MSYS

INTRODUCTION

MSYS is a Copyrighted program that I am making available to the amateur radio community free of charge. Source code is not generally available (if you want it you will have to pay dearly for it!).

KA-NODE and GTOR are trademarks of Kantronics Inc. The KANODe component of MSYS is compatible with the Kantronics KA-NODe but is completely different code from that used in the Kantronics tncs.

NET/ROM is a trademark of SOFTWARE 2000. The network NODe component of MSYS is compatible with NET/ROMs but is completely different code from that sold by SOFTWARE 2000.

The network NODe component of MSYS is completely different from that in the KA9Q-NOS program and G8BPQ Node software.

IBM is a trademark of International Business Machines Corp.

SAM is a trademark of RT Systems

j-comm is a trademark of j-comm Inc.

MSYS is available from the Cleveland Hamnet phone BBS. The number is 216-942-6382. It's speed is 1200 to 14,400 baud. This is a free BBS (you pay for the call). MSYS may also appear on other phone BBS systems, but this is the only one I put it on directly. MSYS may also be obtained by anonymous ftp on Internet from coffman.csuohio.edu [137.148.21.6].

You may also obtain MSYS from me directly on disk. Distribution is on 5.25 inch 360K formatted disks. Costs per disk which includes the disk, mailer and postage are as follows:

U.S.	$5.00 per disk
Canada & Mexico	US$7.50 per disk
All Others	US$10.00 per disk

The number of disks may vary from release to release. Recent releases have been one disk to three disks. Four disks may be needed in the future. You may order several versions at a time, I will hold your request until the versions are released. This way you get to be the first kid on the block with the new toy! When ordering from me, please specify which version you already have or which one you want.. I keep records with the following information for people who order MSYS, so you may want to give me this info. Also please indicate if you want me to call you (collect) if it would be helpful to answer questions you send me about MSYS.

```
CALL:          Your callsign.  This is the key I use to record all
your                information.
NAME:          Where to send the disk to!
ADDRESS LINE 1:
ADDRESS LINE 2:(if needed)
CITY:
STATE:
ZIP:
COUNTRY:
TELEPHONE:     If you want me to call you back (collect)
VERSION:       The version you have now
OWED:          How many you have prepaid
COMPUTER:      Speed, memory, disk space, CPU chip etc.
TNC:           Model & ROM versions
DOS:           Which version?
BBS:           BBS callsign to send you msgs .
IP:            Your IP address, if you have one
```

Mail requests for MSYS on disk to:
Michael Pechura
10809 Beechwood Drive
Kirtland, OH 44094

You may call me with questions/comments at 216-256-1588.

A stripped down version of MSYS that supports only the BBS functions is also available. There is no network NODe, no amtor, no tcp/ip in this version. You can still forward using external network nodes, but the bbs is NOT a netrom network node! The KANODe is still there. All of the BBS functions remain, along with modem support. Expanded memory is also still supported. This version is called MSYSB and is available on a separate disk or on Cleveland Hamnet. If want to get this disk from me, it is the same price as the others: $5 to US addresses, $7.50 to Canada and Mexico, and $10 to other countries.

Internet Group being Formed

This deals with a Internet network mailing list. You have to have access to Internet mail to participate. For those of you who have not experienced a mail list, it is a program that runs on a computer connected to the Internet. Anyone posting a message to a special address on that computer has that message echoed to all members on a mail list. It is, essentially, a newsgroup limited to those who wish to subscribe.

A new mail list has been created to help with communications between MSYS sysops. It has been created by the same people who brought you the Cleveland Hamnet BBS which is the primary host for MSYS. A number of active MSYS sysops will insure that questions get reviewed and suggestions are made.

ADMINISTRATIVE MESSAGES

You must send a message to a different address than the message which is used to post items to the group. Please, if you subscribe, save these addresses so you can modify your subscription later. PLEASE don't send subscribe and unsubscribe messages to the posting address.

To Subscribe:

```
msys-request@hamnet.wariat.org
<subject> subscribe
no body text
```

You will get a ack when your subscription is registered.

