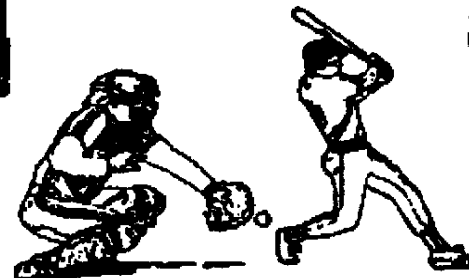


NEWJUG 99ER'S NEWS

OCTOBER 1992



WORLD SERIOUS

QIB70G ONLINE 0 QIB00

Highlights:

GENPROG Re-released	02	Comments & Suggestions	15
VAPP-->Page Pro Part2	05		
AMS 128K MEMORY	08		
99/4a & 9648 Vendors	10		
Joke of the Month	12		
Newsletter Exchange	12		

The opinions expressed in this newsletter are not necessarily those of the officers or members. Articles printed in this newsletter may be copied or reproduced with the proper credit to the author and the newsletter.



NEWJUG 99ER'S UG  
P.O. BOX 1463  
SAVREVILLE, N.J.  
08871-1463

FROM:  
NEWJUG 99ER'S UG  
P.O. BOX 1463  
SAVREVILLE, N.J.  
08871-1463

POSTAGE  
HERE

TO:

AFFIX MAILING LABEL  
HERE



# GENPROG



## RE-RELEASE ANNOUNCEMENT

### PROGRAM DEVELOPERS DEVELOPMENT PACKAGE

BY PAUL CHARLTON

This package contains:

- 1) GENASM macro assembler. Runs in MDOS mode. Ideal for developing code for 99/4a and MDOS applications. Great for C99 programmers!
- 2) GENLINK linker. Runs in MDOS mode. Full support for code libraries. Ideal for developing 99/4a and MDOS applications.
- 3) GENLIB librarian. Runs in MDOS mode. Allows easy interactive maintenance of code libraries for the linker.
- 4) Libraries for 99/4a and MDOS programmers, with commonly used routines.
- 5) GENMAKE make utility. Runs in MDOS mode. Allows an author to easily keep track of file dependencies in programs and large documents. Great for programmers and people who are writing the great american novel. When run, it will update all files which need to be updated, and will (optionally) ignore all files which are already up to date.
- 6) MDOS programming environment documentation, with programming examples for all MDOS library functions, programming tricks for MDOS.

(Video/Math/Keyboard/Taskheader information will be mailed on MDOS Completion)

All utilities can be invoked from a batch file, for complete development automation.

The suggested retail price is \$75.00 for the entire package, which will be available from 9640 NEWS which includes shipping and the subsequent follow-up documentation.

Quick notes on GENASM assembler.  
(c) Copr. 1989 Paul Charlton

This is a one-pass assembler which generates object code for all 9995 opcodes (and it is FAST).

Symbol table size is only limited by the amount of memory on your system. (over 2,000 fully cross-referenced symbols with TIMODE off) Assembler performance:

	GENASM	99 assembler (GPL speed 5)	speedup
floppy(1)	2,500 line/min	900 line/min	x3
HFDC(2)	7,000 line/min	1900 line/min	x4
harddisk	12,500 line/min	2600 line/min	x5

- 1 --- interlace 4 floppy disk
- 2 --- interlace 22 HFDC disk

In addition to the standard 9900 series assembler directives, the assembler supports nested conditional assembly, nested macros, and local symbols.

The assembler can generate several types of reports:

- 1) Error log
- 2) Unreferenced symbols (unused code)
- 3) Cross reference of programmer specified global symbols.
- 4) Full cross reference of all global symbols (w/c registers)
- 5) Full cross reference of all global symbols (including registers)

Error reports are very complete, identifying the file, line number and statement number of the error, as well as a highly detailed description of the cause of the error (good learning aid for people just starting to dabble in assembly.)

Expression evaluation is much more robust than the TI99 assembler, with more numeric operations allowed (+, -, \*, /, mod, and, or, shift left). Expressions are also allowed to contain parentheses to override the normal left-to-right evaluation sequence of expressions.

All symbols are allowed to contain up to 31 characters (including REF/DEF !!!) Get rid of those cryptic label names!

Numeric constants can be expressed in Hex, Decimal, -and- Binary.

The assembler also has support for a symbolic debugger for object code.

Quick notes on the GENLINK linker program.

The linker has full support for developers of MDOS and TI99/4a applications programs, including GPL based programs.

