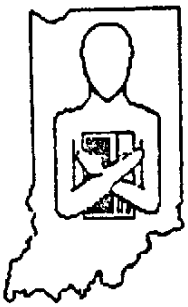


HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP
USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER
GROUP HOOSIER USERS THE HUGgers GROUP HOOSIER USERS
HOOSIER USERS GROUP HOOSIER USERS GROUP
USERS GROUP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

Sept 1990 The HUGgers Newsletter Volume 9, Number 7

OFFICERS CORNER:
By Dan Eicher

This is issue will be small than expected, because of the number of people that did NOT write articles for the newsletter.

The next meeting of the TI Huggers will be Sunday, September 9, 2:00 PM (or 14 hundred hours for you military type) at Saint Anns School.

The next meeting should feature the following:

1. Bud Mills demoing the Memex memory expansion, I am sure he will be glad to answer question on any of his product line. If anyone wants to purchase something from him at the meeting be sure to call him in advance so he will bring it with him.

2. We will have available for playing with an Alpha (comes before beta) version of PRESS complete with docs.

One thing that many Geneve owners do not relize is the fact that their Geneve contains a super DSR in its gate array, this let the designers of the Geneve to use existing pieces of hardware withOUT the need to replace every eprom. Dijit ran into this problem and for maximum compatibility you had to replace the Rs-232, Horizon and P-gram DSR for use in 80-col mode. Up'till now no one has taken advantage of this. A new program that is circulating on the networks is called Setdisk, Setdisk allows you to pop the correct bits in Mdos to set the controller dsr anyway you want (with in hardware limitations).

The most common use of Setdsk is the setting of head step times, another use that will come in handy if you are using a TI controller is the ability to use 80 track drives without the need to physically replace the eprom. In fact Jerry Coffey (author of jumpboot and many other things) has made a special version of his jumpboot just for single density/80 track drives, he warns that if you are going to use 80 trace drives you should either use full height 5.25 or newer 3.5 drives.

Bad News Department:Over the summer break there was an officers meeting, we (your duely elected officers) voted on and passed a rate increase for membership.

The new rate structure is:

New Memberships	------ 20.00
Renewals	------ 17.00

These are the only types of memberships available, the subscribing has been dropped, and we decided against a BBS only subscription.

OFFICERS CORNER:
By Dan Eicher

continued from page 1

I have been with the club since almost the beginning and this is the first increase I can remember. Your membership still represents a terrific value, because in addition to your membership dues we have revenue both earned and earned supplementing your dues, they are: 1. Unofficial corporate sponsors have been helping print the newsletter. & 2. Members of the group have made things to sale to other members and at TI shows, then turning around and donating the profit to the group. This is over and above the several hundred dollars donate by users for upgrading our BBS and all this is over and above the great thanks we OWE John Powell for 1. Purchasing the BBS for our use. & 2. Paying the monthly telephone charges to keep it operating. In short the membership dues WE pay only cover about half of what the group offers.

That was a handy lead in to the last bit of news. I would like to remind everyone to use our BBS. It know operates at 300/1200/2400 baud, full duplex, 8N1 and has over 20 megabyte worth of some of the latest greatest software for the TI & Geneve available anywhere. The software to run this board is a professional & commercial piece of software called TexLink (paid for and donate by Bill Lucid). The Number? (317)-782-994A.

WIRE, by Dan Eicher

On a printed circuit board you will see no wires, but they are there! They are the little lines you see running from place to place. This type of wire is called printed wire. There our two things to keep in mind when choosing wire for an electronics project: 1. How much current will this wire need to carry and 2. Will this wire be moved or will it remain stationary.

Lets answer the last question first (since it is the most simple) if the wire will not be moved ie..like a jumper soldered on a printed circuit board, (PCB) it should be solid wire. If the wire is going to be moved (like the power cords running to you disk drives) then they should be made of multiple strands of wire. This is so if one (or more) of the little wires in the tube (consider the outer covering of the wire the tube and the fine strands of copper wire inside the wire), breaks there will not be a break in the current traveling through the wire from one end to the other.

The size of the wire you use should be proportional to the amount of current that will be moving through it. You can have two possible mismatches. 1. To big of a wire and to little current. this will result in the receiving end getting either a weak signal or none at all (this means the wire itself is causing to much resistance). Case 2. To much current to small of a wire, this is the most dangerous case of all, because it will usually result in a fire.

Cable and wire size is measure in American Wire Gage (AWGS).

1. As the gage numbers increase from 1 to 40, the diameter and circular area decrease. Higher gage numbers indicate thinner wire size.

2. The circular area doubles for every three gage sizes. For example, No. 10 wire has approximately twice the area of No. 13 wire.