To Unsubscribe:

```
msys-request@hamnet.wariat.org
<subject> unsubscribe
no body text
```

To get help:

```
msys-request@hamnet.wariat.org
<subject> help
no body text
```

POSTING ADDRESS:

msys@hamnet.wariat.org

PROBLEMS:

no8m@hamnet.wariat.org

Thanks to Dave, WB8APD, for setting this up!

A Multi-User, Multi-Port, Multi-Function

Amateur Packet Radio BBS

By Mike Pechura, WA8BXN

Version 1.18 NOVEMBER 1994

MSYS Documentation Table of Contents

MSYS INITIAL INSTALLATION

1. Installation of MSYS is not particularly difficult but you should follow the following steps in order.

2. If you are upgrading from MSYS 1.17, please see the UPDATE Instructions.

3. Make a backup copy of the distribution disk. You can copy the entire disk with Diskcopy.

4. Make a default directory on your hard disk. I call mine MSYS. Use the command:
   ```
   md c:\msys
   ```

5. NOTE: You may use any drive letter. All references to the C: drive apply to your selected drive.

6. Go to the default directory: cd c:\msys.

7. Make required sub directories:
   ```
   md c:\msys\mail
   md c:\msys\files
   md c:\msys\help
   md c:\msys\mail.bak
   md c:\msys\mail.bin
   ```

8. Move the file MSYS118.exe into the c:\MSYS directory

9. Un-arc the distribution disk by running the program MSYS118.EXE.

10. To move the help files to the help sub-directory. Type MOVEFILE

11. NOTE: REQUIRED FILE: C:\MSYS\HELP\MSYSMSGS.DAT
    This file contains a number of the messages used by MSYS.

12. Change Directory to SAMPLES. Edit the file C:\MSYS\SAMPLES\MSYS.OPT with your favorite program editor. See documentation on MSYS.OPT for details. After editing move MSYS.OPT file to the default directory.

13. RUN MUTIL.EXE Function 9 to create C:\MSYS\MSYS.DEF from your edited MSYS.OPT

14. Make sure you have at least:

```
files=40
buffers=20
```

in your config.sys file in the root directory of your boot disk. Don't forget to reboot after changing this!

15. Using any ASCII editor, create MSYSFWD.DEF. If you don't know what to put it in now, put the following (to be replaced later):

```
F0 WA8BXN
WA8BXN
-----
```

16. Type MSYS and it should begin to run!

17. Type MERge. You can ignore any messages about MSYSBBSB.DAT at this time.

SPECIAL NOTE
If you have a color monitor or any kind of a color card, the default is OK. If you have a monochrome monitor that uses and RCA phono pin plug then you are using a Color Graphics Adapter interface card and the default should be OK. If you can do the command MODE MONO (this is a DOS command) and not get any error messages and the computer continues to work, you probably have a monochrome (TTL) interface card. You will need to specify WINDOW DISPLAY MONOCHROME in the MSYS.OPT file.

MSYS UPDATE INSTALLATION INSTRUCTIONS

SETUP CHANGES

There are two new directories that are required to update MSYS Ver. 1.13 to Ver. 1.18 One is the MSYS\MAIL.BIN directory that is used to store the compressed messages as they are received and sent, and the other is MSYS\YAPP which is used to store files that have been transferred via YAPP compressed protocol. Please see the MSYS DIRECTORY LAYOUT DESCRIPTION for further details. Use DOS commands to make the new sub-directories. If you are already running MSYS 1.15, no further changes need to be made in the directory structure.

Increase your FILES statement in CONFIG.SYS to files=40. BUFFERS can remain at 20.

It is suggested that a program similar to the Quarterdeck Systems: VIDRAM ON be run prior to running MSYS to create as much working memory as possible. MSYS does not require graphics capability. Of course, if other background applications require graphics, this option is not available. VIDRAM works with EGA or VGA video adapters, and should not be loaded into upper memory.

Run MUTIL 9 (setup) and of course, MUTIL 4 before restarting the new version of MSYS.

