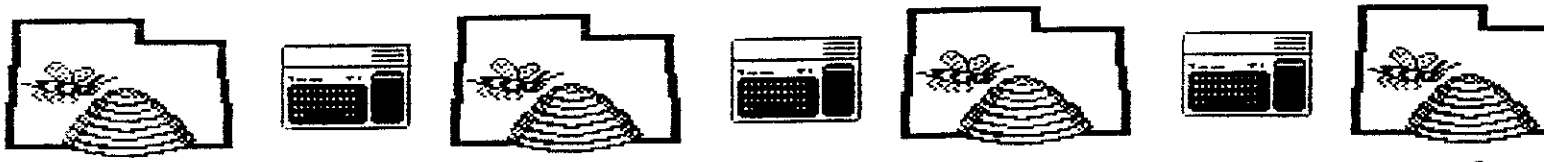
FIGURE 3

As you can see in the chart when all dots are off there i a value of 0 and that when yo turn on the right most dot yo have a value of 1 with the value of each dot doubling as it moves to the left. Notice thatif you have a 3 that not only are the 2 dots on the right tuned on, But more importantly you will now understand why the number is 3; because you simply add them up. After a whle this method will become second nature to you and you will find yourself coding your characters in your head, withot the aid of the chart. Instead of looking up a set lie this "1010" you will auto- maticallly think Oh! thats 8+2=A or "1001" is 9 because 8+1 = Now lets redesign the lower case "a" to a character we will call our ALIEN BIGFOOT. In FIGURE 2 you will see the shape of the character. While in FIGURE 3 it is broken into it's two halves, which are necessary to code it much more efficiently and make it easier to see how it is done. Even though the split is in the midle it is still coded from left to right and top to bottom. Le's take each line separately.

	LEFT SIDE DEC	HEX	RIGHT SIDE DEC	HEX
	8421	= F	8421	= F
LINE #1	0100	=4=4	0010	=2=2
LINE #2	0010	=2=2	0100	=4=4
LINE #3	0001	=1=1	1000	=8=8
LINE #4	1010	=10=A	0101	=5=5
LINE #5	0110	=6=6	0110	=6=6
LINE #6	0001	=1=1	1000	=8=8
LINE #7	0010	=2=2	0100	=4=4
LINE #8	1100	=12=C	0011	=3=3

Now we take the HEX CODE and use it in a program with the CALL CHAR state- ment. CALL CHAR(97,"42241 8A5661824C3")

This page was retyped in PagePro_99. This program is one of the best I have seen. Mel Bress



Ogdens 99'er User Group
Mail Address: 1396 LINCOLN Ave. Apt.B Ogden Utah 84404

GROUP OFFICERS

President: JimBuck 773-2552
Vice President: Richad Morrison 621-3897
Secretary-Treasurer: Richard Scott 776-2551
Librarian: Harold Hilburn 773-0622
Asst. Librarian: Mel Bragg 393-9605
Newsletter Editor: Mel Bragg 393-9605
Associate Editor: Harold Hilburn 773-0622

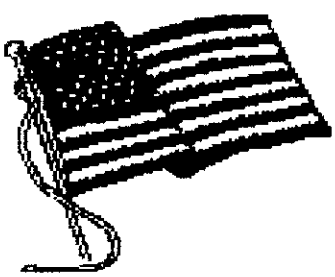
JULY 1989 NEWSLETTER

OUR NEXT MEETINGS ARE:

SATURDAY: JULY 01 TIME: 0900 hrs.
TUESDAY: JULY 18 TIME: 1900 hrs.

We will be meeting in the CIVIL AIR PATROL building at the OGDEN MUNICIPAL AIRPORT AIRPORT ROAD.

TI OGDEN USERS GROUP
1396 LINCOLN AVE. APT.B
OGDEN, UTAH 84404



Have a fun and safe JULY