3. The Higher the gage number and the thinner the wire, the greater the resistance of the wire for any given length.

A typical example is you would use 30 AWG wire for wire wrapping a protoboard.

AN INTRODUCTION TO SUPER BUGGER.
BY Dan H. Eicher

The thing that turns most people on to assembly is the ability to do things that are not easy to accomplish in any other language. I am going to give you an example of some things that anybody with SUPERBUG II can do without much effort!

SUPERBUG II is a program designed to help assembly language programmers debug their own programs but it can also be an easy and fun way to explore your system. SUPERBUG II is major upgrade over the Debugger TI sent out to all the users groups several years ago.

Before under taking this article you must realize that the way Super Bugger interacts with you is kind of unique, just remember:

When entering an address after a command, do NOT enter a space just type the command ex. m and then the address.

1. Ever wanted to make a light in your p-box go on?

To do this in Super Bugger II is easy! All you have to do is write a 1 to its CRU address.

The command to inspect/change CRU bits is interestingly enough C. Lets turn on the light to our disk controller.

Type,
C 1100,1 <Return>

Your display should say
1100=0000, press 1 then <Return>
Your disk controller light should now be on.
To make the light go off, repeat the process, except in the last step type a 0, then <Return>

This same technique can be used to light up cards that do not have DSR's, a good example of this is the MBP clock card, it has an LED that lights up when certain memory address are accessed. By using the Super Bugger command M (for memory inspect/change) you can make it light up.

Who said assembler is hard,

The next thing we are going to do is search for the message - 'Review Module Library', you may have received this message at one time or another after pulling out or pushing in a cartridge. TI had planned on producing a box that you could put 8 cartridges in, you would have (after selecting review library) saw a screen like this:

1. Extended Basic
2. TI Invaders
3. TE II
.. ect,

You also would have been able to do things like OPEN #1:"SPEECH" from inside extended basic if you had the TE-II module also plugged into the box. The system software to do this is already built into your console.

To do this the first thing SUPERBUG needs is the string to search for, you give it this information by typing,

I
Follow the prompts and enter:
REVIEW <Return>

Then type CTRL-F <at the same time>, then enter g0000,ffff <Return>
This tells the computer to search all the groms for the string
you entered earlier.

On my computer, it found the string at G12AA (the G proceeding
the address means that it is Grom/Gram memory).

To display the string in context you must use the handy m command.
Type M G12AA,12FF <Return> and you should see the Review Module
Library string that is built into our operating system.

For the next trick you will need a Geneve or a Gramulator/GramKracker

Have you ever wondered what would happen if you pulled out the
basic groms that come in your console? Well TI must have wondered
the same thing because in the operating system the wrote a flag
that would handle just such an emergency. Now lets make it happen!

Every menu item that appears on your screen must have a valid grom
header, that means the value AA must appear in the right place
some where in memory, If you have one cartridge loaded that place
should be GE000, all you need to do is change the AA at this address
to a zero and the cartridge disappears.

To do this type M GE000 <RETURN>, TYPE 0000 <RETURN>

If AA does NOT appear at this address you have a slightly different
version of the operating system and will need to repeat the find
string procedure from above, but instead of entering REVIEW, you
would enter the name of the Cartridge you have loaded.

After you find out what address the cartridge description line starts,
usually about 15 bytes in front of that you should find AA, this
of course would need to be zeroed.

In addition if you have loaded the version of Super Bugger that loads
into the Super Cartridge you will need to type
M 6000 <Return> 0000 <Return>, This goes to CPU memory address
hex 6000 and wipes out the grom header here.

When you reset the computer you should get the message:

'Insert Cartridge'

Alot of you have probably heard that the only thing worst than assembler
is machine code. Well today you are going to write your first machine
code program! After all this work with DSR and rewriting CRAM's alot
of the TI's internal grom address's and Status Register are messed up
if you gave the SUPERBUG command to exit 'Q' things would just lock up
on you. So that we may exit gracefully we are going to plug in some
machine code. To do a reset the assembler code is:

BLWP @0000
or in machine code:
0420 > Tells the computer to perform a BLWP
0000 > Tells the computer the address to BLWP to.

The first thing we would do is use SUPERBUG's memory inspect/change
option. so type:

M A000 <Return>

This tells the SUPERBUG two things, 1. We are going to do a memory inspect change and 2. the address we want to change in this case a address in hexadecimal A000 (in assembler, most of the time, if a value is represented in hex it will be preceded by a > sign).

You should type 0420 (this will fill all the bytes at one word address) hit the Space Bar, this will advance you to the next address of memory.