ADDITIONS & CHANGES

There have been many changes to MSYS, some major and many minor. Here are some of the changes:

1. One of the major changes in Ver. 1.18 is the addition of G-TOR. MSYS properly interfaces a KAM Plus that has the proper firmware installed.
2. "Attended" G-TOR forwarding on HF can be initiated from the keyboard at the same time VHF forwarding is in progress.
3. Scanning of HF radios is supported in the G-TOR mode.
4. A new command SHUtdown has been added. When on, it prevents new incoming connects, and advises current users of impending shut down.
5. A new command VAlidmsgtypes which takes a string of letters as its arguments that may follow S in send commands in the bbs. Defaults are BTPW. This should prevent type "SR" and related problems.
6. A SE- chan# text command has been added which will send the given text without the usual "Msg from sysop" prefix.
7. Regenerated messages are now stored in the mail.bak directory.
8. The CONFERENCE is no longer accessible from the modem.
9. The bug in MAXBullage has been resolved.
10. The LU RESET bug has been fixed.
11. The sysop commands K< and K> have been implemented.
12. The "TO" call, the "FROM" call and the "@BBS" are now also compared with the bad word file if it is used.
13. Changes have been made in the forwarding syntax to partially accommodate the misguided useage of specific hierarchical components as flood routes, such as USA instead of ALLUS
14. If a message disappears during forwarding, (from housecleaning or manual deletion), the forwarding connection will now be terminated.
15. If a message is refused by a FBB station with the FS = response, the message will be held for sysop evaluation.
16. Bulletins that have the same titles as active bulletins from the same callsign will be held for sysop review for duplication.
17. Bulletins that have an apparent BID change based on the R: lines will be held for sysop review for duplication.
18. Users will have the opportunity to delete a replied to message.
19. Sysops may kill any read message as well as reply to them with a "k" response if autoreply is enabled.
20. MSYS will do an automatic MERGE at the initial startup. It will not do a merge if forwarding is in progress, but will initiate one at the beginning of the next forward cycle.
21. A command RTime has been added to enable the proper reporting of "Z" time in the R: line while running local time in the bbs.
22. Restored messages are now also stored in the mail.bak directory.
23. MUTIL 7 has been added to convert JNOS-style domain.txt files to MSYSHOST.NET format. The file will still need editing to add aliases, but is does save a lot of retyping.

KNOWN PROBLEMS WITH THIS RELEASE

The author is aware of several problems with this release that may be addressed in future releases as time permits.

One concerns FTP sessions between JNOS and MSYS. The sessions do not conclude properly. The temporary fix is for you to manually disconnect the data channel session.

A second problem can occur when a message has come into a MSYS system with "garbage" characters, generally via a FBB or a telephone modem forwarding session, and if the first and only character of a line happens to be a control A. If that message is then forwarded out using RLI protocol, the forward session will abort. The obvious cure is for sysops to review, and edit as necessary, messages that contain extraneous characters. FBB-style forwarding generally is not affected by this problem.

If you find a problem not listed above, please document as carefully as possible, outlining all the conditions that can cause the problem before advising the author. Of course, the condition must be repeatable for it to be found. Messages that say: "It doesn't work" are not helpful. Likewise, if you find errors in the documentation, please advise so that proper changes can be made. Thankyou for your cooperation.

MSYS COMMAND LINE

The MSYS command typed at the DOS prompt may have several switches or modifiers.

NOHEARD	Disables JK,JM,JD,JG,JB and JT recording. Saves 26,880 bytes
NONODE	Disables network node, saves 28,104 bytes
NORAMHDRS	Stores message headers on disk only, saves 136 bytes per possible message. This can be quite slow if you have lots of msgs and no disk cache. Could be useful on 286 systems with cache in EXTENDED memory.
NOEMM	Disables Expanded Memory.
NOEXMEM	Disables Extended Memory.

Example:

MSYS NOHEARD	Starts MSYS but will not log any stations it hears.
MSYS NONODE	Starts MSYS without the NETNODe feature.
MSYS NORAMHDRS	Starts MSYS and stores all headers on disk - not in RAM
MSYS NOEMM	Starts MSYS and forces MSYS to use EXTENDED memory (if any).
MSYS	Starts MSYS, permitting it to use EXPANDED memory, (if any) and all other features that have been optioned.

