FROM:

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

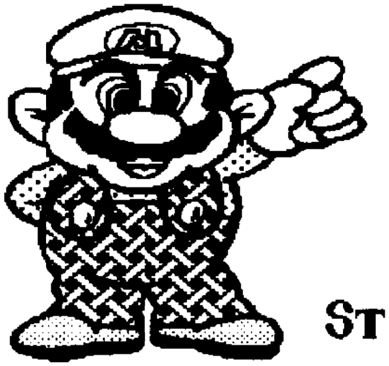
POSTAGE
GOES
HERE

TO: AFFIX MAILING LABEL
HERE

NEWJUG 99ER'S NEWS

NEWJUG 99ER'S NEWS

MARCH 1993



YOU DON'T HAVE TO BE IRISH TO CELEBRATE ST. PATRICK'S DAY

REMEMBER: ALCOHOL AND GASOLINE DON'T MIX

Highlights:

DELEKTO CONFERENCE	02
BITMAP GRAPHICS	10
NEWSLETTER EXCHANGE	14
JOKE OF THE MONTH	15
WANTED	16

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
SAYREVILLE, N.J.
08871-1463

JOE DELEKTO IN CONFERENCE

Delphi TI NET - January 31, 1993

Attendees:

JERRYC	
JDELEKTO	Joe Delekto
C_BJBBITT	Chris_B
FROBOTZ	HTE
TINET	Gary/OPA
MICBRO	
EICHER	DarO
BOOJRNBOS	
CAL47	
JHWHITE	(joined us late)

JERRYC> Settle down everyone - we are about to go formal for an intro from Joe.

Joc Dclckto> I am a 22 year old Georgia Tech student, about to finish a 5-year (actually 4) degree in Electrical Engineering, with a certificate in Computer Engineering.I was first interested in computers at the age of 10, and received my TI-99/4A when I was 12 years old. Extended BASIC was preferred over BASIC,but later Assembly was the "in thing." Having c99 made coding software faster, and being a 'C' and assembly programmer, I love to combine the two.Past projects have been The TMS 9900 Clipboard, an on-disk assembly and c99 tutorial, as well as Screen Preview, and Starbase Raiders for Asgard Software....I have MANY other half-done, or unfinished projects scattered among the hundreds of disks on my floor. Right now AEMS has taken precidence....

.Gary/OPA> Can you tell me more about TMS9900 Clipboard. Will all the issues be uploaded to DELPHI? And will more be made? (I never see an issue of it)

Joe Delekto> I will upload all the issues as soon as I get my Archiver to work. The TMS 9900 Clipboard consisted of a flippy disk, with one side containing articles,and the other side

containing programs, along with the source code. It was distributed by the now deceased Atlanta 99er User's Group,and only 11 volumes have been finished. There were maybe 10 people who responded to them that I Know of, so we stopped all work on it....

Gary/OPA> C: Sounds like it will be helpful for me to download a set for the local 9T9 UG Assembly SIG

JERRVC> c: We have 3 full disks in the library in 6 archived files - 7 more disks to come.

Joe Delekto> That means 14 more files Jerry! <<< EOT >>>

JERRVC> c: I read several articles while checking the files and found them very readable.

.Joc Dclckto> I must also mention, that my cohort, Jon Dyer, Known to myself as the Graphics Wizard, helped write much of the articles and software.....

JERRVC> Joe would you like to describe some of the contents?

.Joc Dclckto> Well, we have a wide range of things, from c99 routines, to macros, to fully-functional software. Some of which were CFONT, the CHARA1 file editor, Jon's Banner Maker, a File Viewer, Disk Copier. Also an operating system written for Mini Memory and Superspace, along with some software that takes advantage of them....Along with the operating system was a fully-featured 'C' PreProcessor, GREP, format, delete, and other utilities.We did speech functions, window functions (that look like those in Telco), and memory allocation routines for c99.I also have a demonstration of Time-Slicing routines on the TI-99/4A, using a c99 function vector, as well as an article on creating an operating system.....

.Joc Dclckto> Questions?

.HTB> Since you've already had some disk copy experience, how about a disk copier for the ams that uses its memory for a single pass... copy for those of us who have to make them by the hundred? PLEASE!?