Then type 0000 <Return>, now all that remains is to execute this little Gem and the following command does just that.

E A000 <Return>

Here is something extra for all you 9640 owners, ever wonder how MyARC made the GPL interpreter so compatible? Well thats easy, they used the same code thats in your console! Want your old TI color bar back (temporarily)? Well lets do it:

Type:
M G01AF <Return> <Now go down the line changing values,
should see something like this>
* Remember to continue changing values you MUST use the space bar.
Then enter the following values.
G01AF=02
01B0=FF
01B1=03
01B2=41
01B3=AF
01B4=06 <Return>

Wonder way when you was modifying CPU memory you had to enter 4 digits and now you only need to enter two?
The reason is Grom is organized in a byte fashion and CPU memory is organized in a word fashion which is TWO bytes!

If you haven't turned off your computer you should be able to execute our little hard reset program again. Know you should get your TI color bars again, complete with TI's copyright notice!

SUPERBUG can be purchased from:

Edgar Dohmann
CO RD 149
RT. 5, Box 84
Alvin, Texas 77511
Price was 15 dollars.

The program alone can be copied from the Groups library.

WIRE, by Dan Eicher *continued from page 2*

The matching of a wire and its jobs is called having good impedance, I can tell you from a personal experience how important having proper impedance is, I have hooked to my clone a composite monitor, when I was setting up the system I didn't have a cable handy, so I took one off my stereo, it was a standard RCA male to RCA male, it was thin and cheap, but it worked. I needed to replace this cable so I could get my music back, so I went down to my local Radio Shack. Radio Shack had a new cable that is a 75 ohm coax cable with gold plated RCA connectors at either end. I replaced the cheap thin cable that I was using on my clone with this new cable and the picture quality increased 75 percent.

I am sure that there are better ways to explain the concepts above, but I am a programmer by training and profession and a hardware hacker as a hobby.

Here is a program by Allan Cox of Alabama. This program will print lines on Gemini 10x printer. With a little digging in your printer manual every one should be able to modify this program to work on their set up. If you really want to be fancy you might want to add the commands to create graph paper. This would be handy for those that like to play games like D&D where you must make a map as you play.

Thanks Allan for submitting to the newsletter.

```
100 CALL CLEAR
110 !*************
120 !* RULEDPAGE *
130 !*    BY     *
140 !* ALLAN COX *
150 !*   1990    *
160 !*************
170 !GEMINI 10X PRINTER
180 DISPLAY AT(8,10):"RULED PAGE"
190 DISPLAY AT(12,6):"Gemini 10X Printer"
200 INPUT "Press (ENTER) To Continue.":E$
210 CALL CLEAR
220 B$=RPT$(CHR$(241),80)!LINE
230 C$=CHR$(27)&CHR$(65)&CHR$(24)!LINE FEED
240 D$=CHR$(27)&CHR$(85)&CHR$(1)!UNI-DIRECTIONAL
250 OPEN #1:"PIO"
260 PRINT #1:C$
270 PRINT #1:C$
280 PRINT #1:D$
290 PRINT #1:B$
300 PRINT #1:B$
310 PRINT #1:B$
320 PRINT #1:B$
330 PRINT #1:B$
340 PRINT #1:B$
350 PRINT #1:B$
360 PRINT #1:B$
370 PRINT #1:B$
380 PRINT #1:B$
390 PRINT #1:B$
400 PRINT #1:B$
410 PRINT #1:B$
420 PRINT #1:B$
430 PRINT #1:B$
440 PRINT #1:B$
450 PRINT #1:B$
460 PRINT #1:B$
470 PRINT #1:B$
480 PRINT #1:B$
490 PRINT #1:B$
500 PRINT #1:B$
510 PRINT #1:B$
520 PRINT #1:B$
530 PRINT #1:B$
540 PRINT #1:B$
550 PRINT #1:B$
560 PRINT #1:B$
570 PRINT #1:B$
580 PRINT #1:C$
590 INPUT "WANT ANOTHER PAGE? (Y/N)":YN$
600 CALL CLEAR
610 IF YN$="YES" THEN 260
620 IF YN$="Y" THEN 260
630 IF YN$="N" THEN 640
640 CLOSE #1
650 END
```

Editor Note:
While putting together this article I used a very handy, but unsung command of TI writer the command is the trusty LF command. It can be used to load parts of a file from another existing file, I use it all the time when I am writting assembly. The Load Files command actual syntax is as follows:

```
203 3 116 DSK1.MyFile
```

