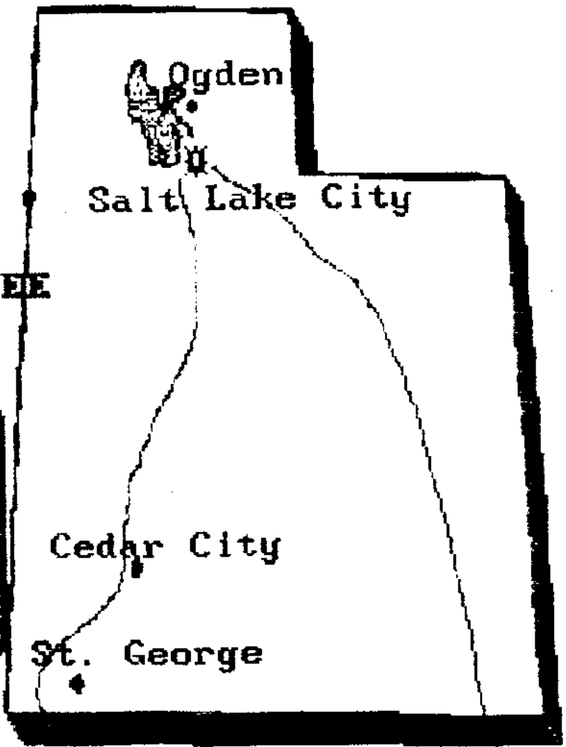
FEST WEST "NORTH" 93

HOSTED BY:
OIIUG- OGDEN II USERS GROUP
AND
SLaVes: SALT LAKE AND VALLEY
USERS GROUP

% FEST WEST "NORTH" 93 COMMITTEE
1396 LINCOLN APT. B
OGDEN, UT. 84404

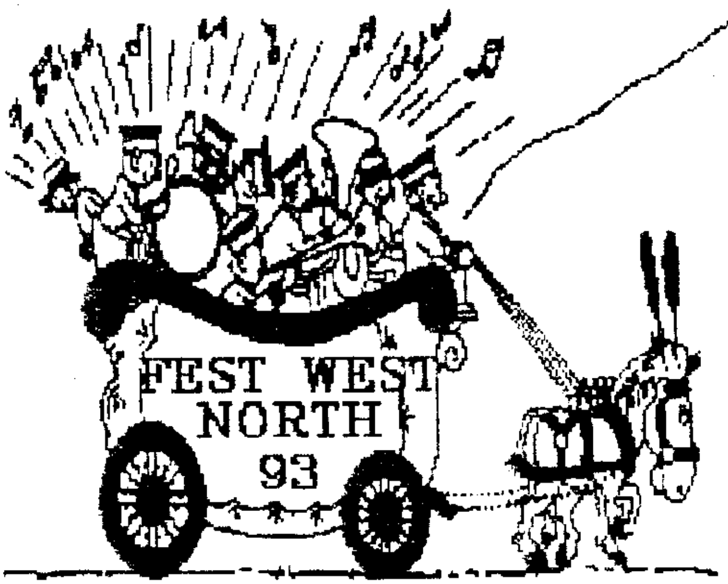


LOCATION: HOWARD JOHNSON HOTEL
122 WEST SOUTH TEMPLE
SALT LAKE CITY, UTAH 84101
(801) 521-0130
TOLL FREE (800) 366-3684
FAX (801) 322-5057

DATE/TIME:
SATURDAY - FEBRUARY 13 1993
9am TO 5pm
SUNDAY - FEBRUARY 14 1993
9am TO 3pm

RATES:
$55.00 for 2 persons
$62.50 for 3 or 4 persons

MORE INFORMATION
WILL BE PUBLISHED
AS WE GET IT..



SALT FLATS BBS
HAS A NEW PHONE #

(801) 394-0064

BAUD 300-1200-2400

24 HRS PER DAY

CALL OUR BBS FOR

FURTHER UPDATES ON

FEST WEST "NORTH" 93

"Life and TImes"

support your local programmer
Copyright © 1992
by Richard P. Phillips
and the SLaVEs & OTIUG UGs

We TIers probably have a higher percentage of hackers in our ranks than can be claimed by the users of any other computer platform[1]. But if this is true, why aren't we seeing more new software being written for our machine?

