



HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP  
 USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP  
 GROUP HOOSIER USERS  
**THE HUGgers**  
**HOOSIER USERS GROUP**  
**People Helping People**  
 HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP  
 USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP  
 GROUP HOOSIER USERS  
 HOOSIER USERS GROUP  
 HOOSIER USERS GROUP  
 HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP  
 GROUP HOOSIER USERS

August 1987

THE HUGgers NEWSLETTER

Volume 5, Number 5

**OFFICER'S CORNER**

The August and September HUG meetings have been cancelled. Anyone with questions, please contact J. Steven Sims at (317)-631-7255 after 5:00 pm weekdays.

Effective upon publication of this notice, I am resigning as newsletter editor and vice president of the Hoosier User's Group.

Special thanks to Dr. Ron Albright for his kind and generous coverage of receipt of the HUGger Newsletters since the beginning of the HUG, in the August 1987 issue of "The Computer Shopper". I have taken pride in the editions I have published, as I am sure Pam Sims and Bill Jones have taken pride in the editions they published.

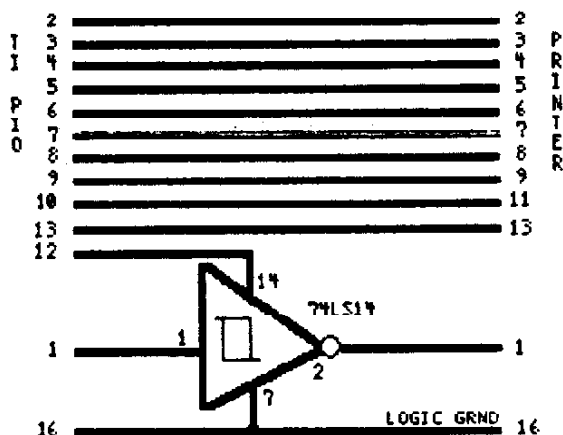
The July HUG newsletter was not mailed out to User Groups on our exchange list. I hope the president will mail the July issue along with this issue to the groups on our exchange list. All HUG funds are exclusively handled by J. Steven Sims. Steve also maintains the HUG mail list, the HUG library, and the HUG bbs (317)-631-9942 at 300 baud. Downloads are for members only, in Caltex #6, dv 163 format. I have sought to get xmodem and other improvements and have been put on indefinite hold forever, it appears.

According to the HUG Constitution, the president may appoint a member to fill an office vacancy. There are two offices vacant now, Vice president and Treasurer are both vacant. I will remain a HUG member until my current membership expires, then I will re-evaluate whether or not to remain an "active" member.

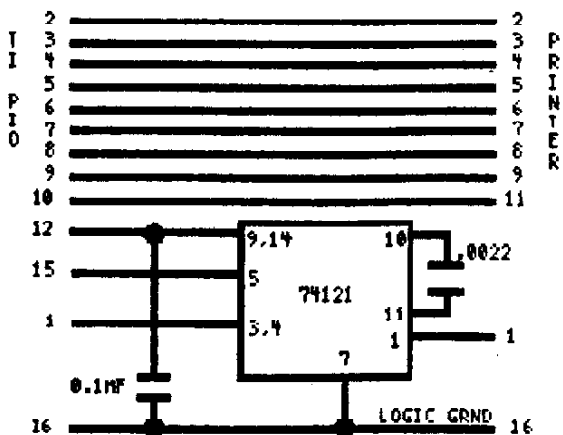
L. L. Conner Enterprises and Electronics has not been at the last three meetings to sell TI and third party products for the TI 99 4/A.

William M. Lucid  
Hoosier User's Group member

TI/Smith-Corona SC-1 Parallel Printer Cable



TI-OKIDATA 82 PARALLEL PRINTER CABLE SCHEMATIC



## STUFF IT AND PRINT IT

By Jim Ellis

No, it's not a turkey! I recently got a printer buffer as commented elsewhere in this issue. I had watched the ad in the 'EVERYTHING FOR THE TI-99/4A' book from TENEX of South Bend, IN. They had the suggested price at \$149.95, well, if you look in Computer Shopper, they start at about \$CALL to \$550 for a 1Meg MacMaster programable buffer. So, \$59.95 for 64K seemed to be to good to be true. But, I ordered it and it worked the very first time. I was concerned because I have an Okidata 82A, which uses a special cable to communicate with the TI. I thought perhaps that I might have to construct another cable. On power up a self-test that the book says takes 8 sec., I timed at 11 sec. If you suspect a problem it contains an extended self-test that can tell you which of the 8 ram chips may be bad. There are no bells or whistles, it has three leds and three switches. The switches are on/off, CLR (clear), and REPEAT. The leds indicate power, CLR, and REPEAT. Regarding timing, I have made up a small chart, all times are approximate.