.Joe Delekto> Jon and I discussed doing that. It would be relatively easy to modify the existing program, since it just does a sector copy.Of course, we can't speed up the TI. Yet.....Of course, Only a SS/SD disk can be done in one pass on the AMS. The AEMS is another story...

.HTB> C: I was thinking of a 512K version

.Joe Delekto> Sure. Chris, get me a 512K chip! <grin>

DOORNBOS> C: Make it a track copier for a one drive system.

HTB> C: It should look to see how much memory is there and then use what there is.

Joe Delekto> Sounds good Brian, but I haven't gotten into track copies yet..... The AEMS boot reads the system file which tells the resident routines (like alloc) how much memory is there...

DOORNBOS> C: I've been looking for one for a long time.

Joe Delekto> Sounds like a good long project..... Just give me the routines to read a track....

ERRYC> c: track copiers are specific to an FDC chip - so you would need several I think.

Chris_B> Or an awfully smart program

ERRYC> c: Right Chris - Hypercopy has a smart loader and 3 separate routines. Some operating systems have track read/write functions, but not TI.

Joe Delekto> Anyone in the background have a question?

Chris_B> Joe; would you like to describe what you are doing with the AMS/AEMS.. or what you are collaborating on?

Joe Delekto> I wrote most of the library routines for the AMS/AEMS card. The memory allocation, memory move, far VDP read and write, as well as later additions.I am currently working on a 32 bit addressing scheme, where pointers will use 32 bit addresses, and can call routines to do 32 bit arithmetic.With them, all a person needs to do is add the code to convert a 32 bit address to their hardware's PAGE/OFFSET, and the code to map in a block....That will help porting software to other memory cards....

Chris_B> Joe: could you describe to some cegree how programming for extended A(E)MS memory will differ from programming with the standard 32K?

Joe Delekto> The only difference is how the page numbers are written to the mapper registers.Right now, the page number is loaded into the most significant byte of a register, or data word, and then written to the mapper registers.AEMS will use an actual page number, in the full 32 bit word. For example, page 10, which on AMS is >0A00, will now be >000A.Other than that, everything will be the same as is on AMS. Very little modification is required if EQUates are used....Because the TMS9900 only has a 16 bit program counter, we can only

address 64K at a time. However, we only use 32k RAM at a time with AMS, by mapping over>2000->3FFF, and >A000->FFFF. (Excuse me, AEMS, AMS only uses 24k at a time)

Because we cannot write a 1 MEG program (as it's program counter will be well over 65,536) we have to use an overlay method..... The AMS/AEMS mapper registers work similar to the base pointer registers in a PC.According to one of my hardware sources, if you write >FFFF to mapper register 2 at >4002, you can tell which card is installed.

[illegible] C: In fact, the first RAM card used to expand the early 64K IBM PC used a '612 on it!Read the value back, and if it is >7F7F, then it is AMS. If it is >0FFF, then it is AEMS....

.Joe Delekto> Anyhow.... On overlays....We only use mapper registers 2,3 and 10-15 for AEMS. If you will note, it corresponds to the most significant nibble of the TI's standard 32K address.Register 2 maps onto >2000, register 3, onto >3000, and 10 (>A) onto >A000 etc.This made the hardware easy to design, and the software an expanded memory programmer's dream. (in my opinion)We split up the memory into 4K blocks, for a total of 32k. (That's low, and high standard 32k memory). These different 4K address ranges are called "frames."When you write a page number into a mapper register, say register 2 for instance, then the frame >2000->2FFF points to that 4K page in memory.We chose 4k blocks, because it was ideal in writing overlay code.Overlay programs are special. They consist of a root program, which always stays in memory, and contains about 12 lines of code used to call in overlays whenneeded. The overlay consists of BLWP subroutines, loaded in when necessary.:Not many routines take up 4k, but many can be packed within an overlay.Overlays may be up to 24k in length (max length contiguous in high memory) and must fall on 4K boundaries.

The overlay manager in the root segment calls the overlay by: 1) writing the overlay's pages into the corresponding mapper registers where the overlay'saddress frame resides, then branching to it. The overlay then returns from where it came. Art Green's linker is designed to handle all the paging code foroverlays. Sorry for the mouthful... Any questions?

.Joe Delekto> Any questions NOT regarding AEMS? (please?)