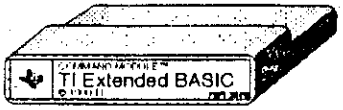
Barry Traver posed this question in the Feb. '92 issue of *Computer Monthly*[2], and his query started me thinking.

When I set out to learn programming, I didn't know anything more about computers than I'd seen in *2001: A Space Odyssey*, and *Star Trek*. At that time I thought there was little that computers *couldn't* do. At the end of my first day with a 99/4A, I came upon a revelation: *computers are as dumb as stumps!* Not simply TI's home computer, but *all* computers. They do *exactly* what you tell them to do. Nothing more - nothing less. This might sound like I'm pointing out the obvious, but for me it was an important truth.

The fact that computers must be told *everything* isn't a problem if you're writing a simple *'Can You Unscramble The Secret Word'* game. But programmers know that we users in the 90's expect more. I wonder if some of them feel that their level of competency isn't up to writing for todays TIer.

I'm not suggesting that we users should lower our standards, but I think it's important to remind software developers that most of us apply different standards to dissimilar software projects. I don't expect the same things out of a low cost shareware *Flash Cards* program for children, as I do out of an expensive *Desktop Publishing Package.*

As was the case for many TI programmers, my first language was TI Basic. I soon followed this up with Extended Basic.



By almost any yardstick TI's Extended Basic is as powerful, as it is slow. It is, however, a good language for writing many of the projects that pop into my head. Additionally, several people have put together packages of Extended Basic callable routines which can perform some tasks much faster than plain vanilla Extended Basic. Some of these routines will even allow you to perform functions that Extended Basic can't manage.

These packages run the gambit from freeware, through shareware. MICROpendium recently ran a series of articles dealing with this very subject, and I feel they're worth looking at.

I've emphasized Extended Basic only because it is a language that many of us understand, and it's accessible to almost everyone. But if your *programming canvas* of choice is another language, by all means do what is right for you.

If you have the talent, the time, and the desire - write! If you have everything but the talent, give it a shot anyway.

I want to leave you with these simple thoughts[3]. When we buy software from a vendor, or on those all to rare occasions send in our shareware fee, in essence we are thanking the developer for his or her efforts. Few freeware programmers ever hear a word about their creation - good or bad. If you like something someone has created, drop the 29¢ and send a letter saying so. If you have suggestions, let them know. Most programmers love to hear from their *audience*. And to you freeware programmers, give yourselves a pat on the back for a job well done. It might be the only one you get...

—Rick

[1] when I say "hacker," I mean someone who "hacks" together programming code. Not someone who breaks into a computer owned by his or her local bank.

[2] Barry writes a monthly column for CM entitled, "TI-99ers In The '90s."

[3] as I am capable of no other...

ASSEMBLY LANGUAGE TUTAOIAL

By Ken Olsen

Edited by Mel Bragg
From the OMAHA TI USER GROUPs NEWSLETTER

I for one am trying to learn assembly. Ken Olsen has a very informative way of understanding how it works. As long as Ken writes an article on assembly. I will be looking for it in there next news letter...
It seems that a lot of people at the meetings want to assembly language. Most, including me, have done some dabbling with it.

I am by no means proficient with this form of programmin This is not going to keep me from trying to give us some ideas on how to do things so we can write programs.

There are a lot of good books and articles on how to do assembly programming. (The TI assembly manual not being one of them.) I recomend that you look at as many as you can to gain insight.

I want you to suggest what you would like to see done in programming, and then put our heads together to see if we can do it.

FIRST THINGS FIRST:

When you start to program there are certain rules you have to follow.

1. Give your program a name so the computer knows where to start.

2. Give REFerences to built-in subroutines.

3. Set aside an area for you to do work in.

OK enough information for one session. Now what everyone has been waiting for. To see a 'program'. There will be explaination with each line for 'clarity'. Those lines that begin with an * are just comment lines that are not acted on by the assembler. To end the program you will have to push the QUIT key. Type in the source code and then assemble it.

You will then have a program that will work as a LOAD AND RUN option from the EDITOR/ASSEMBLER cartridge.

I am getting most of my information from the book "Introduction to Assembly Language for the TI Home Computer" by Ralph Molesworth.

Tell me if it would be worth while to continue or am I boring you. If you want me to continue tell me what you want to learn in assembly.