The linker accepts command input from the keyboard or a script file.

The linker has full support for a symbolic debugger, and

should be robust enough to support symbolic debugging environments for higher level languages such as Fortran, "C", and "PFSCAL".

You are able to specify blocks to load, with priorities for each block. You are able to move data from place to place in the object code before you save the program image. (Good for reducing the disk space used by a program which is fragmented in memory.)

You are able to save chained program images for MDOS, TI99, and GPL program images. You can also save large program images similar to SYSTEM/SYS (for complete control of the computer!) You are able to use any DEF symbols in expressions for saving and moving blocks of memory.

You can resolve references from subroutine libraries provided, and libraries of own creation.

You can execute any MDOS internal command from within the linker.

The linker is also -fast-, it is able to link more than 150K bytes/min.

#### Quick notes on the GENLIB librarian.

The librarian has full interactive support for maintenance of linker subroutine libraries, including insertion and deletion of subroutines.

This comes with libraries for the full E/A programming environment, as well as a library for MDOS programmers.

#### Quick notes on the GENMAKE utility.

The best way to keep your multi-part programs and documents up to date after making changes to parts of them. In general you only want to update the resulting files which are affected by the changes. GENMAKE does this for you automatically. (You can override this default to update any section you want to.)

GENMAKE files can contain:

- 1) macros to be used on later lines in the file.
- 2) rules specifying file dependencies (can be many lines long)
- 3) actions to be executed if a rule fails to be up to date.

The dependencies can be nested and be shared between different rules.

#### Quick notes on MDOS documentation.

The most complete documentation available for the MDOS programming environment. All parameters, quirks and calling sequences are documented for each MDOS library routine. The structure of a TASK's header is explained. Programming tips and tricks for MDOS programmers.

Programming examples are given for each subroutine call.

AVAILABLE EXCLUSIVELY FROM:

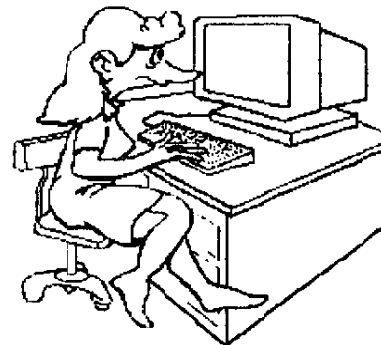
9640 NEWS  
Beery Miller  
P.O. Box 752465  
Memphis, TN 38175

\$75.00 (includes shipping)

RELEASE DATE - NOW

In an effort to help those folks that had their checks cashed (front and back), but order undelivered from JP Software, you must enclose a copy of your cancelled check and \$15.00 (to cover cost of shipping and manual) and you will receive the latest version.

Also, in an effort to help those that already purchased GenFROG and have the uncompleted documentation, arrangements will be made at the appropriate time when the documentation is complete to resolve this matter. Nobody will be forgotten.



## YAPP TO PAGE PRO PART 2

BY DAN GAZSV

Last month, we covered the two different formats (G6 and G7) of YAPP, but we still haven't discussed dithering. In this month's installment we will cover some of the terminology with which I closed last month's article.

Before we can dither, we need a method to represent a color image as 256 shades of grey before dithering. We need a method to convert the color values to some number between 0 and 255 irregardless of the graphics mode of the picture. The following formula can be applied to any color image to accomplish this task.

$$\text{grey} = (.30 * \text{red}) + (.59 * \text{green}) + (.11 * \text{blue})$$

If you were to just plug your intensity values into this formula, the largest grey value you would ever come up with is 82. The formula above assumes that each gun value could

have a maximum value of 256. If you divide 256 by the maximum number of intensities (8) per gun, you would come up with 32. Before you could use the formula, you would have to multiply each gun value by 32. You quickly see that we are doing an awful lot of multiplication to arrive at a number between 0 and 256. Instead of doing that, it would be simpler and quicker to just isolate each gun value as an integer and use it as an index into a gun table. Adding the three indexed values would give you a fair approximation of the above formula.

Intensity	Red	Green	Blue
0	10	19	4
1	19	38	7
2	28	57	11
3	37	76	14
4	46	94	19
5	55	113	21
6	64	132	25
7	73	151	28

If you were curious as to the accuracy of this formula, simply look at the palette colors in a G6 mode VAPP picture. You'll quickly see that the color black has RGB values of zero, while white has RGB values of 7. Keep in mind, these are approximations, so some colors may be slightly misrepresented.

