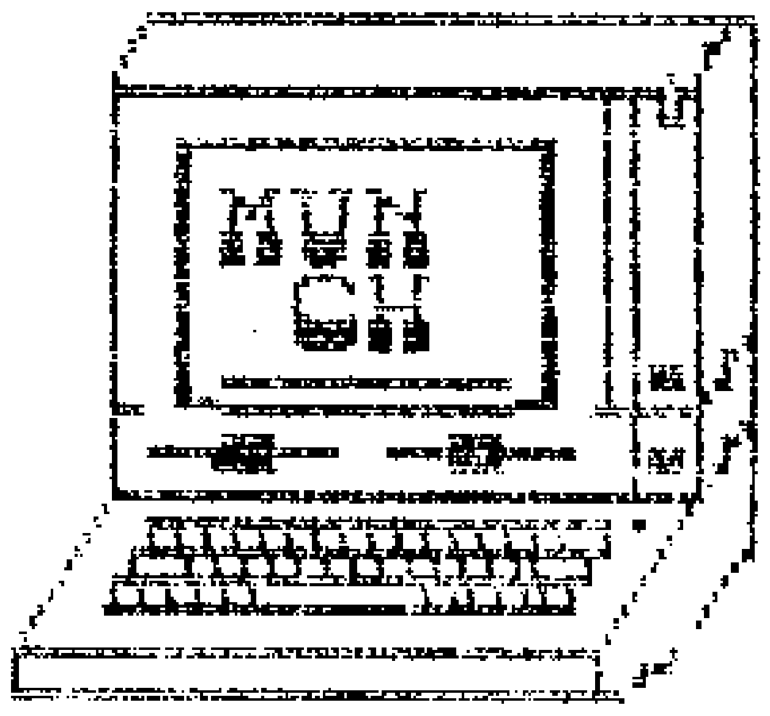
JOHN E. MONROE

John Monroe suffered a fatal heart attack while undergoing tests in the hospital two days after Christmas.

John was a man with a great sense of humor and was M.U.N.C.H.'s most genial member. He was liked by everyone and will be missed by us all.

To his widow and his family we extend our deepest sympathies.



Now, for under $300!

A complete IBM XT clone (Honeywell)
that includes two double/double drives
in console, full COLOR monitor, and
full-size extended keyboard. $295.00

Now you can have the best of both worlds for less than the cost of a RAMdisk or 80-Column card for the TI.

And for a lot less than the Geneve (where YOU supply the color monitor, the disk drives, and the P-Box.

If interested, call Jack (476-7680, and I will bring it to the January meeting. First person who calls will get it. Need the dough for a trip to Lima this spring.

Our Video

Because it will look quite peculiar having steam coming out of our mouths while we are wrapped in heavy-duty winter clothing and because we have only one 3-prong outlet in our entire clubhouse and because the lighting (not to mention the heat) are so lousy, we've decided to put off our video tape while we seek a better location, even if we have to make it a SAT or Sun.

We'd like to make a very professional tape. It needs to be scripted and chored. Perhaps at the Jan meeting we can discuss this for a few (many?) minutes.

We have lots of great ideas. Now we're ready for some input for the best execution of those ideas.

There's another updated version of FUNNELWEB 4.40 now available for copying at the next meeting. This version has updates by Tony through October (our last was July) and by Charlie though December. So it's hot off the press. You'll need two flippies.

John Monroe

John passed away just after Christmas. He was a good friend to all of us. He was a willing worker and spark for many of our activities.

Cards may be sent to his widow: Mrs. John E. Monroe
575 Stafford Street
Worcester MA 01610

TI BITS N' BYTES

by Val Mehling

What The Heck is a BOOLEAN?

Have you ever wondered what a Boolean Variable is, or what it might be used for? Booleans Variables (BV's from here on), are quite different from most other variables in that the range of values they can contain is somewhat limited. The entire range, in fact, covers only two possible choices. You can consider these choices to be True/False (the most common), Yes/No, On/Off or whatever pair of mutually exclusive conditions you desire. If you need more than two choices then what you need is not a BV type, and I'll get to that later.