Length: 152 lines  
Editor PF function: 1 min 18 sec  
Editor PF to Buffer: 4 sec  
Buffer to printer: 1 min 30 sec  
Formater to printer: 1 min 41 sec  
Formater to Buffer: 20 sec  
Buffer to printer: 1 min 28 sec

As you can see, there may not be a major difference in actual print time, but there is a real gain of computer free-up time. All tests were using the same 152 line file, which prints almost 3 pages with a left and right margin of 10. I am using the TI RS-232 card with the PIO port and an Okidata Microline 82A, which prints at the rate of 120 cps. I forgot to mention earlier that the buffer uses parallel connection only. It does NOT support serial. The TENEX ad says units may vary in appearance. Mine looks very much like the ad. It is called the Micro Stuffer and is made by the Supra Corporation of Albany, OR 97321. The unit measures 6 x 3.5 x 1.5 inches. It is just the size to fit under my Radio Shack telephone amplifier. The above

dimensions do NOT include knobs, etc. So, if you are not looking for anything fancy in a printer buffer, but just want something to free up your computer while it does the printing task then this unit is just what you are looking for and it is an excellent bargain.

later.....

## NEWSLETTER CLEANINGS

By Jim Ellis

From Raymond Langevin of the NORTH EASTERN 99'ers via the CIN-DAY USER GROUP of Cincinnati comes the tip to look for a blown fuse INSIDE the power transformer of your 'P' box if you don't have any secondary voltage. It is under the insulation, which will have to be cut. Those power supply boards at Radio Shack have been used by some to stop lock up of their consoles. This info comes from Steve Lisonbee of SLAVE 99ers of Utah via HOCUS of Milwaukee. They advise to only use the ones with certain part numbers, that being 1053201 for the good board. DO NOT use 105314-2, as the 12 v. line jumps from 9.5 to 11.5 volts. The RS part number is #277-1016. You can write to ML Systems, P. O. Box 268, Valley Falls, RI 02864 for information on their KBM 99 Keyboard Interface. It allows you to use any IBM keyboard and comes with 50 preprogramed commands and functions. It sells for \$80 and you can buy keyboards anywhere from \$30 to \$?. If you have a good source, let someone else know. Also, along this same area RAVE 99, 23 Florence St., Bloomfield, CT 06002 has two keyboards to offer for our TI. I am unsure of current price so I will not quote any. The models offered are 99/101 and 99/84, having 101 keys and 84 keys, respectively. A three part project is offered by John Clulow, it consists of a DSR, 8K added to lower memory, and a true clock calender. Bud Mills Services is providing a complete parts kit for the project for \$40. You know him as the supplier for the Horizon Randisk. I received the TENEX catalog recently, and I have been engrossed by an ad for a independent printer buffer. Especially, since I do a lot of printing with TI-Writer. So I broke down and ordered it. It is really....wait, look for the review elsewhere is this issue.

AMNION HELPLINE

A Non-profit Volunteer Effort  
116 Carl Street  
San Francisco, CA 94117

\*  
Helpline hours 9-4 Mon-Sat (Pacific  
Time)

(415) 753-5581

The Free Access Library->A large  
collection of Public Domain software for  
many areas of interest comprising about  
2700 programs.

Single-Sided Volume  
-----

Option 1. \$5.00 incl. disk postage.  
Option 2. \$2.50 + your disk + postage  
Dual-sided Volume  
-----

Option 1. \$8.00 incl. disk postage  
Option 2. \$4.00 your disk postage  
Order Option #2 requires that you send a  
disk AND sufficient return postage in a  
returnable mailer. Since all programs  
are free these charges are only to cover  
the cost of materials, copying, packing,  
mailing and labor. The average cost per  
program comes to about \$0.11. Series #  
Vols General Description  
Id # Avail.

- A 40-42 Entertainment, games, etc.
- B 14-16 Graphics
- C 18-19 Musical, songs, music education
- D 14-17 Learning materials for all  
levels in many different subjects.
- E 8-10 Money, Financial, Business use.
- F 7-8 Technical, higher math, science  
and statistics
- G 7-8 Practical personal and home use.
- H 10-13 Misc.

I 10-11 Informational, unedited text  
files downloaded from various BBSs, etc.  
Also PASCAL and FORTH

Send a note saying the following: I  
certify that all programs ordered are  
not for commercial use or profit and  
that I will maintain the ethics of  
Public Domain. Any fees I charge for  
program copying will not exceed those  
charged me. I will not pass these  
programs on to anyone who will use them  
for sale or personal gain. If I violate  
this agreement in any way, I will pay  
the F.A.L. AND the program's author an  
amount equal to all fees the seller  
receives in excess of those I paid.  
This certification applies to any/all  
programs I received from the Free Access  
Library-Helpline.

Signature \_\_\_\_\_

HOME CONTROL 99 : REVIEW  
BY KEN GLADYSZEWski - NORTHCOAST 99ERS

Home Control 99 by Eagle Software is a disk-based 20K  
program with some assembly language routines (under develop-  
ment for the past four months) used to interface the TI  
with the X-10 POWERHOUSE.

The X-10 POWERHOUSE is an energy management system  
that allows one to control anything in the house which runs  
on electricity. It does this by sending signals over normal  
existing house wiring to remote modules into which lamps,  
etc. are plugged.

the X-10 POWERHOUSE system and Corcomp's Home Sentry  
Interface Cartridge were reviewed in MICROpendium (Dec.86)  
and by OM-NI-TI (Jan.87). The latter was reprinted in this  
newsletter (Feb.87). Please see these articles for a more  
thorough explanation of the X-10 POWERHOUSE system. The  
cartridge software as described in these articles requires  
only the TI console, TV and special cable, but does not  
provide any additional features (files) when used with an  
expanded system (Disk, XBasic, Expanded memory & RS232).