MSYS has the ability to determine if there is an EXTENDED or EXPANDED memory manager in use. It will attempt to use the EXPANDED memory unless the NOEMM option is specified, in which case it will attempt to use the EXTENDED memory. MSYS now requires about 2 MBytes of upper memory.

While MSYS may be run on an 8088 or 8086 system, much better performance will be achieved by using a 286 or higher system.

ABBREVIATIONS/CONVENTIONS

Just a word about this manual. Most commands are not case sensitive, with the notable exception of the \Sysop command. Many commands may be abbreviated. In this manual, we have attempted to show the minimum abbreviation with the UPPER CASE letters with the balance of the command word in lower case. For example, the command ADDUser may be typed as: addu or ADDU or adduser or ADDUSER with the same results. Where there is an argument shown as [on/OFF], the UPPER CASE will be the default. For example, in the case of the command PC [on/OFF], the call server lookup defaults to OFF.

MSYS DIRECTORY LAYOUT DESCRIPTION

The following chart is the recommended organization for your directories and files. NOTE: Some of these files are created at time of installation and others are operating files that MSYS creates while operating.

```
C:\MSYS   Contains all of the MSYS operating programs and
  |       configuration files
  |  MSYS.EXE $     MSYS.MSG #      BIDLIST.DAT #  MSYSPASS.IP @
  |  MUTIL.EXE $    MSYS.BBS #      MSYSPASS.MOD @ MSYSPASS.DAT @
  |  ROUTES.DAT $   MSYS.DEF #      MSYSBBSB.DAT @ MSYS.HCL @
  |  STATES.DAT $   MSYS.FWD #      MSYSTODO.DAT @ MSYSHCL.DO @
  |  ZIPCODES.DAT $ MSYS.HRD #      MSYSHOLD.DAT @ MSYS.OPT @
  |  CALLSRV3.EXE $ MSYS.LOG #      MSYSFWD.DEF @  MSYS.OTD @
  |  MSYS.BBS #     MSYS.USR #      MSYSHOST.NET @ MSYS.REP @
  |  BBSTONTS.BIN # MSYSNODE.DAT #  BADWORDS.DAT @ MSYS.RMT @
  |  BBSTONTS.DAT # MSGLIST.DAT #   MSYSTNC.1 @    MSYS.DO @
  |       Files marked with a: $ are Distribution files.
  |                             # are created by MUTIL or MSYS
  |                             @ are SYSOP created files
  |
  |--------FWD
  |   Directory which may contain forwarding files (.FWD)
  |   the reverse forwarding  files (.REV) and the
  |   PASSWORD.RMT file and is used by MERge command
  |
  |--------HELP
  |   Directory which contains HELP messages including
  |   MSYSMSGS.DAT
  |
  |--------MAIL
  |   Directory which contains ACTIVE messages
  |
  |--------MAIL.BAK ***Optional
  |   Directory which may contain BACKUP messages  See ADVANCED
  |   USER section for suggestion!
  |
  |
  |
  |       (continued next page)
  |
```

```
|
|       (continued from previous page)
|
|--------MAIL.BIN
|   Directory which contains COMPRESSED messages
|
|--------PASSWORD ***Optional
|   Directory which may contain user PASSWORD files
|
|--------PUBLIC   ***Optional
|   Directory which may contain uploaded FTP/SMTP messages
|
|--------YAPP     ***Optional
|   Directory which may contain YAPP files
```