Before any dithering of the image can take place, you need to establish a threshold. In the words of Steve Rimmer, "...This simply means to assign an arbitrary level below which a pixel will be regarded as being black and above which it will be considered white." For formats which use palettes, you use the palette colors to establish values for black and white. Keep in mind that the darkest color (black) would approach 0 and the lightest color (white) would approach 255. While a palette represents all the available colors for the picture, it often turns out that some are not used. In the case of G7 mode where you have up to 256 colors, a lot of the colors are not used. If you were considering pre-scanning the source (something I tried) for darkest and lightest colors used in the picture to come up with a more accurate value for the darkest and lightest color, FORGET IT! For the length of time required to read through the source file, more often than not you'll end up with color values of 0 and 255. In addition, you spend a lot of time doing it. In case you weren't certain of how to calculate the threshold, the formula is:  $(\text{darkest color} + \text{lightest color})/2$

With a threshold value established, it's now time to build a source image. In the IBM world where memory abounds, dither programs tend to read the entire file before they dither the image. For us TI owners, this luxury does not exist so we must work around this. The simplest work-around is to read only enough of the original file to dither and create an output image. For most dither patterns, reading and building 14 scan lines of source image will be adequate. Using the specifications I listed in installment 1 of this article, you read the picture data, get the color value, convert the color value to a

greyscale equivalent and store it in an area set aside for the source image.

Before we take this installment any further, we need to discuss artifacts, filters, error diffusion and fixed dither patterns. The purpose of setting up the threshold was to establish a point of reference as to which pixels in the printed image are on and which are off. If we were to print the image based on just that principle alone, the output image would lose a lot of detail. At best the image would look like a black/white negative. To counteract this problem, fixed pattern dithers were applied to the source image. One in particular was called a Bayer fixed dither. The pattern was an eight by eight matrix and it was compared to an eight by eight block of the image. Pixels that were greater than the corresponding elements in the matrix were white pixels in the destination image. Those that were less ended up as black pixels. Many of the initial MacPaint files were created with just such a dither. As you might expect, this dither pattern tended to introduce patterns into images that really weren't in the initial image. These unwanted visual byproducts in the image are known as artifacts. In an effort to get rid of these unwanted artifacts, a method called error diffusion was used in the dithering process. The best description of error diffusion that I've seen comes from Steve Rimmer in a book entitled "Bit Mapped Graphics".

"...You have to consider the situation from the point of view of a pixel to really understand error diffusion. Let's allow that there's a grey scale pixel about to be turned to a monochrome pixel by comparing it against a threshold. The threshold is halfway between black and white, which will be 127. The pixel in question has a grey level of 150, so it will be white. However there's an error of 23 here - the difference between the real grey level of the pixel and the level of the threshold.

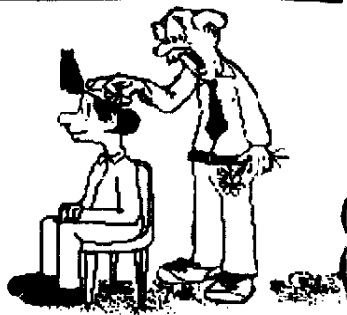
If the error is diffused out to the surrounding pixels, its effects will be less noticeable in the final dither. Bear in mind that each of these pixels will have errors too. The result - if it's done properly - will be to set up a series of alternating black and white pixels that look a lot like the grey scale of the original..."

With that out of the way, the only thing left is to talk briefly about filters. As you would expect, people have found different ways to diffuse the error to the surrounding pixels. The manner in which the errors are diffused as often referred to as the filter process and they tend to bear the names of their creator. Some of the more common filters are the Floyd-Steinberg, Burkes and Stucki filters.

Next month, we will discuss the actual dither process, list the filter values for some of the more common filters and explain how to convert the monochrome image to Page P1 Picture format.



# ASGARD SOFTWARE NEWS RELEASE



Subject: ASGARD 128K MEMORY SYSTEM RELEASED

For more information contact:

Asgard  
Attn: Chris Bobbitt  
P.O. Box 10697  
Rockville, MD 20849-0697  
(703)255-3005

Asgard is pleased to announce the completion and eminent availability of the ASGARD 128K MEMORY SYSTEM.

The AMS is a product of a two-year research and development program focused on increasing TI-99/4A memory capacity. Designed by a team of hardware and software experts guided by experienced businessmen, and with the assistance and insights of a wide range of users, this device represents the beginning of a new direction, as well as a blending of new and proven technology.