Knowing when to use a BV shouldn't be too hard once you are familiar with them. For example, in the Disk Catalogue program I'm working on, I am keeping track of whether or not files have been Loaded or Saved, whether an Error condition exists, a disk name has been Duplicated, a disk is Full or whether I want to Quit the program. The first of these which I thought of was the SAVED flag. If you are working with any kind of data or text files and you add or change anything, you might be pretty upset if you decided to take a break and forgot to save the file first. Having the program look at a BV when you select quit, and remind you if files aren't saved, can save a lot of grief, not to mention hours of work. Now we'll have a look at how these BV's are set up in my program.

As is the case with most Basics, ours has built in BV's. True and False are defined as having the values of 0 and -1, respectively. You can stay with those values if you like or you can redefine them, as I did, so that True = 1 and False = 0. Either way, use of the BV's is the same. When a program is started, the flags are initialized to reflect the status at that time. The flags QUIT, EROR, DUP and FULL would all be set to False. SAVED would be set to true. If you changed your mind about working on something and wanted to quit before you loaded any data, there is no reason to caution you about unsaved files. Any routine which is used to update a file would automatically set the SAVED flag to False each time it was called. Your routine to Save the file would set the SAVED flag to True. Now, when you select Quit, this routine should look at the SAVED flag and then either allow you to Quit or issue a reminder about the unsaved file, and allow you to go back and save it. This is where the QUIT flag is used. If it is okay to leave, (or you chose to leave anyway), this flag comes back as True and your program then goes on to its Exit routine. If the QUIT flag comes back as False, you would be returned to the Main Menu.

For an example of how a BV might be used in conjunction with a SUB routine, we can try this one out.

```
XX10 SUB DISK_NAME(DISK$,NAME$,USED,FREE,EROR,TRUE)
XX20 ON ERROR XX60
XX30 OPEN #1:DISK$,INPUT,RELATIVE,INTERNAL
XX40 INPUT #1:NAME$,A,TOTAL,FREE :: USED = TOTAL - FREE
XX50 CLOSE #1
XX50 SUBEXIT
XX60 EROR = TRUE
XX70 SUBEND
```

The variables we are looking for, NAME$, USED and FREE, as well as DISK$ ("DSK?." ?=drive #), the EROR flag and the BV TRUE, are all passed to the SUB. SUB's can alter any variable that is passed to them (except constants) and pass the result back. The BV EROR was set to False before calling this SUB and only needs to be changed if there was an error. Since I have redefined my True and False values, I must pass TRUE in case it is needed. If you stick with the original values of True and False you shouldn't have to pass them into a SUB to use them. After calling this SUB, you would look at the flag EROR and proceed accordingly. Should the EROR flag come back true you can then make a call to ERR(C,T,S,N) (page 83) to see what the error is. According to the manual, making this call is supposed to clear the System's Error flag and then allow you to try again. Upon noticing the error you would want to be able to just, let's say close a drive door if you left it open, and try again. So would I. Unfortunately, I have found that this doesn't work on my system. It may work on yours but, if you are writing anything that will be used by other people you must allow for the possibility that their system may not handle this function properly. (Any info or ideas on this point would be appreciated.)

If you have been following this column, then you know we are trying to follow some specific rules with the creation of SUB routines. The one shown above can be saved and later included into any program where it might be needed without modification. Building program parts in this manner is certainly more work and requires a little more mental effort sometimes, but the end result is that you should soon find that you have developed a nice library of commonly used modules. When developing something new you won't get bogged down re-creating these parts. Instead you will be able to put more effort into the more difficult parts of your project.