This will load lines 3 through 116 of Myfile after line 203 of the file you are currently editing. But remember some programs do NOT like lower case character in a file name, this is a hold over from the original 99/4 which did not have a lower case character set. This still does not explain why TI chose to just shrink their upper case character set and call it lower case. One can only assume TI did hundreds of hours of research and found that the font they where using was optimal for the primitive display technology available to most home users at the time. :)

How to program 25 series eproms when your programer only does 27xx.
By Dan Eicher

One problem I have come upon many times is the need to duplicate or modify 25 series eproms. These are used in TI disk controllers, TI rs-232 and are used in the console to store part of the operating system code. Only TI used the 25 series eproms - the rest of the world including Corcomp and Myarc all use 27 series eproms in their equipment. One unfortunate result of all this is that the commonly available eprom programmers are only designed to "burn" 27 series eproms. This leaves the TI hardware hacker in a world of hurt unless he or she has: 1. The Mechratronic eprom programmer or 2. Builds the daugther board described in this article.

I present to you in this figure the pin configuration of both the 27 and 25 series eproms:

A7=	o V	=Vcc	A7=	o V	=Vcc	A7=	o V	=Vcc	A7=	o V	=Vcc
A6=		=A8	A6=		=A8	A6=		=A8	A6=		=A8
A5=		=A9	A5=		=A9	A5=		=A9	A5=		=A9
A4=		=Vpp	A4=		=A11	A4=		=Vpp	A4=		=Vpp
A3=	2	=!G	A3=	2	=!G/Vpp	A3=	2	=CS	A3=	2	=PGRM
A2=	7	=A10	A2=	7	=A10	A2=	5	=A10	A2=	5	=A1
A1=	1	=!E	A1=	3	=!E	A1=	1	=PGRM	A1=	3	=A1
A0=	6	=Q8	A0=	2	=Q8	A0=	6	=Q8	A0=	2	=Q8
Q1=		=Q7	Q1=		=Q7	Q1=		=Q7	Q1=		=Q7
Q2=		=Q6	Q2=		=Q6	Q2=		=Q6	Q2=		=Q6
Q3=		=Q5	Q3=		=Q5	Q3=		=Q5	Q3=		=Q5
GND=		=Q4	GND=		=Q4	GND=		=Q4	GND=		=Q4

If you build a daughter board you can tell your eprom programmer that you are going to program a 2732 and then place your 2532 in the "carrier". Then put the daughter board containing your 2532 in the programmer, select the voltage level at 25 volts and then burn away.

You will need the following parts.

1. A 24 pin ZIF <Zero insertion force socket>
2. A 24 pin wirewrap socket. RS# 276-1983
3. A wire wrap protoboard. RS# 276-148
4. Electrical tape or heat shrink tubing.

=	1 V	24	=
=	2	23	=
=	3	22	=
=	4	21	=
=	5	20	=
=	6	19	=
=	7	18	=
=	8	17	=
=	9	16	=
=	10	15	=
=	11	14	=
=	12	13	=

Take the wire wrap socket move pin 18 to 21
pin 19 to 19
move pin 20 to 18
move pin 21 to 20

The right side should look something like this.

TI states in their 9900 Family Systems Design book - 2532 EPROM

* 4096x8 Organization
* Single + 5v Power Supply
* Pin Compatible with all 8K & 16K EPROMS
* Plug in Compatible with the TMS 4723 32K ROM
* Static Operation* Interchangeable with Intel 2716

This project was devised by Tom Spillane of Dijit for his own use, and neither the author take any responsibility for damages arising from this project.

#

A little BIT of math
by Dan Eicher

To really understand how computers work the first key element that must be understood is boolean algebra. An easier way of thinking of this is as TRUTH tables.

And Table	Or Table	Exclusive or(XOR)	Not Or(NOR)	Not And(NAND)
1 and 1 = 1	1 or 1 = 1	0 XOR 0 = 0	0 NOR 0 = 1	0 NAND 0 = 1
1 and 0 = 0	0 or 1 = 1	1 XOR 0 = 1	1 NOR 0 = 0	1 NAND 0 = 1
0 and 1 = 0	1 or 0 = 1	0 XOR 1 = 1	0 NOR 1 = 0	0 NAND 1 = 1
0 and 0 = 0	0 or 0 = 0	1 XOR 1 = 0	1 NOR 1 = 0	1 NAND 1 = 0

Examples

00111010	00111010	00111010	00111010	00111010
01011100	01011100	01011100	01011100	01011100
--------	--------	--------	--------	--------
00011000	01111110	01100110	10000000	11100111

For the sake of terminology lets say a 1 is true or positive and 0 is negative or false. This is how the chips in our computers determine what is going on ... a negative voltage level means a negative bit and a positive voltage level means a positive bit.