JERRYC> c: I suppose in a complex program you might have similar code segments in several overlays (?)

Joe Delekto> Yes, you could. But you have enough memory to do that. Overlays loaded previous to higher levels of overlays may be called without generating another overlay

Joe Delekto> Anyone notice I haven't said anything yet about the BASIC or 'C' compilers?

ERRYC> Can you page in data space as well - to simulate a very large buffer?

Joe Delekto> Jerry: You sure can, and even use the utility routines to dump buffer data back and forth to udp, or move it around in memory.

ERRYC> c: I am thinking of the way MyWord and Gen-TRi use memory in GPL mode on the 9640.

DOORNBOS> Do you have any ideas for future projects?

Chris_B> (Ah - the long awaited change of subject :>)

Joe Delekto> I would like to finish a full K&R 'C' compiler for AEMS. I also have a hardware project in mind, and Jon Dyer, Jim Krytch, and I will write some new GAMES!

Chris_B> The ultimate 1Mb arcade game?

Joe Delekto> Many many many many many levels....

DOORNBOS> Like to see a Jet simulator.

Joe Delekto> I wish the speed for the 4A would allow such a beast! I could however, see doing one w.th a small cockpit interior, a small window, and sprites....

Chris_B> What kind of games?

Joe Delekto> Arcade mostly. Lends itself well to many levels, many different graphics setups, lots of music, etc.....

.Chris_B> Joe: Where do you see development going for the AMS - or rather, what would.. you like to see done with it? ga

.Joe Delekto> I would like to see a graphical user interface. I would also like to see programming languages, and operating systems ported over.There also needs to be a COMPLETE desktop publishing kit, and perhaps even a simple CAD system....

--- Oh yes, I would also like to see existing programs like Telco, Picasso, and TI-Artist take advantage of the memory to store the software...

JERRYC> I have a question -- how can more memory create the impression of higher speed?

.Joe Delekto> It won't really create the impression of higher speed as far as program execution, but for programs that load overlays from a disk will be noticebly fast.Even faster than a RAMdisk, since execution is instantaneous, rather than being loaded, then executed....

JERRYC> How about switching screens and windows and such?

.Joe Delekto> Well, for windows, more memory is available to save the overlap portion of the screen. Switching screens isn't noticeably faster,but large graphics such as TI Artist will fit completely in memory. For animation, one only needs to write the WHOLE picture from RAM into the VDP. It's very FAST, and doesn't need the disk except for initial loading....

JERRYC> c: that's what I had in mind.

.Joe Delekto> Too bad Jeff and Don aren't here. They promised to give me a run for the money....

.Chris_B> The title fight of the decade, deferred.

JERRYC> I know you have had chats with Don - what are your impressions of the differences between A(E)MS and the proposed 4a-MEMEX concept?

.Joe Delekto> Well, Memex offers perks such as a spooler and RAMdisk. We only wanted MEMORY. It's nice to have the other features, but what they accomplish throughhardware, is more expensive than what we can do through software. It is my opinion that AMS/AEMS is the probably one of the most "programmer-friendly" piecesof hardvare, and as long as people can develop for it quickly, and easily, there will be a large softare base in no time. Memex will be along a little late in the game, but will probably appeal mainly to Geneve users....

I can say this however -- Just like AEMS, programmers will have to use some sort of paging/overlay scheme. There's no EASY way around it. If they can't get software support for theircard soon, then the number of programmers for it will decrease.

JERRYC> OK - if software is "easy" - how about a progran to run a simulated ramdisk through a 4K "data window"?

.Joe Delekto> It's possible. But I don't know if it can be used as a standard 'DSKx' device without a DSR. With a new DSRlink to trap such a setup, it could be done....

Joe Delekto> The overall goal of AEMS. was not to be used mainly as a RAMdisk though. MANY E/A #5 programs can be loaded. and made resident. then executed instantaneously!

Chris_B> Joe: as Art's menu utility allows right now.

Joe Delekto> Right. Except many more programs resident, including overlay programs for AEMS. The nice thing about AMS paging, is that E/A #5 programs are "page relocatable" or can be fragmented into 4K chunks and spread around memory....

JERRVC> c: a well thought out spec could result in "ramdisk-compatible" AEMS programs.