See Ken's assembly language program next page.

KEN S ASSEMBLY LANGUAGE PROGRAM

```
       DEF  START   This tells the program where to begin
       REF  VMBW    Tell the computer we are going to use the built-in
*                   routine (V)DP (M)ultiple (B)yte (W)rite
DISPLY TEXT 'You have just done it!'  This is what you are going to see on the
*                                     screen. You need the single quote marks
*                                     around what you want to display
SAVRTN DATA >0000   Set aside 2 bytes of memory
WRKAIR BSS  >20     Your work area. You need this so the computer knows where
*                   to put the working registers. >20 (32 bytes) is enough
START  MOV  R11,@SAVRTN  This is the beginning of the program. You are also
*                        (MOV)ing what is in (R)egister 11 and putting into
*                        memory location called SAVRTN
       LWPI WRKAIR  You are (L)oading the (W)orkspace (P)ointer register
*                   (I)mmediately with the address of your work area
       LI   R0,385  (L)oad (I)mmediately (R)egister 0 with the number 385
*                   385 is the screen location to display the text. The screen
*                   locations are from 0 to 767. Register (0)  is where you
*                   have to put this information.
       LI   R1,DISPLY  (L)oad (I)mmediately (R)egister (1)  with the address of
*                      what you want to display. Register 1 has to have this
*                      information
       LI   R2,22   (L)oad (I)mmediatley (R)egister (2) with the length of text
*                   to display. Register 2 needs to know how long the text is
       BLWP @VMBW   (B)ranch and (L)oad (W)orkspace (P)ointer to the (V)DP
*                   (M)ultiple (B)yte (W)rite routine. The VMBW routine takes
*                   the information that is in registers 0,1, and 2 and puts
*                   it on the screen
       MOV  @SAVRTN,R11  Moves what is at SAVRTN and puts it back into
*                        (R)egister 11
       CLR  @>837C  (CL)ea(R) the GPL status byte (837C)
       DECT R11     (DEC)rease by (T)wo the value in (R)egister 11
       RT           (R)e(T)urn to the area that called the program
       END  START   (END) is a directive for the assembler to quit. (START)
*                   tells the computer to run the program START after it has
*                   been loaded. If START is left out here you would have to
*                   type START at the the program name prompt
* Now assemble this program using the R (to label the workspace Registers). C
* (to produce a Compressed object code). and L (to produce a Listing which is
* best put on disk but can be printed as assembly is taking place. There is
* also the S option which will produce a Symbol table which will allow you to
* study what is happening and where things are located
```

Same program without remarks.

```
       DEF  START
       REF  VMBW
DISPLY TEXT 'You have just done it!'
SAVRTN DATA >0000
WRKAIR BSS  >20
START  MOV  R11,@SAVRTN
       LWPI WRKAIR
       LI   R0,385
       LI   R1,DISPLY
       LI   R2,22
       BLWP @VMBW
       MOV  @SAVRTN,R11
       CLR  @>837C
       DECT R11
       RT
       END  START
```

ttc the taylor company
hardware · software

1233 N. Mesa Drive, #2118 . Mesa, AZ 85201 . (602) 464-0354

Dear Computer Enthusiast,

On 15 February 1992, in Phoenix, Arizona a new era in computing will begin. On that day ttc will introduce the "**aTI**", "**xTI**", and "**mTI**" which represent an advanced, expanded and multimedia respectively version of the Texas Instruments 99/4A home computer which was discontinued in October 1983. The 99/4A was chosen because many of its powerful capabilities lay dormant until this event. More importantly, the 99/4A is a concept computer, **ttc** is developing this concept hereforth to be known as "**Concept 99**." Like any new business I must have customers to survive but the dwindling base of 99/4A users may prevent **Concept 99** from developing fully. ***I need your support.***

The majority of Concept 99 software is under the trademark "**t_ware**" and is easily recogniable. I have developed core modules (usable but in no way user friendly) for the following t_ware:

t_draw (a drawing program) **t_chess** (a chess program)
t_font (a bit map font program) **t_sked** (a scheduler program)
t_base (a dbase manager) **t_write** (a word processor)
t_learn (languge tutor programs: *Mid-East, European, Slavic, Asian*)

I can use these programs but to make them marketable, they must be completed, tested and most importantly have manuals written. To accomplish the above tasks I must know where to concentrate my resources. Please let me know for which program you would be willing to make a deposit in order to support continued development. Use the address above to correspond.