Before MSYS can be used a configuration file (MSYS.OPT) must be created and the program SETUP (MUTIL.EXE Function 9) must be run to create the file MSYS.DEF. The file MSYS.OPT is a standard ASCII file and may be created with any text editor. The basic format of the commands in the file are:

```
PORT number AT number INT number SPEED number [NAME description]
PORT number SUBPORT number OF [PORT] number [NAME description]
DIGI callsign FROM [PORT] number TO [PORT] number [AS callsign]
NUMCHANS number
BBS CALL callsign
BBS ALIAS callsign
KANODe CALL callsign
ANSWER CALL callsign
FORWARD CALL callsign
ID CALL callsign
ID EVERY number
AX25 FRACK number
AX25 MAXFRAMES number
AX25 RETRIES number
AX25 CHECK number
MONITOR IFRAMES on/off
MONITOR UFRAMES on/off
MONITOR SFRAMES on/off
MONITOR BFRAMES on/off
MONITOR OUTGOING on/off
MONITOR PORTS [MASK]
WINDOW number SIZE number number number number
WINDOW number BACKGROUND number
WINDOW number FOREGROUND number
WINDOW number PAGE number
WINDOW UNUSED BACKGROUND number
WINDOW UNUSED FOREGROUND number
WINDOW DISPLAY color/monochrome
PROCESS processname [WINDOW] number
STACK processname number ***NOT USED IF USING EXT/EXP MEMORY
                          (SEE TEXT)
SYSOP NAME name
SYSOP QTH location
SYSOP ZIP zip code
MAKE
--------------------------------
```

Notes:

1. Items enclosed in square brackets ([...]) are optional
2. Numbers assumed in decimal unless prefixed by $ then hex is assumed
3. on/off means the word ON or OFF
4. Where more than 1 space is shown, 1 or more space are OK
5. Callsign can include SSID (-number)
6. Ordering of the statements is generally not important
7. If there is an * at the beginning of a line it is ignored
8. The BBS alias, KANODe, Answer and Forward calls must be different calls or SSID's.
9. MASK is described on page 24

Output from the execution of SETUP is the file MSYS.DEF (a binary file) and a listing to the display console. The listing consists of two parts: the original input statements followed by a complete list of all options generated (including defaults).

If any errors are detected, the MSYS.DEF file will not be created.

A sample of a MSYS.OPT file is found in MSYS.OPT on the distribution disk.

A minimal MSYS.OPT file will have similar statements to these:

```
port 0 at $3F8 int 4 speed 1200 name 145.01
bbs call wa8bxn-13
answer  call wa8bxn-14
sysop name Mike
sysop qth Kirtland, OH
sysop zip 44094
id call wa8bxn
make
```

EXPLANATION OF EACH STATEMENT TYPE

```
PORT number at number INT number SPEED number [NAME Description]
[RADIOSPEED #][TNCtype#]   (ALL ON ONE LINE!)
PORT number SUBPORT number OF [PORT] number [NAME description]
```

The PORT statement has the above two variations. The number following the word PORT is the port number. It must be between 0 and 6. The version containing the word 'at' defines an actual serial interface, while the SUBPORT version defines a second logical port for multiport TNCs such as the KPC-4. There must be one PORT statement for each RADIO port to be used AND each MODEM port to be used.

Assign port numbers beginning with 0 in increments of 1.

Following the word AT is the base I/O address of the serial port, such as $3F8. Following the word INT is the interrupt number for the serial port, such as 4.

After SPEED is the baud rate to be used, such as 4800. This is the speed between the computer and the TNC or the MODEM. It is STRONGLY suggested that with faster modems that a value of 19200 be used to prevent buffer overruns. Some adjustment of this speed may be necessary to avoid overruns on a per port basis.

Optionally the word NAME followed by some description of the port can be given. A good description would be the corresponding radio frequency.

If MSYS is interfacing a NET/ROM matrix, or a TNC using the NetRom Serial protocol (NRS), precede the NAME with a ">" as in: >MATRIX

If MSYS is interfacing a modem, the NAME must be: MODEM

If MSYS is interfacing a PACTOR or GTOR tnc, the NAME must begin with: PACTOR or GTOR. You may choose to name it PACTOR_20M or GTOR_20M for better description. The name must not exceed 10 Characters total.

For the other form of the PORT statement, the number following the word SUBPORT is the logical channel number, typically 1 for the KPC-4. Next comes the word OF in the statement optionally followed by the word PORT. Next comes the port number of a port statement defining an actual port (i.e., contains the word AT in its definition). The optional NAME and description is the same as above.

PORT description lines can also have two additional parameters:

RADIOSPEED bitrate# TNCTYPE #

RADIOSPEED specifies the bitrate used by the TNC to the radio (300 or 1200 probably). If 300 bps is selected, shortened prompts are used in the BBS and perhaps elsewhere.