Remember 8 bits make a byte, 2 bytes make a word. The ability of a single assembly instruction to operate on 16 bits (a word) at a time is what makes our machine a true 16 bit computer. This separates it from earlier technology like the 6502 or the Z-80, these cpu's move data around the computer 8 bits (a byte) at a time.

The last two logical operators Nor and Nand are used far more commonly in electrical engineering than in programming.

At this point you are probable wondering "Ya.. So whats the big deal, how is this going to help me write better programs?"

Well one of the major uses for these types of instructions are in setting "flags". A flag is an indicator to your program of a condition that can be true or false. A possible uses would be to save the configuration of the machine your program is running on in a single byte instead of wasting memory.

In one byte you could hold all the following information:

```
Bit#     What it could mean.
1 If = 1 then color monitor, else adjust colors for black and white.
2 If = 1 then disk system, else assume cassette and do not try to save
         high score.
3 If = 1 then system has a printer, give option to output to printer
         else prepare all output for screen only.
```

Here is were you would need to use your logical operaters to get at these bits, TI's extended basic is rich in these types of commands, for a detailed look at their use buy the program Night Mission by MG.

I think you get the idea of how the use of flags can save an enormous amount of memory when used properly. This is especially important in a machine like ours. In a day when most computers are counting memory by the Megabyte (thats thousands of Kilobytes which is, inturn thousands of bytes).

1 Meg = 1000K = 10,000 Bytes

One last bit of arcane knowledge that you should at least have a passing knowledge of is how to get the "Twos Compliment" of a number.

Basically the twos compliment of a number is how a computer distinguishes

between a positive and a negative number, the computer must do this internally before subtracting two numbers, actually computers can't even subtract, they have to add a positive and a negative number to get the result!

Here is how it is done.

Lets say you have the number 53 thats 00110101 in binary,
and you want to subtract 26 thats 00011010 in binary.

First the computer must compliment 26 and heres how its done,

first you take the number=============>	00011010 (26)
change all 0 to 1 and all 1 to 0======>	11100101
add 1 for good measure================>	1
The complimented number is ---------->	11100110
Take 53==============================>	00110101
ADD 26's compliment==================>	11100110
The answer--------------------------->	100011011

Since this is BYTE arithmetic the 9th digit goes in the bit buck leaving only 00011011 thats 27 decimal which is the result if you subtract 26 from 53.

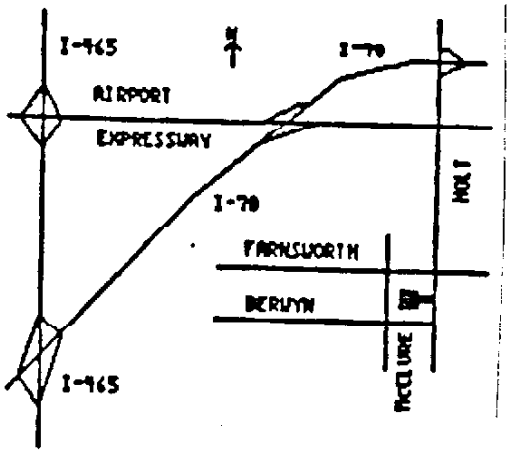
DISCLAIMER

This newsletter is brought to you by the efforts of the officers and members of the HOOSIER USERS GROUP.

THE OPPINIONS EXPRESSED HEREIN ARE THE AUTHORS AND DO NOT NECESSARILY REFLECT THE THE VIEWS OF THE PUBLISHER.

OFFICERS

PRESIDENT..DAN EICHER	241-994A
VICE-PRES..DELBERT WRIGHT	895-1765
SECRETARY..DARLA WRIGHT	895-1765
TREASURER..GARY McQUADE	888-5654
LIBRARIAN..BRYANT PEDIGO	255-7381



MONTHLY MEETING LOCATION
LITTLE HOUSE NEXT TO THE
ST.ANN'S SCHOOL
2839 S. McCLURE
INDIANAPOLIS,IN
MEETINGS OPEN AT
2:00 PM
APRIL 8 1989

BBS
Hoosier Users Group
Baud rate 300,1200 & 2400
On Line 24 Hours Daily
782-994A

Now with a Hard Drive
on an experimental basis
courtesy of Gary McQuade

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

Forwarding and Address
Correction Requested

TIME DATED
SEPTEMBER 9,1990
MATERIAL

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 ou lie 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.

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: $17

Amount Enclosed: $ _____

_____ D _____
$ _____

Name: _____ Today's Date: _____

Address: _____ Apt. # _____

City: _____ State: _____ Zip: _____

Phone: (_____) _____ - _____

Interests/Comments: _____