While I have a little space left, I would like to mention those other flags that you might need sometime; the ones that can hold more than two values. Actually, by definition, a flag is really a BV. It is On/Off, Set/Reset (Up/Down ?), but nothing in between. The value that is obtained from making a menu selection might be in the range of 0 to 9, or 65 to 90 (ASCII A to Z), and this can't really be considered a flag type of variable. If you think about it, though, the smallest piece of data that the computer can pass around is a byte (if you use a character variable). Each byte consists of eight bits, each of which can be on or off. So, for the average BV, we actually have 8 bits doing the work of only 1. This usually doesn't matter but there have been times when I have wanted to make better use of those seven other bits (or 15 in a two byte variable). Being able to set/reset the individual bits in a variable isn't hard at all. A small array to hold the values array setup, and the SUBs that might be used, would look like this

Uh oh! The bottom of the page is getting dangerously close. One thing you never want to do is fall off the end of the page. I don't know what would happen but I'd rather not take the chance. I will continue with this part of the discussion next month. It shouldn't take more than a paragraph or two. And then.... who knows? I'm not sure what I'll be doing tomorrow, let alone next month. As always, though, I will do my best to be entertaining as well as informative. After all, if it isn't fun, why do it? And, in a more serious vain, "If it isn't good for the Earth, why do it?" Until next time, MTFBWY. Val M.

eprinted from
ERIE 99'ER U.G.
pril 1991

FROM THE TEACHER'S DESK-

Over the years, you've read much about the virtues of computers in schools - much of it in this Newsletter. Therefore, it is only fair to view the other side of the coin. To do this, I've selected comments found in a recent issue of SCHOOL TECH NEWS entitled:

"WHAT THE CRITICS ARE SAYING ABOUT COMPUTERS IN EDUCATION"

Are we spending too much money on computers in school? Sometimes we are, says Michael Wessells, Professor of Psychology at Randolph-Macon College in Ashland, Va.

Wessells thinks computers are used too often for "electronic page-turning" - presenting lengthy texts that could be handled just as well in book form - and for displaying "electronic flashcards" in learning languages. He thinks the money spent on such seemingly misguided use of computers in the classroom might be used more wisely by allocating costly computer time for "richer, more satisfying, more interactive activities that use the power of the computer" to better advantage. Wessells says he is neither skeptical nor "starry-eyed" where computers are concerned. He sees his job as "encouraging people to ask hard questions" about computers, especially computers in school

Computers have a significant place in education and a great contribution to make there, Wessells says. But as with other technologies such, such as educational television, "naive idealism" tends to make some people "see computers as the salvation of education," he points out. "I don't think they are."

Setting priorities for computer education is important in this age of fast-evolving technologies, Wessells says. What goal should schools aim for to make every student capable of using a state of-the-art computer? Programming is less important in the curriculum now, but schools still struggle with the issue of what to teach about computers, he adds.