Also, if you are aware of any group willing to pay for custom software (especially, educational), I'm available to discuss terms.

Thank You

Chris Taylor

Chris Taylor
owner, **ttc**

The National Committee for TI Standards (NCTIS)

Committee proposal, generated at Fest West 1992, Phoenix Arizona.

To form hardware, software and configuration standards to extend the life of the 99/4a and bring order the the community.

In these hard times, the TI community need a direction to go. In the past other committee's have been formed, such as ANSI to generate standards for hardware and software developers to follow. The standards set forth by NCTIS will aid the users and developers in providing a better software/hardware solution for you. Once standards are set, it is recommended that all current and new software is labeled as standard #1... compliant. These standards should have acronyms for easy recognition.

The following guidelines were discussed at a "Vendors Forum" on February 15, 1992 in Phoenix Arizona. These are recommend standards for the community to ponder upon until May 1992 at the Lima fair, at which time the standards will be decided and publicized.

LEVEL #1: TI 99/4a Console, 32k memory expansion, cassette, and EA/5 loader (EA, Supercart, TI Writer, Multiplan, etc.)

LEVEL #2: Level #1 system PLUS: RS232, and DSSD Disk drive and controller

LEVEL #3: Level #2 system PLUS: at least 128k of CPU RAM, bankable at the >6000 space.

LEVEL #4: Level #3 system PLUS: 9938/58 VDP with 192k VDP RAM

PLEASE remember that these are recommendations generated by this first meeting, and are by no means locked in stone. We are presenting these ideas to you, the user, the developer, the market. Please take our recommendations and think carefully about them, and forward your ideas to your local user group, and then on to the Lima fair.

We appreciate your support.

TI-Base -By- Inscebot, Inc.
P.O.Box 291610, Pt.Orange, FL 32129
Version 3.01 Tutorial 25.1.1 By *Martin A. Smoley*
NorthCoast 99'ers User Group - Dec. 21, 1991

This is a re-hash of the tutorial information I did around April, May and June of 1990, which I added to and updated. If you read my tutorials word for word, some of the information will be familiar. It is a very, very useful part of TIB.

The INSTALL Memory Area or TI-Base Macros

```
filename \MC
MODIFY COMMAND

filename \DS
DISPLAY STRUCTURE

filename \DST
DISPLAY STATUS

filename \IC
INSTALL CATALOG

filename \RES
SET DATDISK=DSK6.
SET PRGDISK=DSK5.
SET PRINTER=PIO.CR.LF
SET PAGE=000
SET HEADING ON
SET TALK ON
SET SPACES=01
SET RECNUM ON
SET LSPACE=256
CLEAR LOCAL
SET CURSOR=02
SET CRLF ON
CLEAR
DISPLAY STATUS
INSTALL CATALOG

filename \DSPA
PRINT (Drft),(E)
DISPLAY STRUCTURE
SNAP
PRINT (Drft),(f)
PRINT ALL

filename \D1
SET DATDISK=DSK1.

filename \D6
SET DATDISK=DSK6.

filename \D7
SET DATDISK=DSK7.

filename \D8
SET DATDISK=DSK8.

filename \D9
SET DATDISK=DSK9.
```