TNCTYPE is a number from 0 to 999. TNC type 0 is a KPC-x or a KAM. Any other number causes a file MSYSTNC.# to be read for initialization statements to be sent to the TNC. These parameters would be used for a TNC other than a KPC-X or a KAM, or for the file to initialize a modem..

NOTE: You must specify RADIOSPEED if you use TNCTYPE#

For example:

PORT 2 AT $2F8 INT3 SPEED 4800 NAME 145.01 RADIOSPEED 1200 TNCTYPE 1

IN THE MSYSTNC.1 FILE YOU WOULD HAVE THE COMMANDS THAT WOULD PUT YOUR TNC INTO KISS MODE. (see note in general info section)

DIGI callsign FROM [PORT] number TO [PORT] number [AS callsign]

The DIGI statement defines a digipeater callsign. Repeat this statement type up to a total of 25 times as desired. The first callsign is the callsign that would be used in the VIA list by a station trying to use the digipeater. The port number following FROM indicates the port that will be using this call. The TO port number indicated the port on which packets having this the first callsign as digipeater should be output on. For a normal digipeater the two port numbers will be the same; for a gateway they will be different. The optional AS callsign specifies a replacement for the first callsign in the digipeater list when the packet is sent out the TO port. If omitted the first callsign is assumed (i.e., no change). Consider the following examples:

```
DIGI MYDIGI FROM 0 TO 0
DIGI MYDIGI FROM 1 TO 1
DIGI W1XX-3 FROM 0 to 0
```

This would allow the callsign MYDIGI to be used in the via list on either port 0 or 1. The packet is retransmitted on the same port was heard on. The call W1XX-3 can also be used to digipeat on port 0.

```
DIGI MYGATE FROM 0 TO 1
DIGI MYGATE FROM 1 TO 0
```

This pair would define a gateway between ports 0 and 1 with the callsign MYGATE used in both directions.

```
DIGI TOF1 FROM 0 TO 1 AS TOF2
DIGI TOF2 FROM 1 TO 0 AS TOF1
```

This interesting pair allows TOF1 to be used as a gateway digipeater call on port 0 (no effect if used on port 1). When the packet is repeated it will have TOF2 in the digipeater list where TOF1 originally was. The opposite is true on port 1: When TOF2 is used as a digipeater call the packet is repeated on port 0 with TOF1 being substituted for TOF2. The substitution is transparent to the users.

NUMCHANS number

NUMCHANS is used to specify the number of logical channels that will be allocated. Each connection (while connected) uses one logical channel. When the KANODe is being used a second logical channel is used for outgoing connects. Each logical channel requires about 7K of memory when in use. They are available for reuse when a station disconnects.

BBS CALL callsign

This statement gives the callsign that can be connected to for the BBS. The callsign can contain an SSID. This statement is also optional; if omitted stations will not be able to connect to the BBS unless a BBS alias is given.

BBS ALIAS callsign

This statement gives the callsign that can be connected to for the BBS. The callsign can contain an SSID. This statement is also optional; if omitted stations will not be able to connect to the BBS unless a BBS CALL is given.

KANODE CALL callsign

This statement gives the callsign of the KANODe which is similar to the KANODe provided in the Kantronics tncs. The callsign can (and probably should) contain an SSID. This statement is optional; if omitted stations will not be able to connect to the KANODe.

ANSWER CALL callsign

This statement gives the callsign that can be connected to by stations wanting to talk to the SYSOP. It is also used as the callsign when the Connect command is used. The callsign can (and maybe should) contain an SSID. This statement should not be omitted.

FORWARD CALL callsign

This statement gives the callsign used when the BBS is forwarding. It can (and probably should) contain and SSID. This statement should not be omitted if the BBS is to do forwarding.

ID CALL callsign

This statement gives the callsign that will be used for identification purposes. It must be your actual amateur callsign and should NOT have an SSID specified. This statement MUST be used.

ID EVERY number

This statement specifies the number of minutes between IDs. The list of stations for which there is mail is generated with every ID.

AX25 FRACK number

Number is the number of seconds to wait before retransmitting a packet. All of the AX25 statement forms are optional.