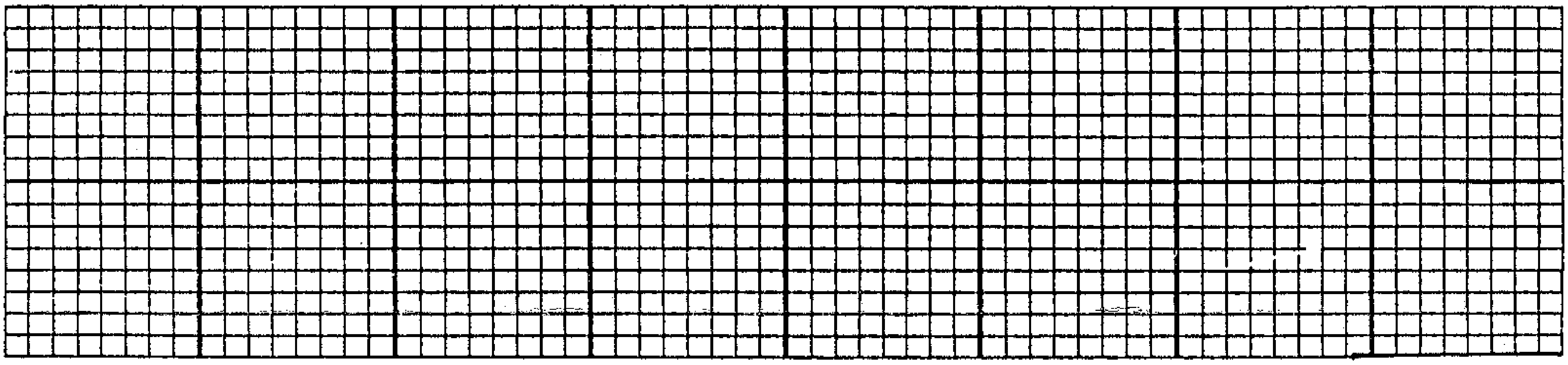
Another aspect of computer education that concerns Wessells is, "the social context of computers in education," as he puts it. Consider gender inequities, for example. At early ages (up to third grade or so), boys and girls use computers about equally. Then a divergence appears: boys start using computers more, while girls use computers less. This divergence reflects the emphasis and expectations of a culture that treats computers as as a masculine domain, Wessells says.

Class inequities are a problem too, Wessells goes on. Using sixth-graders as a case in point, Wessells says that computer use at school is proportional to computer use at home - so that children from middle-class and upper-class households enter the computing environment at school with an advantage over lower-class children who have less opportunity to use computers at home.

Schools in areas where lower-class children predominate, find that students use computers less frequently and in less productive ways than in schools serving more alluent communities. This is one way computers "can be used to reinforce the dominant social order," Wessells says.

This last observation of Wessells is precisely why I loan computers to my computer students whose families cannot afford their own. I credit the ERIE 99'ers User Group for making this possible.

Need Graph Paper In A Hurry? Here's A Program To Do I



```
100 ! GRAPH
110 E$=CHR$(27)
120 A$=RPT$(CHR$(128),228)
130 B$=RPT$(CHR$(255)&SEG$(A$,1,6),8)
140 B$=RPT$(B$&CHR$(255),4)
150 A$=E$&"K"&CHR$(228)&CHR$(0)&A$
160 B$=E$&"K"&CHR$(228)&CHR$(0)&B$
170 OPEN #1:"PIO.CR"
180 FOR I=1 TO 11
190 PRINT #1:E$;"@";E$;"3";CHR$(24)
200 FOR J=1 TO 8
210 PRINT #1:B$;B$;CHR$(10)
220 NEXT J
230 PRINT #1:A$;A$;E$;"3";CHR$(2)
240 NEXT I
250 PRINT #1:RPT$(CHR$(13)&CHR$(10),9)
260 PRINT #1:E$;"@"
270 CLOSE #1
```

ARCHIVING—A HEADACHE?

By: Andy Frueh, Lima UG

A lot of people are puzzled by archiving and how to use Barry Boone's Archiver. What follows is both a reference guide and explanation of Archiver III. It is not meant to totally replace the documentation for this program. Actually, I haven't seen a distribution copy that comes with a set of instructions. There may be hidden features of ArcIII that aren't obvious to me (for example, Disk Utilities by John Birdwell has a feature to figure decimal-to-hex conversions).

What exactly is archiving? Putting it simply, when you archive you take file or a set of files, and group them as one file then compress them so they take up less disk space. Some software comes archived. These ALMOST always include the archiving program. Examples are Jack Sughrue's PLUS! and the Complete Adventure disk set.

What is the purpose of archiving? Well it started out as a money saver for modem users. It is faster, and thus cheaper, to send 90 archived sectors as 1 file, than 120 sectors for 3 programs. Now it is also a means of backing up disks. You can save each of your disks as a one file, squashed archive. You can specify whether you want compressed files or not. The reason you have a choice is that some unusual files actually take up more space when they are compressed. Another useful application of archiving is when you have programs you want to keep, but don't need ready to use. You can keep archives of all these files instead of taking up disk space.