Macro Instructions have got to be one of the big new features in TI-Base. A Macro, or Macro Instruction, is roughly the ability to execute a large command, or a large group of commands, with a single keystroke or a very short key input. TI-Base Version 3.0 or later has that capability. It's a little repetitive to set up a large number of Macros, but once you've done it the rewards are great. Dennis has set up a usable area in VDP RAM, which is handled by the phrase INSTALL, for TIBs use. You should think of the word INSTALL more as the name of the area and not as a command. The things which you can do to the INSTALL area are CLEAR, ADD, REMOVE, CATALOG, LOAD and SAVE. You must create a command file on disk for each Macro Phrase you want to use. For example, I entered MODIFY COMMAND DSK1.\MC. This created the CF named "\MC" on disk drive #1. When the Edit screen appeared I entered two words "MODIFY COMMAND" and I pressed <FCTN 8> to save the CF. I did not enter any comments or place RETURN at the end of the CF. Then, at the dot prompt I entered INSTALL ADD DSK1.\MC. TIB retrieved the CF named "\MC" from DSK1 and placed its contents (MODIFY COMMAND) in the INSTALL area under the name "\MC". This allows me to execute that command by simply typing \MC at the dot prompt. This may not seem like much at first, but here's the big picture. TIB can execute many individual commands from VDP by their names and a Macro can be as large as a Command File. I created each of the Command Files you see on this page under their individual filenames and used the ADD directive to place them all in VDP at the same time. After that I entered INSTALL SAVE DSK6.INST2. TIB SAVEd the complete INSTALL group to DSK6.INST2, with the suffix "/I". Next, I added the line "INSTALL LOAD DSK6.INST2. to my SETUP CF. This tells TIB to automatically LOAD all the commands when TIB is powered up. I haven't tried it yet, but I think that you should be able to stack up your ADD commands in a CF to make it easier to modify the overall INSTALL package. The number and size of Macros placed in VDP are only limited by space, which is currently 2546 Bytes. With everything you see to the left loaded into INSTALL I still have 1879 Bytes left. "Not Bad!" This Macro package means a lot to non-ramdisk users, because the execution is very fast compared to disk access. You could load several large CFs, which you use often, into INSTALL and execute them when needed. I wanted to demonstrate this idea, so I loaded the complete CF named 1LBL91 from Tutorial 24.1.2 (Sept. 14, 1991) into INSTALL. I already had the CF on DSK7 of my RAM Disk. I merely typed INSTALL ADD DSK7.1LBL91 at the Dot prompt (Dp) and pressed ENTER (<E>). This would be a good test because 1LBL91 contained a wide variety of TIB commands, including RETURNs and COMMENT lines. After TIB ADDed 1LBL91 to the INSTALL area I typed \IC <E>. This runs the INSTALL CATALOG Macro you see to the upper left. This told me that the 1LBL91 CF used 1471 Bytes of INSTALL memory space and that I still had 405 Bytes remaining to use. "That's great!" At that moment I had placed twelve Macros in INSTALL, the eleven on the left of this page and 1LBL91. INSTALL contained the twelve Macro names and one hundred and three lines of commands and comments, and I still had 405 Bytes left. Next I typed 1LBL91 at the Dp and <E>. 1LBL91 ran just fine. It opened the Database (Db), set my printer, asked me for the record number, found the name I wanted, printed some labels (using my special printer control commands), reset my printer and TIB commands and RETURNed me to the Dp. "And I think it's a little faster than my RAM Disk. I love it." If you manage this space well, the speed advantages over regular disk drives will be enormous.

Next Page.

TI-Base -By- Inscebot, Inc.
P.O.Box 291610, Pt.Orange, FL 32129

Version 3.01 Tutorial 25.1.2 By *Martin A. Smoley*

NorthCoast 99`ers User Group - Dec. 21, 1991

1LBL91 is a great example of a large CF which can be stored and run from the INSTALL area, but I normally run small CFs as Macros. I always seem to be using the wrong disk for my DATDISK when an idea strikes me for something to do. This led me to the creation of the last five Macros on 25.1.1 (\D1...\D9). "DSK6, 7, 8 and 9 relate to my Bud Mills Horizon RAM Disk." If I want to access a bunch of stuff on drive #7, I type \D7 (E) at the Dp, and TIB processes the command SET DATDISK=DSK7. The key stroke savings are not much for one Macro use, but if you do this ten times during one computer session it means a lot. A Macro that means even more to me is \DSPA. I make a lot of changes to several small Dbs on a frequent schedule. Whenever I do, I like a printout I can use to check my work while I'm away from the computer. I might type \D7 (E), USE CLUB91 (E) and then \DSPA (E). These three short Macros would switch my DATDISK to DSK7, USE CLUB91 located on drive #7 and \DSPA would set my printer to Emphasized mode, DISPLAY the Dbs STRUCTURE to my screen and then SNAP the screen to my printer, next it changes my printer to Condensed and prints the whole Db.