Home Control 99 uses text exclusively to an advantage,  
emulating the capability of X-10's software for the IBM.  
The user types any amount of locations and device descrip-  
tions up to the controller limitation of 256 devices. In  
comparison, the cartridge allows only 14 choices of  
locations and 9 choices of device types for a total of 126.

Using the software, the controller is programmed for  
up to 128 timer events. Each timer event consists of an  
on, off or dim command for up to 16 devices within a single  
housecode (while the module allows for only one device per  
timer event).

The best feature of this program is the ability to  
save collections of timer events to disk as a file. This  
allows one to have a file for vacation, summer, winter,  
etc. These files can be edited, printed and downloaded to  
the controller.

The X-10 POWERHOUSE controller, including required IBM  
RS-232 cable can be purchased for \$19.90 + S&H from  
DAK INDUSTRIES, INC. (Cat. No. 4410)  
8200 Resnet Avenue, Canoga, Park, CA 91304  
1-800-DAK-0800

Home Control 99 is provided on a 5SSD disk with docu-  
mentation on hardcopy, including instructions on how to re-  
connect 2 wires in the IBM RS-232 cable. The program sells  
for \$10.00 and is available from:

(IN Ohio)	(OUTSIDE OHIO)
EDU-COMP	EAGLE SOFTWARE
6516 O'Henry Circle	1269 E. 348th St.
North Ridgeville, OH 44039	Eastlake, OH 44094
216-327-6579	

The August and September  
Hoosier User's Group  
Meetings have been  
  
CANCELLED!!!

## XB:BUG, A Review

=====

by Scott Darling, (C)opyright 1987

XB:BUG is written by J. Peter Hoddie and Distributed by Genial Computerware. XB:BUG is an unusual program. It is like DEBUG for the Editor Assembler. It can be resident in memory and called upon at any time. It allows you to follow a program as it progresses through what YOU programmed. As an experienced XB programmer, I can't tell you how many Hours I spent MANUALLY tracing, deciphering, and endless mapping of a program to see where I went wrong!! We know have such a program to take all the FUN(?) out of the old methods. XB:BUG requires XB version 110 and above, Disk, and J2K. Printer is optional tho almost essential. I also ran XB:BUG on the new Triton Superxb with no problems. XB:BUG will NOT work with Myarc XB II. as the memory locations are totally different.

The only limitation to XB:BUG is not the program but the memory limitations of XB. The program is 5K long. If you attempt to use XB:BUG on hybrid XB/AL files. You will have to remember that XB:BUG loads in Low Memory. There is a >A000 version on the distribution disk. You will be limited to and 12K program in the >A000 space. Like I mentioned, this is not a limitation of XB:BUG, but the 4A's.

To load XB:BUG, use the standard "RUN DSK1.LOAD" format or auto boot from power up. Normally you would load XB:BUG first, If, for some reason, you want to load XB:BUG after your code, there is a version on the distribution disk. After XB:BUG is loaded the READY prompt will return on the screen. Now you can load your XB program in the normal method. To activate XB:BUG you press the Control and Shift keys simultaneously, or you can do a CALL LINK("GOSUB"). Then you will be presented with the main debugger prompt. The following commands are available. Arrays: This allows you to inspect the contents of an string or non string array.

Breakpoints: Setting breakpoints allows you to stop the execution of the program at various points to check for the other functions of XB:BUG. It is NOT necessary to do this in all cases. But, sometimes the program may execute too fast for XB:BUG to literally catch what you want to examine.

Change: This allows you to change the value of any numeric variable. You first have to invoke a V or A commands.

Data: This gives the line number from which the next READ will get its DATA and also shows the next actual DATA item that will be read.

Files: Lists the unit number and device name associated with each open file. The "mode" of the file was opened in is also given. Input, update, append, or output. Any data in the I/O buffer will be displayed.

Graphics: This item gives you information on 3 items. 1) Character definitions. 2) Color Definitions, and 3) Sprite status. You can manipulate all 3 items.

Kill Sound: This turns off the sound chip. You will like this after going back and forth from XB:BUG and XB.

List: Will list the program you are working on. You can set the line numbers you want to list.

Other Variable Spaces: This item is a beauty. It allows you to inspect variables in the main program AND also in Subprograms. This one is complicated to explain so read the manual!

Program: This supplies information about your program. Line number executing, ON ERROR line number, and OPTION BASE. Also, On BREAK, and TRACE if they are active.

Quit: Quits XB:BUG.

Subprograms: Lists all defined subprograms.

Trace: This will trace back all pending GOSUB and SUBPROGRAM returns.

Variables: List variables and functions with their current values. If there is an array, it will list the DIM. This also works on Subprograms. ?: Will list a line of valid keystroke commands for XB:BUG.

Math functions: Allows you to perform simple calculations.