The AMS combines flexibility with reliability and compatibility. It is the first advanced memory system for the 99/4A designed to be used exclusively as memory for programs and data.

When installed in your Peripheral Expansion Box it functions as a 32K card with standard TI-99/4A software. It is completely transparent to virtually every other TI-99/4A peripheral - it will not conflict with any floppy or hard drive controller, or even some RAM-disks. The card does not need to be configured - simply plug it in and turn on your computer. Because it uses little power the AMS is highly reliable.

Programs designed to work with the card can access up to 128K of CPU memory simply and with a minimum of restrictions on program design. Memory can be barked in 4K increments, within a few clock cycles, anywhere within the standard 32K memory space available to TI-99/4A programs. The design used by AMS is similar to that used by TI in their TI-99/8 computer - and is currently readily

accessible to programs written in Assembly and SPL.

To assist in programming for the AMS example programs with source code as well as extensive technical documentation, included with the device. All materials were prepared by software designers to be as clear and comprehensive as possible to programmers - and not just other hardware designers. The result is what we believe to be the easiest to program extended memory device for the TI-99/4A.

For non-programmers, AMS will open the door to a variety of new programs currently under development by some of the brightest programmers in the TI community today. With four times as much space available, AMS compatible programs will be more capable, faster, and have much more capacity for storing data. Types of programs can be written that would be impossible in 32K. Compatible languages under development will allow even casual programmers to write programs with access to the memory.

AMS is not just a promise of new possibilities, it also represents a different way of doing things as well as a different approach to past problems.

While it may seem unusual that a software company would take the initiative in producing a new memory card, it's not so strange when you consider that you need software to make hardware useful, and a software company can insure that some of that software is written.

Further, to break with the long history of some developers (including TI) of playing favorites and of secrecy, Asgard guarantees we will freely provide any and all software developers as much information needed to take advantage of the AMS. It's time to end the games that have hurt the community in the past, and to bury the hatchet somewhere other than in each others' backs.

Finally, since no one likes to buy something that becomes obsolete tomorrow - all users can be assured an investment in AMS will be protected by a company that has been serving the TI community for 10 years. Asgard will provide reasonably priced upgrades and even trade-in options as we continue to develop this technology. Further, any software written for AMS will be fully compatible with future developments with few if any changes.

AMS is not an end in itself, it is a beginning on a path liberating the TI-99/4A from memory constraints. It also represents a new way to do business in the TI community.

The ASGARD 128K MEMORY SYSTEM requires a TI-99/4A with a Peripheral Expansion Box and a disk system. It is compatible with all disk controllers, all video cards, and some RAM disks and memory cards, as well as virtually all other cards for the TI-99/4A. It is not guaranteed to function with the Myarc or Comcomp RAM-disks, or the TI, Comcomp or Myarc 32K cards. No problems have been encountered with Horizon RAM-disks to date.

The suggested retail price of the AMS is \$119.95. At this

time all design and testing of the design has been completed, and it is expected to be in stock by the end of September.

To order, send a check or money order for \$119.95, plus \$10.00 S&H (in North America - \$20.00 for Airmail shipping elsewhere) to:

Asgard Peripherals  
P.O. Box 10697  
Rockville, MD 20849-0697

100 and credit card orders are not accepted. All orders to U.S. customers will be shipped via UPS Ground - please allow 4-6 weeks for delivery.

Programmers may receive a free packet containing programming information by sending a post card to the above address. Again, please allow 4-6 weeks for delivery.

# 99/4A - 9640 VENDORS



Alboes Computer Supplies  
5298 Hamilton Rd.  
36 Main Street Village  
Columbus, Ga 31909  
(404) 327-4900

Asgard Software  
P.O. Box 10696  
Rockville, Md 20850  
(703) 255-3085  
Catalog Available

Braatz Computer Services  
719 East Byrd Street  
Appleton, Wisc 54311  
(414) 731-3478  
(414) 731-4320 after 6pm  
Catalog #2

Bud Mills Services  
166 Dartmouth Dr  
Toledo, Ohio 43614  
(419) 385-5946

CaDD Electronics  
52 Audubon Rd.  
Haverhill, Ma 01830  
(603) 895-0119

Competition Computer  
2629 W National Ave  
Milwaukee, Wisc 53204  
(800) 242-7902 in Wisconsin  
(800) 662-9253 all others  
Catalog #1