SETUP/Command File

```
  SET TALK OFF
WRITE 22,4,"Welcome to TI-Base Ver.;
3.01"
*                             SETUP/C
*               Ver. 3.01   04/14/90
  INSTALL LOAD DSK5.INST2
*
COLOR WHITE DARK-BLUE
PRINTER EPSON
DO \RES
*
*            Version  3.01
*
*   Type QUIT to terminate TI-Base
*
*   \MC    = Modify Command
*   \DS    = Display Structure
*   \DST   = Display STatus
*   \IC    = Install Catalog
*   \RES   = RESet TIB Options
*   \DSPA  = \DS, SNAP, PRINT ALL
*   \EDIT  = EDIT
*   \MS    = Modify Structure
*   \D1    = SET DATDISK=DSK1.
*   \D6    = SET DATDISK=DSK6.
*   \D7    = SET DATDISK=DSK7.
*   \D8    = SET DATDISK=DSK8.
*   \D9    = SET DATDISK=DSK9.
*
RETURN <FCTN 7> help not available
```

After \DSPA is finished I type CLOSE (E) and go on to the next job. I have included another listing of my SETUP CF in the lower left corner of this page because I want to run through parts of it again. SETUP is the CF that automatically runs right after you type in the current date when TIB is loading. The first important line is INSTALL LOAD DSK5.INST2. This line takes the one file that holds all 11 Macros from 25.1.1 and loads them into INSTALL. As soon as that is done the 11 Macros are ready to use. Next I set the screen colors I like. The third thing I do is load up the printer commands from my personalized Db as I tried to demo in TUT 24.1.1. The last command I issue from SETUP is DO \RES, which runs the RESET Macro that should now be in the INSTALL area. You should take special notice that you must include the [DO] when running a Macro from a CF. If you run a Macro from the Dp the [DO] is not used. The last 20 lines of SETUP are all comments. By my positioning, these lines will remain on the screen after SETUP is finished. This allows me to refresh my memory as to the Macros which I have loaded into INSTALL and to possibly print out a copy of this screen, using SNAP, if need be. This whole job was a lot of work the first time, but now that it's done, and runs itself each time I start TIB, it's a great little tool. I need to throw in another important note. INSTALL works like a stack with the last item you put in being the top of the stack. This means that you cannot remove and/or replace an item in the middle of the stack without extracting and replacing all the items above it in the stack. For example, if I want to remove \D1 from INSTALL, I must enter INSTALL REMOVE \D9 (E), INSTALL REMOVE \D8 (E), INSTALL REMOVE \D7 (E), INSTALL REMOVE \D6 (E) and finally INSTALL REMOVE \D1 (E) to accomplish my goal. You would then need to replace, using ADD, any of the Macros you did not want removed with \D1. For this reason you need to place the most temporary Macros at the end, or closest to the top, of the stack, as I did with 1LBL91. If you need to REMOVE something that is more than half way into the stack, you should consider using INSTALL CLEAR to CLEAR everything out of the INSTALL area and then put back what is needed. There is a little more information on ADDing CFs to the INSTALL area, using those CFs and then REMOVE(ing) them from the INSTALL area, from another CF in the April, May and June 1990 Newsletters, if you're interested. This is something that you must be determined about, plus you must start small and expand the number and size of your Macros slowly. As a matter of fact you could say that about TI-Base in general.

A special note from Marty

I do not anticipate doing any more TI-Base Tutorials. If I find the time I may try and write something, but I do not expect that right now.

Good luck.
Marty.

A SURVEY FOR 1992
FOR THE
OTIUG USERS GROUP
AND
SLaVes USERS GROUP

Complete it and bring it to your USER GROUP meeting.

Name ____ Date__/__/1992
Address ____
(NUMBER/STREET) (CITY) (STATE) (ZIP)
Phone:Home ________________________ Work ________________________
I do/donot authorize the public release of my telephone number to other members
Signature ________________________

Please rank your computer use(s) from 1 to 8 with 1 being most important
Telecommunications ( ) Games/Entertainment ( ) Word Processing ( )
Business ( ) Programming ( ) Education ( ) Home Finance ( )
Other ( ) specify ________________________