AX25 MAXFRAMES number

Number specifies the maximum number of frames that can be outstanding before waiting for an ACK. Must be between 1 and 7.

AX25 RETRIES number

This is the number of retries that will be made before abandoning a connection.

AX25 CHECK number

If there is no activity on a logical channel for this number of seconds there will be an automatic disconnect.

MONITOR IFRAMES on/off

Turns on or off the displaying of Information frames (SABM, DISC, UA). All forms of the MONITOR statement are optional.

MONITOR SFRAMES on/off

Turns on or off the displaying of System (RR, RNR, REJ) frames.

MONITOR BFRAMES on/off

Turns on or off the displaying of Beacon frames (Un-numbered information).

MONITOR OUTGOING on/off

Turns on or off the displaying of all outgoing packets.

MONITOR PORTS [MASK]

Determines which ports will be monitored. The rightmost bit of this number corresponds to port 0, the next bit to the left port 1 and so on. Ports with 1 bits are displayed (as selected above). A value of 13 (decimal, $D in hex) would enable ports 0, 2 and 3, for example.

SEE MASK on page 20 for further details.

WINDOW number SIZE number number number number

This form of the window statement gives the bounds of the window whose number is given after the word WINDOW. The four numbers after the word SIZE are the upper left column (1 to 80), the upper left row (1 to 25), the lower right column, and the lower right row respectively. All of the WINDOW statement forms are optional and should be used with great caution.

WINDOW number BACKGROUND number
WINDOW number FOREGROUND number

These two forms of the WINDOW statement give the background and foreground colors of the window specified after the word WINDOW. The rightmost number is a color number from 0 to 15 for foregrounds and 0 to 7 for backgrounds. The corresponding colors are:

Black - 0	Magenta - 5	Lt. Cyan - 11
Blue - 1	Brown - 6	Lt. Red - 12
Green - 2	Lt. Gray - 7	Lt. Mag - 13
Cyan - 3	Gray - 8	Yellow - 14
Red - 4	Lt. Blue - 9	White - 15
	Lt. Green - 10	

```
  WINDOW number PAGE number
```

This statement assigns a window to the specified page (0-3) which corresponds to the F1, F2, F3, and F4 keys.

```
  WINDOW UNUSED BACKGROUND number
  WINDOW UNUSED FOREGROUND number
```

These two statements give background and foreground colors to the places on the screen not assigned to any window.

```
  WINDOW DISPLAY  color/monochrome
```

This statement selects the kind of display adapter you have, COLOR or MONOCHROME. If you have a CGA card, select COLOR.

```
  PROCESS processname [WINDOW] number
```

This statement specifies the window that is to be used for output from the allowable process names . Be extremely careful if you use this statement.

```
  STACK processname number
```

This statement specifies the number of bytes to be allocated for the stack for the allowable process names . You probably shouldn't mess with the defaults! ***THIS OPTION IS IGNORED ON A 286/386 equipped with EMM or EMS.

DISCUSSION:

First, if using extended or expanded memory, you have no control over stack size, they are all 16K. Only when just 640K (or less if that is possible) regular memory alone is used can adjustments be made to stack sizes. Normally these should not be necessary. The default values set by MUTIL 9 should be correct. This means that normally there should be no process stack statements in MSYS.OPT.

How to tell when you need to adjust stack sizes --- If you get unexpected crashes that can't be solved by other means, do somewhat frequent process commands when conditions are similar to when crashes occur (like when lots of channels are in use, when forwarding is happening, etc.). If the SPmin value for any process gets below a few hundred an increase in the stack allocation for that kind of process may be needed. In the event SPmin goes to 0 or negative, an adjustment is required.

How to adjust stack sizes --- Get a printed copy of the output from MUTIL 9. You will see there a list of the process type names and current stack sizes for each. Add a process stack line in MSYS.OPT for the one you need to increase. Any increases should probably be done in 1K increments.

Stack sizes should probably not be adjust downward greatly from the default values even though it appears on most process command lists that the SPmin is always very high. How much stack space is needed is dependent on the particular activities that are occurring. Values that appear excessive may be required for infrequently performed operations that if done with small stacks would result in unpredictable results. (Can you say "CRASH"? How about "HANG"?)