Match Function: Several of the commands in XB:BUG will prompt for a MATCH string. The one thing I will explain about this concerns the manual. It says you can use a wildcard character. Well, the printer made the asterisk so SMALL you say miss this in the manual! I DID!! The quotation marks and the asterisk combined to make a nice inl blob to my old eyes!!

This is all of the commands. The manual documents each command far more than I have here.

In the manual are detailed instructions on how to manipulate some actual code. There are 5 sample files to play with. I would recommend that these are followed through, before attempting to work on a program that you are writing. XB:BUG is NOT a beginners program, it is very powerful and as such has the capability to destroy a program in memory! If this were to happen and you saved the resultant memory to disk.....you may be cussing for a long time! So what GRADE do I give XB:BUG? I have to give it an A in everything except Ease of Use. Why? As I stated above, this program is not for the novice. If they feel they are buying a program that will teach them XB programming or literally do it all for them. They are sadly mistaken.

## XBASHER, A Review

=====

by Scott Darling, (C)opyright 1987

Written by Mike Dodd, Distributed by Genial Computerware,  
Box 183, Grafton, Ma. 01519, \$19.99, All A+'s.

This program is needed by anyone and everyone!! No clarification you say?? EVERYONE has an Extended Basic program! AT LEAST one!! This program will make that one program run faster and reduce its size. GUARANTEED!! Most of us who have been around the Ti World for awhile remember what SMASH is. The BAD part about SMASH is you had to start it at night and HOPE it was done by morning!! You won't have to worry about XBASHER! Xbasher runs out of the Extended Basic environment. There are two versions available. One for TI XB and one for Myarc XB II. No mention was made of the 9640 compatibility. Probably because the 9640 will be so much faster. You can even run XBASHER on combined XB and A/L programs. Complete instructions are given on how to do this!

To run XBASHER requires that you save your Program in Merge format using the following: "OLD DSKn.filename" then "SAVE DSKn.merge, MERGE". Then insert the XBASHER disk in drive and select XB. The disk files will determine which XB you are using and load the correct version of XBASHER. After the program has loaded, you are presented with a title screen. Next is the option screen. Which is: Shorten Variables, Crunch Lines, Remove REM's and !'s, Remove Let's, Change CALL CLEAR to DISPLAY ERASE ALL (this one alone saves you 5 Bytes!), Don't Change CALL SUB routine Digits, and Change Constants. Some of these are obvious as to what is going on.

Shorten Variables will take all your String and Nonstring Variables and shorten them to one then two character variables. There is an immense saving in memory by doing this. Tho, most people like to have a 'name' for variables. If the variable name is less than 3 characters it is no saving in memory. Its when you go over this limit that memory is being eaten away. There is also an option to print the Variable list to an output device.

Next is crunch lines. This was VERY impressive. XBASHER will crunch or combine lines together. So what about the lines that are GOTO'ed you ask?? (Well somebody will ask!!) The A/L in XBASHER keeps track of the logic flow of the program! THIS part makes the program FAR superior to SMASH!! The only bad thing about this function is that the line length of a line number is so long you may not be able to edit the new line!! Considering this is the only drawback, it is a worthwhile option! I have been able to get 8 lines of code to a line number.....so did XBASHER. Next is REMOVE REM's and !'s. Remarks are good for developing a program but are a hindrance when actually running the program. This option will delete them and restructure the resulting deletion of them. Remove LET's. PLEASE I hope everyone by now realizes the LET statement is inconsequential to programming!

Change CALL CLEAR to DISPLAY ERASE ALL. Nothing irks me more in XB programs than to see a "345 CALL CLEAR" then "350 DISPLAY AT(12,1):...". If you use "350 DISPLAY AT(12,1)ERASE ALL:" it does the same thing as CALL CLEAR and saves memory!!

Next is Don't change Sub Digits. What this option does is change the numeric constants to the characters 0, \, [, ], and . This saves 2 bytes per each occurrence of the variable. But, because of the nature of CALL SUB routines this may cost you MORE memory than any savings. Also, note. CALL SUB routines are like a separate YB program within a program. Consequently you can use identical variable names in CALL SUB's as in the program without any type of error received by the Basic Interpreter. Also, CALL SUBS are slower processing than GOSUB's. The only advantage is to CALL SUB's is variable variable passing!! (Are we confused yet??) Lastly is the Change Constants option. Basically what was said in the previous paragraph applies to this option. EXCEPT in this environment, this option will save you memory. Don't ask me why there is a difference. Just believe me!! So much for the option list. Each option has a letter reference. By pressing that Letter toggles each option on and off. Hitting X says you like what you see on the Screen.

Next screen asks for the input file name. The one you saved in MERGE format and checks to see if you remembered the filename correctly. Then asks for an output name. And even provides a suggested name. Next is an output device and name for the variable listing if you selected that option. FINALLY the computer starts doing the work!! The screen will show you the status of the program. A line count, the last line number referenced by a goto, gosub statement will be shown on the screen. Xbasher makes two passes thru a program. First to make lists of variables, line numbers and other info. The second pass will write the new program to disk. How long will it take?? The size of the program involved is the ONLY factor. I ran an 11 sector file thru Xbasher and it took 5 minutes to do the job. The savings were 599 bytes. Next I ran the ultimate EGO test on XBASHER. I wrote a BBS program that is 98 plus sectors long. Almost 23K in bytes. So, I ran XBASHER against it. I felt I was a decent XB programmer and there was no way XBASHER was going to save any bytes in MY program!!