OK, now that you have the "what", here's the "how". As far as I know, the only archiver is Barry Boone's program. Its operation is completely different from Archiver II. Rather than add new features to past versions, Archiver was completely re-written. It usually contains an XB LOAD program, but may be loaded from E/A. The program's filename is usually ARC1. It can be found on almost all of the bulletin boards, as a commercial version with Geneve utilities, in user group libraries, with other Fairware programs or from the author. Chances are, you can definitely get a copy.

First things first, so get the program loaded. After that, you should see a Fairware notice. Press any key to pass this. You then see a menu. Each menu option is described in detail below.

1) Archive Files - These options are largely self-explanatory. As you may have guessed, this option archives files. Pressing one will deliver a set of prompts. These are "Source Drive (1-Z)". Yes, you can have drive numbered from 1-9 and A-Z. Then comes, "Output Drive (1-Z)". You may use one drive. Archiver will prompt you to change disks when needed. It is highly recommended that you use a blank output disk, since archives may fill or almost fill a disk. Next comes "Output Filename". This is usually the name of the disk you are archiving, or some related heading. For example, a set of D/V 80 articles may be named "ARTICLES". The following prompt is "Pack all Files? (Y/N)". If you answer "Y" then all the files on the source disk are archived. If you answer "N", then when Archiver is working, you are asked "Include filename? (Y/N)" If you answer "Y" then that file is archived, otherwise it is ignored This is a handy feature if you have programs and files for example, and need them seperated. This process repeats for each of the files on the source disk The final prompt is "Compress? (Y/N)". Saying "Y" and Archiver attempts to squash each file so it takes up less space. Remember that some unusual file types will actually get LARGER if compression is attempted. When all the prompts are answered, press REDO to correct an error in your answers, BACK to return to the menu, or any other key to continue. When Archiver is done performing any operation, pressing a key goes back to the main menu.

2) Extract Files - This is the opposite of archiving. It will let you pull (extract) files from an ARC file. You are first asked for the source drive. Next you input the source filename. After that, you are asked for the output drive. It must be stressed that the output drive for ALL operations of Archiver should be different than the input drive. You may run out of space or overwrite a file accidently. Output disks should be blank.

The next prompt asks, "Extract all files?" If you answer "Y" then every

file stored in the ARC file will be taken out. If you answer "N" then when extracting starts, the program asks, "Include filename?" for every separate file in the archive. Again, press REDO (to restart this option), BACK (return to main menu), or any other key to continue.

3) Catalog Disk - This is fairly self explanatory. Simply input the source drive name. The program will ask if you want a printout. If you answer yes, then you are asked for the printer name. If there are more files than can be displayed, then [more] is printed on the screen and pressing a key advances the screen.

4) Catalog ARC File - If you aren't sure what files are contained in an archive file, than this option tells you. You are asked for the source drive, source filename, and whether or not you want a printout of the list of files.

5) File Copy - This option will copy a file (obviously). Simply supply the source drive and filename, and the output drive and filename.

6) File Rename - Again, this option should explain itself. Give the source drive and filename, then the output filename.

7) File Delete - Supply the source drive and filename.

8) File Un/Protect - You first supply the source drive and filename. You are then asked "Protect?" If you answer "Y" the file is protected. Otherwise file protection is lifted.

9) List Text File - This will display or print a D/V 80 file. Give the source drive and filename. You are then asked if you want the file printed or not.

10) Load FW - This returns to Funnelweb. Simply give the drive number on which the UTIL1 file is located.

NOTE: When an I/O error occurs, pressing a key returns to the main menu. If you have a Geneve, this is for you. Using a sector editor, find the string 04E08C00 and replace it with D8018C00.

I think that this should get people on the road to understanding archiver Remember that it is fairware, so if you find it very useful, send the author (Barry Boone) a donation.