Comproline  
1949 Evergreen Ave.  
Fullerton, CA 92635  
(714) 990-4577 12-9pm PT

Computer Shopper  
P.O. Box F  
Titusville, FL 32781

CorComp  
2211-G East Winston Rd.  
Anaheim, Ca 92806  
(714) 956-4450

Dijit Systems  
4345 Hortensia St  
San Diego, Ca 92103  
(619) 295-3301 voice  
(619) 278-8155 bbs

Disk Only Software  
P.O. Box 244  
Lorton, Va 22079  
(310) 340-7179

Genial Computerware  
335 Green Valley Dr.  
Philadelphia, Pa 19128  
(215) 488-1379

Great Lakes Software  
304 E. Grand River Ave.  
Howell, Mi 48843  
(517) 546-0566

Harrison Software  
5705 40th Place  
Hyattsville, Md 20781  
(301) 277-3467

Hunter Electronics  
4 N. 370 Pine Grove  
Bensenville, IL 60106  
(312) 766-9503

Insebot Inc.  
P.O. Box 29160  
Ft Orange, FL 32029

Jim Leshar  
722 Huntley  
Dallas, Tx 75214  
(214) 821-9274

Joy Electronics  
P.O. Box 542546  
Dallas, Tx 75354-2526  
(800) 422-3892 in Texas  
(800) 527-7438 all others

L.L. Conner Enterprise  
1521 Ferry St.  
Lafayette, In 47904  
(317) 742-8146

McCann Software  
P.O. Box 34160  
Omaha, Ne 68134

Micropendium  
P.O. Box 1343  
Round Rock, Tx 73680  
(512) 255-1512

Midwest Engineering  
203 Arcadia Dr.  
Vernon Hills, IL 60061  
(312) 362-3034

Muaco Inc  
2524 Ramier Drive NE  
Birmingham, AL 35215  
(205) 854-5843

Not Polyoptics  
P.O. Box 0443  
Woodbridge, Va 22191  
(703) 499-5543

Oasis Pensive Abacutors Of  
432 Jarvis Street  
Suite 502  
Toronto, Ontario  
CANADA M4V-2H3  
(416) 960-0925

Pilgrims Pride  
5 Williams Ln.  
Hatboro, Pa. 19040  
(215) 441-4262

Program Innovators  
4122 Glenway  
Mauwatosa, WI 53222

Quality 99 Software  
1884 Columbia RD #1021  
Washington, DC 20009  
(202) 667-3574

Queen Anne Computer Shoppe  
6102 Roosevelt Way NE  
Seattle, Wa 98115  
(206) 522-6558

Ranchage Computers  
6467 E. Vancey Dr.  
Brockpark, Oh 44142  
(216) 243-1244 evenings

Rave 99  
112 Rambling Road  
Vernon, Ct 06066  
(203) 871-7824

Joe Ross  
119 Knollwood Terrace  
Clifton, NJ 07012

Tenex  
P.O. Box 5578  
South Bend, In 46660  
(800) 343-2778  
(217) 253-7051

Texaments  
53 Center St  
Patchogue, N.Y. 11772  
(516) 475-3480  
(516) 475-6463 bbs

Tex-Comp  
P.O. Box 99084  
Grenada Hills, Ca 91344  
(818) 366-6631  
Catalog \$2

The Bunyard Group  
P.O. Box 62929  
Colorado Springs, Co  
80962-2923  
(719) 488-2572

Tigercub Software  
156 Collingwood Ave.  
Columbus, Oh 43213  
(614) 295-9545

TM Direct Product Marketing  
1650 Broadway  
Redwood City, Ca 94063  
1-800-336-9966  
Catalog Available

Trio+ Software  
P.O. Box 114-A  
Liscomb, Ia 50148

9648 News  
c/o Beery Miller  
5455 Marina Cove #1  
Memphis, Tn 38115

# Joke Of The Month



Two drunks were well in their cups at their favorite watering hole when one spotted movement on the bar top. "Whazz that?" he asked. "A bug?" "Iz a ladybug", his drinking pal replied. "Damn," the first gushed, "you have good eyesight!"

# Newsletter Exchange

Last month, I published a exchange column. only to have another new exchange start a few days later. The following list represents the latest and greatest exchange list for our group. We look forward to hearing from any group wishing to initiate an exchange and encourage others to use this list for communication with other groups. If you know of any UG (not on our list) that would like to exchange newsletters, we'd be very interested in hearing from you via the comments and suggestions on the cover page.