Well after about 30 minutes and by selecting ALL the options. The darn program found 200 bytes somewhere!! I'm still trying to see where it found them!!

To sum it up, Xbasher is the perfect compliant to any XB program. You only need to run it once, and save the resulting code. XBASHER will show you what XB programming is all about! There is a lot of power in that cartridge!!

## TEN STEPS FOR TYPING IN AND RUNNING ASSEMBLY PROGRAMS

By L.R. Livergood, Decatur 99er US, 9/86

Many consider assembly language programming to be more difficult to learn than BASIC; however, you don't have to become an expert assembly programmer in order to take advantage of it. If one just learns how to enter and assemble the code, then he or she can begin to utilize the expanded capabilities that this language has to offer.

If you are familiar with only the BASIC language, then the concept of "compiling", or in the case of assembly language, "assembling" a program may not be clear to you. With either a high-level language such as BASIC or an intermediate-level language like assembly, a program must be translated into machine language before it can be executed by the computer. TI's BASIC uses an interpreter rather than a separate compiler which allows the "writing" and "running" of BASIC code to be done at the same time without the need to first "edit" and then "compile" the program--a two step process in other languages.

In the case of TI 994/A Assembly Language, you must first enter the program with the EDITOR (provided with the E/A module) and then assemble it with the ASSEMBLER. The key to doing this is to familiarize yourself with the following words:

SOURCE PROGRAM  
OBJECT PROGRAM  
LISTING

where the SOURCE PROGRAM is the collection of assembly language statements typed in by you that are translated by the assembler into the OBJECT PROGRAM. Also produced is a LISTING that includes the assembly language statements typed in by you that are translated by the assembler into the OBJECT PROGRAM. Also produced is a LISTING that includes the assembly language statements, the resulting machine code produced by these statements, the machine code memory locations, as well as other information.

With this in mind, it is fairly easy to get assembly programs typed in and running by doing the following:

CREATE A SOURCE PROGRAM WITH THE EDITOR  
ASSEMBLE THE SOURCE PROGRAM INTO AN OBJECT PROGRAM WITH  
THE ASSEMBLER  
RUN THE OBJECT PROGRAM AFTER CORRECTING ANY  
ERRORS APPEARING IN THE LISTING

Both the EDITOR and ASSEMBLER programs can be found on the disk supplied with the E/A module called PART A. Place this diskette in drive #1 before beginning the 10 steps below:

### CREATING THE SOURCE FILE

1) Bring up the EDITOR Selection List menu on the E/A module. This is done by pressing 1 to EDIT from the Editor/Assembler Selection List.

2) Making sure you have the Editor/Assembler diskette in drive #1, press 2 to EDIT. The Computer should automatically load the EDITOR from the system diskette. If it is already present in memory, then the screen is cleared so that you can begin typing in a new program.

3) Enter the program using the editor functions. It operates similar to the TI-WRITER editor which you are probably familiar with. If you are having trouble, then read 2.1.2.1 of the E/A Manual which explains the editing features. It should be easy to see that the listing you

are about to type in is divided roughly into four fields or columns. It is important to make sure that the information is being entered into the appropriate columns. For reference they are described as the LABEL FIELD, INSTRUCTION OPERATION CODE or ASSEMBLER DIRECTIVE FIELD, OPERAND FIELD, and COMMENT FIELD. Note that by placing an asterisk in the first column, the whole line becomes a comment.

The tab positions of the EDITOR default to the beginning of each of these fields. It is important to keep the columns separated. However, it is not necessary for all columns to contain information on each line.

4) After you have entered the program, you must save it. Press FCTN 9 (or escape key) twice to return to the Editor Selection Screen. The press SAVE and answer (Y)es to the VARIABLE 80 FORMAT (Y/N)? question. Next place a formatted diskette in another drive or replace the system diskette if you have only one drive and type in a file name for the program. The saved program is the source file (not runnable as is) and should be coded as such when you type in the file name.

### ASSEMBLING THE SOURCE FILE

5) Next, bring up the Editor/Assembler Selection List (press the escape key if you are in the Editor) and press 2 to ASSEMBLE. You should have the system diskette in the appropriate drive again before answering the LOAD ASSEMBLER (y/n) question. After pressing (Y)es, you will be asked for the source file name which should be the name given above. Next, you will be prompted for the object file name. This will be the location of the file created by the assembler and should be coded as such.

6) Give a file name for the List File which will contain a listing of the errors encountered along with other information such as line numbers, memory locations, machine code and source program statements. You can use a printer name or disk name, but a name is required even if you don't want a listing.

7) The next prompt is for the OPTIONS. They are R for prefixes to be included, L to produce a listing (if you really do want it), S for a symbol table, and C to save the object file in compressed format. If you aren't sure what to use then type in RLSC and see what happens.

8) Next, you should get the message ASSEMBLER EXECUTING at the bottom of the screen and will have to wait for the PRESS ENTER TO CONTINUE message to appear. If you selected a printer for the listing then you should have that in front of you by now. If you chose to send the listing to a disk instead, then you can examine it now by calling up the EDITOR (see above) and loading the listing into it.