[This article/item comes from the January 1991 issue of BITS, BYTES PIXELS (Charles Good, editor), the newsletter of the Lima OH 99/4A User Group, P.O. Box 647, Venedocia, OH 45894.]

NEXT MEETING TUESDAY, JANUARY 14, 1992   HAPPY NEW YEAR!!!

MUNCH OFFICERS AND NUMBERS (all in 508 area unless noted)

President	W.C. Wyman	865/9683
Vice President	Bruce Willard	852/3250
Secretary	Jim Cox	
Treasurer	Jim Cox	869-2704
Acting Editor	Jim Cox	
Adv.Prog. Chair	Dan Rogers	248-5502
Library	OPEN	
Disk Librarian	Lou Holmes	617 965/3584
Tape Librarian	Walter Nowak	413 436/7675
NEW-AGE/99	Jack Sughrue	476/7630

MUNCH DUES	
NEW MEMBERSHIP	$25.00
RENEWAL MEMBERSHIP	$15.00
NEWSLETTER ONLY SUBSCRIPTION	$12.50

DECEMBER MEETING. We had 15 members present at the meeting. Jack wasn't feeling well so he just said a quick hello aand he left. Lou was able to fill in and did the demo. He even got all of the equipment to work! Ben Parda won the raffle.

JANUARY MEETING. This month will feature Jack doing a surprise demo, i.e. he didn't tell me what it was. I'm sure it will be of interest to us all. We haave decided to postpone the video taping until we find a better location with more heat and light. If you know of anyplace we could use let us know.

RAFFLE. Every month we have a raffle to help defer the rental cost of our meeting hall. A typical raffle will have game and utility programs T-Shirts, books, bumper stickers, blank discs and all sorts of odds and ends for the T.I.

LIBRARY NOTICE. Please return any items borrowed from our library. If you can not come to a meeting or give these items to someone who will be at the meeting.

REPRINTS. Reprints are permitted as long as credit is given to M.U.N.C.H.

ARTICLES. I am always looking for articles for this newsletter, anything which interests you will probably interest other members of the TI community, so please share your ideas and opinions with all of us.

DISK LIBRARY. The disk library will be at the meetings from now on. We have copies of all disks in the library and they are available to members for just $1.50 each for single discs, $2.00 flippies, $3.00 double discs and $4.00 double flippie.

FOR SALE. The group has a TI Count Business Software package available for sale. If interested contact Jim Cox at the above numer or the club address.

DISK OF THE MONTH. This month's disk is Infocom Adventure Hints. Thiss flippie contains hints and maps for all Infocom games. A must for the Infoccom gamer. This flippie costs $2.00. We haave a new update of Funnelweb 4.40+. If you want a copy bring two formatted flippies to the meeting and make a copy of them. I will try to have some copied so you can exchange your disks for ours. If you want to purchase them they will cost $4.00.

The B.C.S. has announced the date for the 1992 T.I. Fayah, it will be on Saturday April 4th at the Waltham Junior High School. Let's start to make plans now for the Fayah.

!