Gary/OPA> c: OPA is working on two projects. One is a new DSRlink (needed to support programs in our POP-CART), this could be used in the future for the A(E)MS board. The second is a new better operating system, installed like the SOB. This O.S. could support memory like the AMS system. And its use could be divided into ramdisks, etc.

JERRVC> Bingo!

Chris_B> Sounds wonderful to me!

Joe Delekto> That's Excellent Gary!

Gary/OPA> In my last week CO, I mention the that, in talking with users and programmers, by not making the AMS board a RAMdisk FIRST, it would be more likely that programmers will make use of the memory for larger programs, as that is what it is for. By making it a ramdisk, sales may be higher, but I found programmers don't look at the "hardware" in the same fashion.

Joe Delekto> Gary is correct. Programmer's get the idea that a RAMdisk is the paging nightmare. Expanded Memory pretty much stands on it's own.

Chris_B> (Well, *I'm* excited)

JERRVC> Any more discussion before I close the official log?

Joe Delekto> Well Jerry, it tapered off quickly. I think I got a little carried away earlier, and scared everyone off! <hehe>

Gary/OPA> (I would like to see more discussion, but since I Know almost everything about the AMS project, its hard to think of ?'s) I am sure Don and Jeff if they were on the system, could Kept this CO going like mine, until 7am!

.Chris_B> Well, no controversy either

TDOORNBOS> OK, See ya guys later.

Joe Delekto> Perhaps I can come back when Jeff & Don are available..... <hehe>

Gary/OPA> I think another AMS CO is in the order, once the Art Green package is released. I would like to see all the AMS people online at once. Art, Jim, Joe, Chris

.Joe Delekto> That would be great! You'd have to be there Gary!

Gary/OPA> (I will try to make it) (Sunday is normally a "free" day for me)

.Chris_B> We can drag Harry, Art Gibson and maybe Brad Snyder would drop in.

JERRVC> Sounds good to me.

.Gary/OPA> Yes, it would also be good for a AMS CO, to have all the current authors of AMS programs online too!

.Joe Delekto> I'm sure people wanted to know how my BASIC compiler was doing, but like in the past, they dropped out before I could get to it....

.Joe Delekto> Suppose it could wait another 4 years....

.Chris_B> Maybe just as well, Joe.

.Chris_B> You might want to finish the K&R first :>

.Joe Delekto> Of course....

JERRVC> Joe - do you want to say a few words about the BASIC compiler for the transcript?

.Joe Delekto> Well, I would like to say I have a small vanilla BASIC compiler. It is written in 'C', and will handle only BASIC right now. Parsing is difficult, since people want to print their program to the disk, and compile. However, there is no line terminator, and it's hard to tell where lines are split up! The compiler I have works only with single line instructions, and generates BLWP calls for ALL commands. All arithmetic is integer, but will eventually be ALL floating point, making use of all the built-in ROM routines.

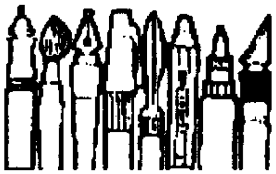
.Chris_B> Joe: how about parsing the tokens?

.Joe Delekto> Parsing the tokens isn't too difficult. However, PRINT, INPUT, and OPEN are the most difficult, as they have variable argument lengths. I just parse the objects seperating them into numbers, strings, and variables, then pass an argument count, as well as the addresses of the objects to the BLWP routine.I use a simulated stack for all calls to the compiler's internal rcutines.Right now, no sound, or graphics are supported. Was in the middle of working on them when Screen Preview popped up....

Chris_B> I figured it would make it a little easier.

Joe Delekto> Since I prefer 'C' much more than BASIC, it's in Joe's best interest to finish his 'C' compiler.

*** CO officially ended after about 2 hours, but was followed by a chat among the developers after Jeff White arrived - Ed.