9) If you have any errors in your source program, they will appear in the listing. You must go back and correct these by loading the source program into the EDITOR and re-saving with the corrections. In turn, you must now reassemble the source program. Continue this procedure until you get an error-free listing.

10) Finally, go back to the Editor/Assembler Selection List and select 3 for the LOAD AND RUN option. Give the object program file name at the prompt. If everything is as it should be, then the program should be up and running.

Some additional points to remember. Unless your program includes a way to terminate, you will have to shut off the computer to stop the program. Additionally, just because the listing is free of errors, does not mean the program will run error free. There may be logistical errors in addition to syntax errors which the assembler might not pick up.

# QB MONITOR ~ QB-99' NEWSLETTER

**DELETE**

DELETE FILE: TE-II | HELP |  
 PRINT ON | PRINT OFF | HALF/FULL DUP | BUFF ON/OFF | SAV/CLR BUFF | SEND ASCII | CATALOG | CONFIGURE | MODEM XFER | AUTO-DIALER | QUIT

**CONSOLE**

MODEM BAUD: PRNT ON/OFF | MODEM PARITY | MODEM PORT | PRINT PARITY | PRINT PORT | PRINT BAUD | ISCR'N WIDTH | QUIT  
 M=SND FILE | ID=DUMP BUFF | Y=CLEAR LOG | CANCEL | WINDOW RIGHT | X=SET TIMER | TEXT COLOR | SCREEN COLOR | LOG ON/OFF | FREEZE TOGL | QUIT

**C-TERM**

TOGL PRNTR | LOAD BUFFER | INIT NIT LINE | XMT X/O | ON/OFF | RESET BUFFER | DUMP BUFFER | RESTART PROG | QUIT  
 E=TOGL ECHO | ISCR'N COLOR

**TE-11**

SPEAK | OUTPUT | CANCEL | TRANS | WRAP | CASE | PAGE | EXIT | QUIT

**BASIC**

DELETE | INSERT | ERASE | CLEAR | BEGIN | PROC'D | AID | REDO | BACK | QUIT

**TE-WRITER**

DOPS | REFORMAT | SCREEN COLOR | NEXT PARA | DUPE LINE | LAST PARA | WORD TAB | NEW PARA | NEW PAGE | WORD WRAP |  
 DELETE CHAR | INSERT CHAR | DELETE LINE | ROLL DOWN | NEXT WINDOW | ROLL UP | TAB | INSERT LINE | COMMAND/ESC | LINE #'S | QUIT

**MULTIPLAN**

HOME | TAB | NEXT UML CELL | FORM'D CHAR | FORM'D WORD | CHNG WINDOW | REL/ABS REF | CANCEL  
 LOWER RIGHT | BACK CHAR | BACK WORD | HELP | RECALC | BACKSPACE | DEL FORWARD

**SAVE 8700**

DOPS | REFORMAT | SCREEN COLOR | NEXT PARA | DUPE LINE | LAST PARA | WORD TAB | NEW PARA | NEW PAGE | WORD WRAP | LINE #'S  
 LOWER RIGHT | REL/ABS REF | FORWARD CHAR | BACK CHAR | BACK WORD | FORWARD WORD | HELP | RECALCULATE | NEXT WINDOW | NIT UML CELL | RAVE 99/101

**EDIT/EDIT**

DELETE CHAR | INSERT CHAR | DELETE LINE | ROLL UP | NEXT SCREEN | ROLL DOWN | TAB | INSERT LINE | ESCAPE | QUIT  
 DELETE | INSERT | ERASE | CLEAR | BEGIN | PROCEED | AID | REDO | BACK | QUIT

**FORN**

INSERT LINE | FORTH  
 DELETE | INSERT | ERASE | NEXT SCREEN | NEXT WINDOW | LAST SCREEN | DEL/ENDOLIN | COPY LINE | EXIT EDITOR

**DISK**

FCTN E=UP | FCTN I=DOWN | FCTN S=LEFT | FCTN R=RIGHT | DISK  
 HEX DISPLAY | ASCII | DISPL | LEAVE PROGRAM | BACK | SCTR | RBTAT | SUBPRG | FRMR | BCTR | IREWRITE SCTR | RSTRY PROGRAM | QUIT

**ADV/DIAG**

E=PG/BUF UP | X=PG/BUF DNI | ADV DIAGNOS  
 DELETE | INSERT | ERASE | INXT SCTR/TK | EXIT PROGRAM | PREV SCTR/TK | HELP | RETES PADMEN | ESCAPE | SCREEN DUMPI | ASCII/HEX