A) Please check the items you own and indicate brand where asked:
TI99/4A( ) GENEVE( ) MICRO EXPANSION( ) RAVE EXPANSION( ) PE BOX( )
SPEECH SYNTHESIZER( ) JOYSTICKS( ) MONITOR( ) BRANDS ________________________
____
RS232: TI( ) CORCOMP( ) MYARC( ) OTHER( )
EXTENDED BASIC: TI( ) SUPER EXBASIC( ) SUPER EXBASIC II( )
MECHATRONIC EXBASIC( ) NONE( )
32K MEMORY: TI( ) CORCOMP( ) MYARC( ) OTHER( ) BRAND ________________
RAM DISK: CORCOMP( ) MYARC( ) HORIZON( ) RAMBO( ) SIZE ________________
DISK CONTROLLER: TI( ) CORCOMP( ) MYARC( ) OTHER( ) BRAND ________________
HARD DISK CONTROLLER: YES( ) NO( ) --- HARD DRIVE SIZE? ________________

B) Please check the language package that you own.
EXTENDED BASIC( ) E/A( ) LOGO( ) FORTH( ) C99( ) PASCAL( )
WHICH LANGUAGE ARE YOU MOST FAMILIAR WITH? ________________
COULD YOU TEACH THIS LANGUAGE IN CLASS? YES( ) NO( )
WHICH LANGUAGE WOULD YOU LIKE TO LEARN? ________________

C) Do you have a modem? YES( ) NO( ) Make Model ________________
baud rates: 150( ) 300( ) 900( ) 1200( ) 2400( ) 4800( ) 9600( )
D) What Terminal Emulator do you use?
TEII( ) TELCO( ) FAST-TERM( ) TRIAD( ) OMEGA( ) MASS-TRANSFERE( )
E) What TE programs do you own?
TEII( ) TELCO( ) FAST-TERM( ) TRIAD( ) OMEGA( ) MASS-TRANSFERE( )
F) Do you subscribe to an on-line service? YES( ) NO( )
Which? COMPUSERVE( ) DELPHI( ) SOURCE( ) STAR-LINK( ) OTHER ________________
G) Which "SPECIAL INTEREST GROUP" would you be interested in?
WORD PROCESSING( ) ASSEMBLY( ) FORTH( ) X-BASIC( ) BASIC( ) OTHER( )
H) Which program do you use MOST often? ________________
I) Which program would you like to learn ________________
J) Do you consider yourself a "NEW USER"? YES( ) NO( )
K) Which committees would you be interested in serving on?
LIBRARY( ) MEMBERSHIP( ) PUBLICITY( ) FUND-RAISING( ) PROGRAMS( ) PARTY( )
L) I would like to see a demonstration about ________________
M) I would like to see a program about ________________
N) I would like to present a program about ________________

TI SLAVES AND OGDEN TI USERS GROUPS OFFICERS

	TI SLAVES		OGDEN TI USERS GROUP	
PRESIDENT	JOE MASARONE	966-3694	HAROLD BINGHAM	394-6382
VICE PRES	WARREN YOUNG	278-1052	DAVE DEHEER	394-6815
SEC/TREAS	RENN CRUMP	966-7144	HELEN HILBURN	773-0622
LIBRARIAN	RENN CRUMP	966-7144	ED ISLER	825-9158
ASST.LIB.			TONY LEAVITT	723-3597
NEWSLETTER EDITOR FOR BOTH GROUPS			MEL BRAGG	393-9605

APRIL 1992 NEWSLETTER

TI SLAVES

OUR NEXT MEETING IS APRIL 18 1992 AT 9:00 am WE MEET IN THE DISABLED AMERICAN VETERANS HALL AT 273 E. 800 S. PLEASE BE THERE PROMPTLY.!!

WE COULD USE YOUR HELP WITH FEST WEST "NORTH" 93.

OGDEN TI USERS GROUP

OUR NEXT MEETING IS APRIL 4th AT 9:00 am and MARCH 21

WE MEET AT THE OGDEN MUNICIPAL AIRPORT IN THE FIRST BUILDING JUST EAST OF THE NEW TOWER. HELP!!! US WITH FEST WEST "NORTH" 93

OK! FOLKS. WE ARE HOSTING FEST WEST "NORTH" 93. SO WE REALLY DO NEED HELP FROM ALL USER GROUP MEMBERS, FROM BOTH GROUPS. COME ON OUT TO THE MEETINGS AND GET AN ASSIGNMENT IT'S GOING TO BE A LOT OF FUN AND HARD WORK. BE PART OF FEST WEST "NORTH" 93.

Slaves & Otiug
1396 Lincoln APT B
Ogden, Utah 84404