BITMAP GRAPHICS

BY

ERIC LAFORTUNE

it-map modes on the TI-99/4A

his document was created as a result of a discussion about raphics modes for the TI-99/4A, on Delphi, 17 January 993. It discusses some documented and undccumented pplications of bit-map modes. Enjoy!

ric Lafortune (Eric.Lafortune@cs.Kuleuven.ac.be)

ontents:

1. Regular bit-map mode
2. Mixing text and graphics in regular bit-map mode
3. A hybrid graphics/bit-map mode

1. Regular bit-map mode

his is the mode as it is described in the Editor/Assemblcr anual. The Screen Image Table is placed starting at >1800 r any other unused region of VDP memory. It is nitialized to three times 0...255. The Pattern Descriptor able then determines the setting of each individual pixcl n the screen and the Color Table determines the color of ach row of 8 consecutive pixels.

dvantages:
- This mode offers the largest degree of freedom to draw on the screen.

```
VDP register definitions:
  R3 = >07 => >2000 = Color Table              (>1800 bytes
  R4 - >7F -> >0000 - Pattern Descriptor Table (>1800 bytes

Screen Image Table (>300 bytes):

+------------------------------+    \
+ 0 1 2 ...                    +    !
+                              +    ! First third of the
+                              +    ! Screen Image Table.
+                      ... 255 +   /
+------------------------------+   \
+ 0 1 2 ...                    +    !
+                              +    ! Second third of the
+                              +    ! Screen Image Table.
+                      ... 255 +   /
+------------------------------+   \
+ 0 1 2 ...                    +    !
+                              +    ! Last third of the
+                              +    ! Screen Image Table.
+                      ... 255 +   /
+------------------------------+

Pattern Descriptor Table (>1800 bytes):

+---------------------------------------------+
+                                             +
+                                             +
+ Pattern definitions of the pixels on the    +
+ screen.                                     +
+                                             +
+                                             +
+---------------------------------------------+

Color Table (>1800 bytes):

+---------------------------------------------+
+                                             +
+                                             +
+ Color definitions of the 8-pixel rows on    +
+ the screen.                                 +
+                                             +
+                                             +
+---------------------------------------------+
```

2. Mixing text and graphics in regular bit-map mode

Instead of initializing the Screen Image Table to the usual 0...256 three times in a row, one can use any of the three parts of the screen for text input and output as in ordinary graphics mode. For this purpose the Pattern Descriptor Table and the Color Table have to be initialized accordingly.

Advantages:
- Printing text to the screen is simple and fast.
- Each character can have its own color; or each character can even have several colors. This can be used to obtain rainbow effects or understriking in a different color for instance.

```
'DP register definitions:
  R3 - >07 -> >2000 - Color Table                (>1800 bytes)
  R4 - >7F -> >0000 - Pattern Descriptor Table  (>1800 bytes)
```

or example, in an adventure game the top two thirds of a creen may be used for pictures and the bottom third for ext input and output. The following setup accomplishes his effect.

creen Image Table (>300 bytes):

```
+-----------------------------+   \
+ 0 1 2 ...                   +   |
+                             +   | First third of the
+                             +   | Screen Image Tablc
+                     ... 255 +  <
+-----------------------------+   \
+ 0 1 2 ...                   +   |
+                             +   | Second third of the
+                             +   | Screen Image Table.
+                     ... 255 +  <
+-----------------------------+   \
+                             +   |
+ Any text or graphics        +   | Last third of the
+ characters.                 +   | Screen Image Table.
+                             +   /
+-----------------------------+
```

attern Descriptor Table (>1800 bytes):

```
+-------------------------------------+ \
+                                     +  |
+                                     +  |
+ Pattern definitions of the pixels   +  | First two thirds
+ on the top two thirds of the screen,+  | of the Pattern
+ ie. the picture.                    +  | Descriptor Table
+ picture.                            +  | (>1000 bytes)
+                                     +  |
+                                     + <
+-------------------------------------+ \
+                                     +  | Last third of
+ Standard definitions for text       +  | the Pattern
+ characters.                         +  | Descriptor Table
+                                     +  | (>800 bytes)
+-------------------------------------+ /
```

olor Table (>1800 bytes):

```
+-------------------------------------+ \
+                                     +  |
+ Color definitions of the 8-pixel    +  | First two
+ rows on the top two thirds of the   +  | thirds of the
+ screen.                             +  | Color Table.
+                                     +  | (>1000 bytes)
+                                     + <
+-------------------------------------+ \
+                                     +  | Last third of
+ Color definitions for the character +  | the Color Table
+ set.                                +  | (>800 bytes)
+                                     +  |
+-------------------------------------+ /
```

4. A hybrid graphics/bit-map mode

This mode was used for the first time in the game Rock Runner. It is based on the same idea as in the mixed text/graphics mode above. A static character set is not only used for some part of the screen but for the whole screen.
Furthermore a single character set is used instead of three independent sets for each third of the screen. The result is a classical graphics mode with more versatile color definitions. For this purpose the VDP registers have to be initialized to some officially undocumented values. The mode can be used on both the TMS9918 and the more recent MS9938 video processors however.