**GRAPHX**

BLOWER | FASTER | DRAW | ERASE | NO HELP | ZOOM | COLORS | LINES | CIRCLES | COPY | MENU

**CHES**

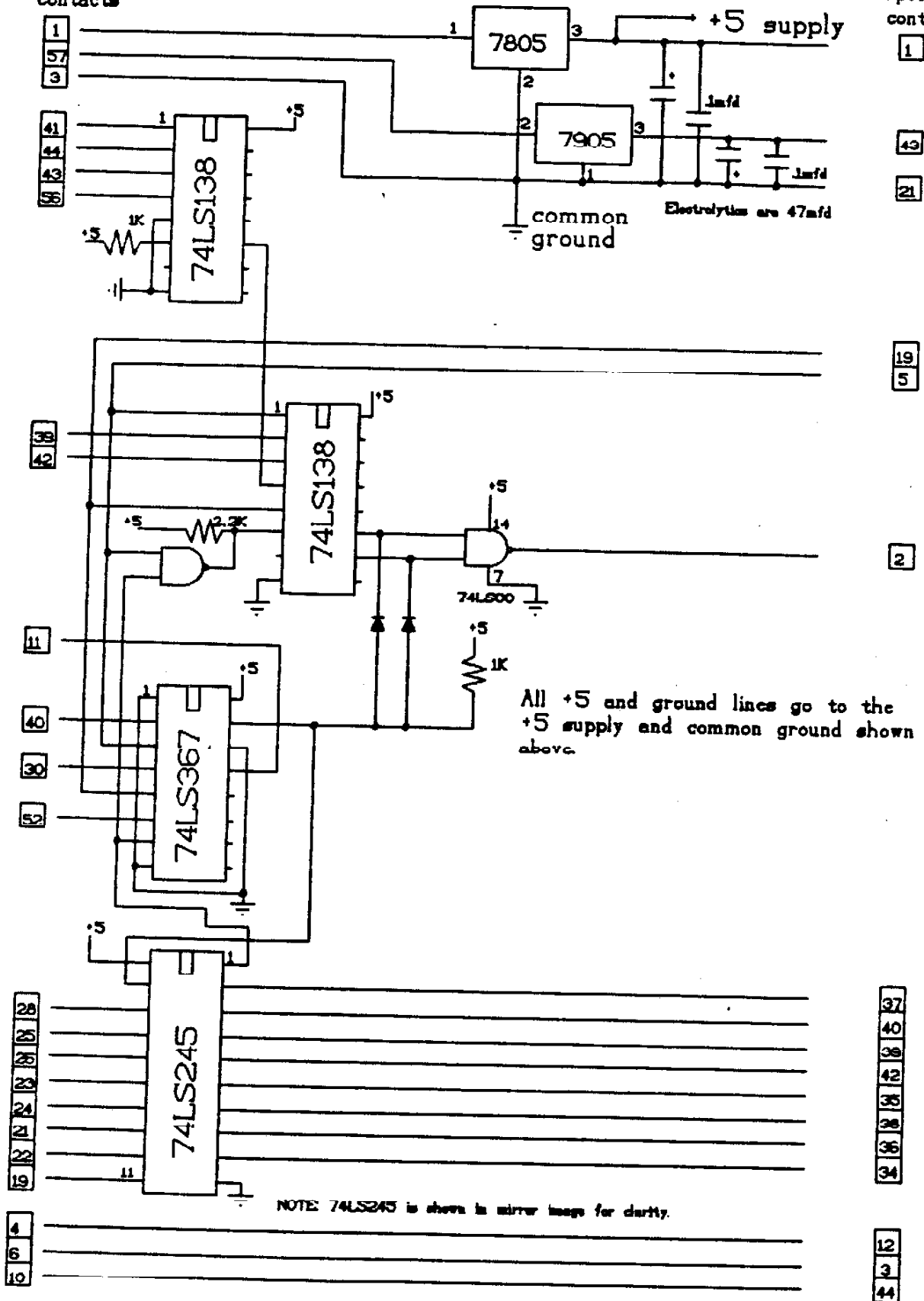
NO CONTROL | KEY PRESS | REQUIRED | MODE CHANG | <=HELP | CHES  
 WITH FUNCTN: D=OFFR DRAW | S=TIME OUT | 4=ERASE | E=RESIGN | P=ARAND POSN | S=SWITCH | R=REPLAY | I=BACK UP | X=SAVE GAME | =QUIT

**CREDITS**

COMPILED BY: ED NACHONIS |  
 PRINTED | USING "PRINTSTRIP" | WRITTEN BY: MIKE NACHONIS | AS PUBLISHED IN MICROPENDIUM: AUGUST 1986

PEB bus contacts

Speech Syn. contacts





# HOOSIER USERS GROUP DIRECTORY

## OFFICERS

President.....Steve Sias 631-7255  
 Vice-President...  
 Secretary.....Dan Eicher 787-4184  
 Treasurer.....  
 Regional Center:  
 South.....Dennis Sherfy 881-5918

## HUGGER NEWSLETTER EDITOR

## MONTHLY MEETING LOCATION

ST. Ann's School  
 2839 S. ...  
 Indi...is, IN

**MEETING CANCELLED**

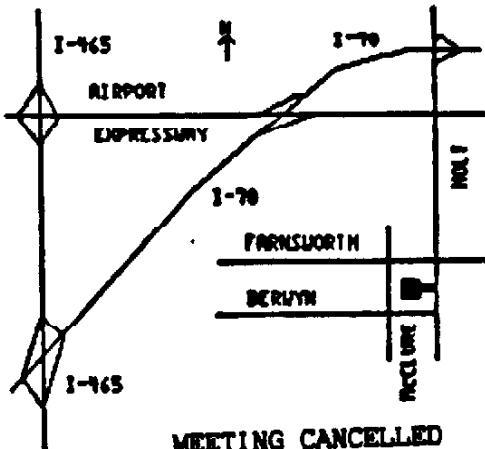
...ings open at 2:00 PM

## NEWSLETTER EXCHANGE

The HOOSIER USERS GROUP is participating in a Newsletter Exchange program with other TI Users Groups. This offer is made with the understanding that, with proper credit, your Users Group can reprint articles from the HOOSIER USERS GROUP Newsletter, and with proper credit, we can reprint articles from other TI Users Groups Newsletters.