North Jersey TI IBM UG  
16 Judith Ann Dr.  
Ringwood, NJ 07456-1893

Central Garden State JG  
61 Country Lane  
Hamilton Square, NJ 08690

LII 99er's UG  
93 Myers Avenue  
Hicksville, NY 11801-2424

Twin Tiers UG  
c/o R. Sass  
97 Fir Tree Point Rd.  
Rock Stream, NY 14878-9707

Pittsburg User's Group  
P.O. Box 8043  
Pittsburg, PA 15216

Erie 99'er User Group  
2812 West 33rd Street  
Erie, PA 16506

Lahigh 99er Computer Group  
c/o Jack Cawedick  
219 N 12th Street  
Alientown, PA 13102

\*MANNERS  
c/o Bill Howard  
15204 Louis Mill Dr  
Chantilly, VA 22021

Broward Computer Group  
c/o Mark Wacholtz  
2141 NW 64th Ave #15  
Sunrise, FL 33313-3950

CONNI  
161 Helsehman Ave  
Worthington, OH 43085

N.W. OHIO 99'ers User Group  
\*First Church Unity  
3535 Executive Parkway  
Toledo, OH 43606  
Attn: Earl W. Hoffsis

\*Cleveland Area 99/4a UG  
c/o Harry Hoffman  
1925 Trowbridge Ave  
Cleveland, OH 44109

\*Greater Akron 99er's  
c/o Jim Harris  
2022 Tenth Street  
Cuyahoga Falls, OH 44221

Lima 99/4a Users Group  
P.O. Box 547  
Venedocia, OH 45894

Great Lakes Computer Group  
P.O. Box 152  
Roseville, MI 48066-0152

Milwaukee Area Users Group  
4122 N. Glenway  
Wauwatosa, WI 53222

Siouxland 99er's  
c/o Bob Daggitt  
4606 Bluestem Circle  
Sioux Falls, SD 57106

Chicago TI User's Group  
P.O. Box 578341  
Chicago, IL 60657

\*Kansas City TI99/4a UG  
c/o W. Blood  
2032 N. 32nd Street  
Kansas City, KS 66104

Gallop TI Home Computer  
P.O. Box 29863  
Dallas, TX 75229

\*Net99er HOCUS  
c/o Barbara Massey  
3209 Amber Dr. S.  
Fort Worth, TX 76133

\*Houston AUG  
c/o R. Lumpkin  
11610 Inga Lane  
Houston, TX 77064

\*JSC TI99 User Group  
c/o P.C. Van Nordstrand  
490 Shadow Creek Rd.  
Seabrook, TX 77586

The FRONT RANGER  
P.O. Box 9572  
Colorado Springs, CO  
80932-9572

\*TI SLaves  
1396 Lincoln Apt #B  
Ogden, UT 84404

Southwest Ninety Niners  
P.O. Box 17831  
Tucson, AZ 85730

Southern Nevada UG (SNUG)  
P.O. Box 26307  
Las Vegas, NV 89126-0301

\*LA 99ers Computer Group  
7730 Emerson Ave  
Los Angeles, CA 90045

North County 99ers UG  
P.O. Box 2500  
Escondido, CA 92025

UGOC  
c/o Earl G Raguse  
17161 Edwards  
Huntington Beach, CA 9264

WORDPLAY  
The PUNN Newsletter  
P.O. Box 15037  
Portland, OR 97215

\*Kawartha 99er's  
224 Woodward Ave  
Peterborough, Ontario  
CANADA K9L 1J7

## COMMENTS & SUGGESTIONS

9T9 User Group  
c/o Steve Mickelson  
15 Kenedale Avenue  
Toronto, Ontario  
CANADA M6N 1C9

Sudbury 99er's  
c/o Jim McLaren  
Site 1 Box 7  
Whitefish, Ontario  
CANADA P0N 3E0

B.C. 99er Users  
c/o Ron Warfield  
216 10th Ave  
New Westminster, B.C.  
CANADA V3L 2B2

DETACH THIS PAGE FROM THE NEWSLETTER AND SUBMIT IT TO  
THE EDITOR AT THE NEXT CLUB MEETING YOU ATTEND OR MAIL  
IT TO: NEWJUG 99ER'S USER GROUP  
P.O. BOX 1463  
SAVREVILLE, NEW JERSEY 08871-1463