The advantages are mostly the same as in mixed text/graphics mode:

- Printing and drawing is fast as it only involves writing to the smaller Screen Image Table most of the time.
- Each character can have its own color, or several colors
- Color cycling (as used for the diamonds in Rock Runner) and pattern cycling (as used for the lava in Rock Runner) can be achieved easily by changing the definition of a single character.
- The mode requires less memory than a full bit-map mode.

```
VDP register definitions:
  R3 - >9F -> >2000 - Color Table               (>800 bytes)
  R4 - >00 -> >0000 - Pattern Descriptor Table  (>800 bytes)
```

Screen Image Table (>300 bytes):

```
+-----------------------------+
+                             +
+ Any text or graphics        +
+ characters.                 +
+                             +
+-----------------------------+
```

Pattern Descriptor Table (>800 bytes):

```
+-------------------------------------------+
+                                           +
+ Pattern definitions for the character set +
+                                           +
+-------------------------------------------+
```

Color Table (>800 bytes):

```
+-------------------------------------------+
+                                           +
+ Color definitions for the character set   +
+ (8 foreground and background colors per   +
+ character)                                +
+                                           +
+-------------------------------------------+
```

Newsletter Exchange

ıfter reviewing the past ıcar's newsletter exchanges ıith other groups, it has ecome necessary to trim he exchange list. This lownsizing has nothing to ıo with originality or ıuality of the other groups ıewsletters and everything o do with the infrequent ırrival of their ıcwsletter. To this end, we ıre no longer able (or ıilling) to eat the postage ·osts without an equitable xchange. I expect this ist will change from time o time and I'll try to eep it as current as ıumanly possible.
ıe look forward to hearing rom any group wishing to nitiate an exchange and ncourage others to use his list for communication ıith other groups. If you now of any UG (not on our ist) that would like to ·xchange newsletters, we'd ·e very interested in ·earing from you via the omments and suggestions on he cover page.

North Jersey TI IBM UG
16 Judith Ann Dr.
Ringwood, NJ 07456-1863

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

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

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

Erie 99'er User Group
2812 West 33rd Street
Erie, PA 15506

Lehigh 99er Computer Group
c/o Jack Zawediuk
719 N 12th Street
Allentown, PA 18102

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

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

Lima 99/4a Users Group
P.O. Box 647
Venedocia, OH 45894

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

Milwaukee Area Users Group
4122 N. Glenway
Wawatosa, 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

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

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

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

TI SLaVes
1396 Lincoln Apt #8
Ogden, UT 84404

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

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

UGOC
c/o Earl G Raguse
17161 Edwards
Huntington Beach, CA 92647

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

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

*Oshawa TI 99/4A UG
c/o Tom Jakabfy
660 Given Rd.
Oshawa, Ontario
CANADA L1H 8L7

9T9 User Group
c/o Steve Mickelson
15 Korsdale Avenue
Toronto, Ontario
CANADA M6M 1C9

Sudbury 99cr's
c/o Jim McLaren
Site 1 Box 7
Whitefish, Ontario
CANADA P0M 3E0

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

Joke Of The Month

Two Irishmen were digging a ditch directly across from a brothel. Suddenly, they saw a rabbi walk up to the front door, glance around and duck inside.
"Ah, will you look at that?" one ditchdigger said. "What's our world comin' to when men of the cloth are visitin' such places?"
A short time later, a Protestant minister walked up to the door and quietly slipped inside. "Do you believe that?" the workman exclaimed. "Why, 'tis no wonder the young people of today are so confused, what with the example clergymen set for them."
After an hour went by, the men watched as a Catholic priest quickly entered the whorehouse. "Ah, what a pity," the digger said, leaning on his shovel. "One of the poor lasses must be dyin'".

WANTED

COMMENTS & SUGGESTIONS

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
SAYREVILLE, NEW JERSEY 08871-1463