## PRINTOUTS

Library listings can be ordered for \$.25 6 X 9 self addressed envelope with \$.66 postage. The HUGbbs Reference Guide can be ordered for \$.50 and a 4 X 9 self addressed envelope with \$.22 postage. Please send orders to our P.O. Box. SORRY, PRINTOUTS WILL BE SENT TO ACTIVE MEMBERS ONLY.



## HUGbbs INFORMATION

317-631-994A 300 baud only  
 The HUGbbs is on-line 24 hours a day.

SPONSOR THE HUGbbs: Any member or retail business can sponsor the HUGbbs. For \$5.00 donation, you get 5 (40 column) lines on the Log-on Title Screen for a week (or for a \$10.00 donation, you get 10 (40 column) lines plus a 24 line by 40 character ad in the Sales option of the file module. To sponsor the HUGbbs, send a check or money order to our P.O. Box (or turn in at our Monthly meeting) specifying how many weeks (and how many lines) you want to sponsor, your name (or company name), address, phone, what you want to say, and the week (and an alternate week) you want the ad to appear.\*

## BACK ISSUES

Back Issues purchased at the monthly meeting are \$1.00 each. Mail order price is \$1.50 per Newsletter (postage included). Orders will be filled within 3 weeks of receipt.

## ADVERTISING POLICIES

There will be no charge for advertisements submitted to the HUGger Newsletter by members (for private sale only). Format for the advertisements is 45 characters wide by 10 lines long. The Ad should be typed or hand printed exactly how it is to appear in the Newsletter. Deadline for an Ad to appear in next month's Newsletter is the 2nd Saturday of the month.\*

For companies who wish to advertise in the HUGger Newsletter, our rates are as follows:

- Pre-Printed Inserts (one page): \$20.00
- One Full Page (one sided) Ad: \$25.00
- One Half Page Ad: \$13.00
- One Quarter Page Ad: \$ 7.00

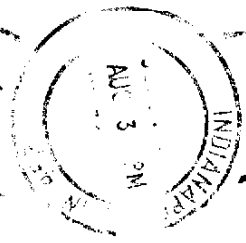
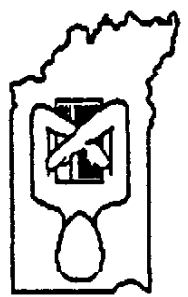
All Ads must be in ready to print condition. Advertisements must be in our P.O. Box before the 2nd Saturday of the month to appear in the following month's Newsletter.\*

\*NOTE: The officers of the HOOSIER USERS GROUP reserve final approval on all advertisements submitted for the HUGger Newsletter and the HUGbbs. The officers and the Newsletter committee are not responsible for typographical errors due to illegible advertisements. All proceeds are accepted as donation to the HOOSIER USERS GROUP.

**MATERIAL  
TIME DATED**

88/03 7880E  
Dan Eicher  
4410 Cardinal Drive  
Indianapolis, IN 46287

HOOSIER USERS GROUP  
P.O. Box 2222  
Indianapolis, IN 46206-2222  
Forwarding and Address  
Correction Requested



### APPLICATION FOR MEMBERSHIP

Below you will find an application for membership to the Hoosier Users Group. Active membership entitles you to the Newsletter, up and download on the HUGbbs, attendance and voting rights at regular club meetings, access to the HUGger Library of Programs, special club activities and special guest speakers for one year. Subscribing members will receive the NEWSLETTER only.

Make check or money order payable to Hoosier Users Group. Send completed application to:

**HOOSIER USERS GROUP**  
P.O. Box 2222  
Indianapolis, IN 46206-2222

(Cut on dotted line)

Check One:

**Active Member**

New: \$20 \_\_\_\_\_  
Renewal: 15 \_\_\_\_\_

**Subscribing Member**

New: \$10 \_\_\_\_\_  
Renewal: 7.50 \_\_\_\_\_

Amount Enclosed: \$ \_\_\_\_\_

---

# \_\_\_\_\_ D \_\_\_\_\_  
S \_\_\_\_\_ O \_\_\_\_\_

Name: \_\_\_\_\_ Today's Date: \_\_\_\_\_

Address: \_\_\_\_\_ Apt. # \_\_\_\_\_

City: \_\_\_\_\_ State: \_\_\_\_\_ Zip: \_\_\_\_\_

Phone: (\_\_\_\_) \_\_\_\_\_ - \_\_\_\_\_

Interests/Comments: \_\_\_\_\_