Mass Users of the Ninety-nine and Computer Hobbyists

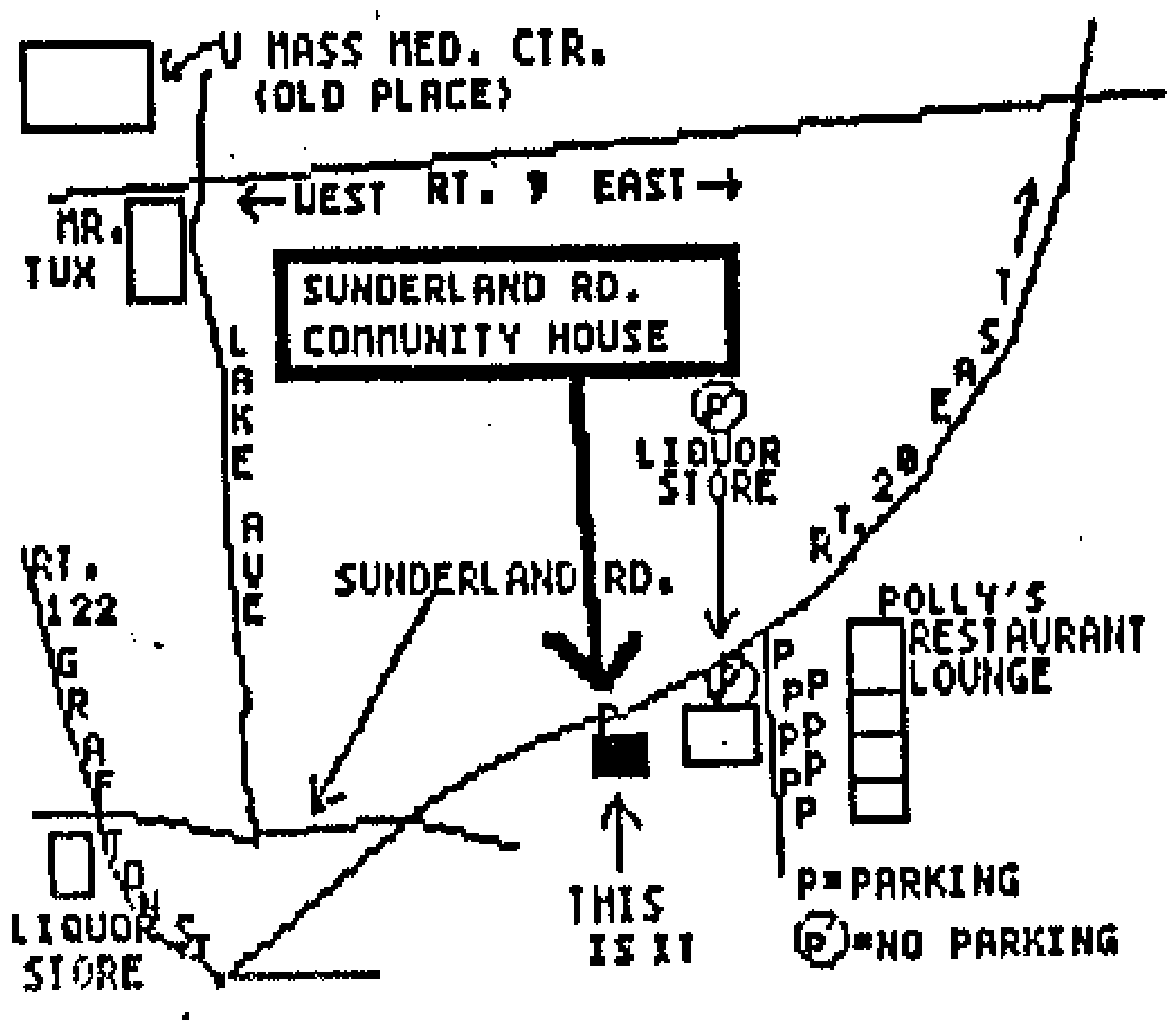
JANUARY 1992 Monthly Newsletter Version 11.01

M.U.N.C.H.
560 LINCOLN ST.
P.O. BOX 7193
WORCESTER, MA. 01605-7193

FIRST CLASS

Next Meeting JANUARY 14th

POSTMASTER: Forwarding and Address Correction Requested.


U MASS MED. CTR.
(OLD PLACE)
MR. TUX
←WEST RT. 9 EAST→
SUNDERLAND RD.
COMMUNITY HOUSE
LAKE AVE
LIQUOR STORE
RT. 20 EAST
SUNDERLAND RD.
RT. 122
GRAFTON ST
POLLY'S RESTAURANT LOUNGE
LIQUOR STORE
THIS IS IT
P=PARKING
Ⓟ=NO PARKING