SYSOP NAME name

You must use this statement to specify your first name.

SYSOP QTH location

You must use this statement to specify your location.

SYSOP ZIP postal code

You must use this statement to specify your zip or postal code.

MAKE

Use this statement if you want to produce the new MSYS.DEF file if there are no errors.

Here is a sample MSYS.OPT file for you to figure out!

```
port 1 at $3e8 int 4 speed 1200 name 223.70
port 0 at $2f8 int 3 speed 1200 name 145.01
digi bxng from 0 to 0 as bxng
digi bxng from 1 to 1 as bxng
digi bxn501 from 0 to 0 as bxn501
digi bxn501 from 1 to 0 as bxn370
digi bxn370 from 0 to 1 as bxn501
digi bxn370 from port 1 to port 1 as bxn370
digi test-4 from port 0 to port 1 as test-5
digi wa8bxn-3 from 0 to 0
bbs call wa8bxn-13
kanode call wa8bxn-1
answer  call wa8bxn-14
forward call wa8bxn-12
sysop name Mike
sysop qth Kirtland, OH
sysop zip 44026
id call wa8bxn
window display color
make
```

MASK INFORMATION

Several commands require the use of a MASK to determine the exact ports you wish to perform the function. For example, you. You may wish to only permit NetRom connects on port 2, or you may wish to have bbs connects only on port 3, or you may wish to monitor only ports 2 , 3 and 8 on the F2 screen. The MASK is merely a hexadecimal representation of the ports. The BIT PATTERN CHART is the way to determine what the MASK value will be.

BIT PATTERN CHART

To be used for DPorts, KPorts, MONPorts, NPorts and TPorts

MASK VALUE	PORT NUMBER							CONDITION
	6	5	4	3	2	1	0	1=ON, 0=OFF
0	0	0	0	0	0	0	0	All Off
1	0	0	0	0	0	0	1	0 - ON
2	0	0	0	0	0	1	0	1 - ON
3	0	0	0	0	0	1	1	1,0 - ON
4	0	0	0	0	1	0	0	2 - ON
5	0	0	0	0	1	0	1	0,2 - ON
6	0	0	0	0	1	1	0	2,1 - ON
7	0	0	0	0	1	1	1	0,1,2 - ON
8	0	0	0	1	0	0	0	3 - ON
9	0	0	0	1	0	0	1	3,0 - ON
A	0	0	0	1	0	1	0	3,1 - ON
B	0	0	0	1	0	1	1	0,1,3 - ON
C	0	0	0	1	1	0	0	3,2 - ON
D	0	0	0	1	1	0	1	0,2,3 - ON
E	0	0	0	1	1	1	0	1,2,3 - ON
F	0	0	0	1	1	1	1	0,1,2,3 - ON

When you have more than 4 ports, a 2 digit hexadecimal value is required. The following table represents the left-hand digit for ports 4,5, and 6.

MASK VALUE	PORT NUMBER 6	5	4	CONDITION 1=ON, 0=OFF
0	0	0	0	4,5,6 OFF
1	0	0	1	4 - ON
2	0	1	0	5 - ON
3	0	1	1	4,5 - ON
4	1	0	0	6 - ON
5	1	0	1	4,6 - ON
6	1	1	0	5,6 - ON
7	1	1	1	4,5,6 - ON

For ALL ports to be turned on, the MASK value would be 7F, and is written as $7F.
For Ports 0 and 1 on, the MASK value would be 3, written as $3.
For Ports 2 and 4 on, the MASK value would be 14, written as $14.
For Ports 3 and 5 on, the MASK value would be 28, written as $28.

DRSI CARDS

Minimum things that must be done to use DRSI cards:
Put HAPNDRSI.SYS in root (C:\) directory
Put HAPNKISS.COM where it can be found to be executed.
Put DEVICE=HAPNDRSI.SYS IRQ=7 (or whatever interrupt you have the board set up for) in CONFIG.SYS.
Reboot system.
Put:

```
  PORT 1 AT $0 INT 7 SPEED 1200 NAME *DRSI-0
  PORT 2 AT $1 INT 7 SPEED 1200 NAME *DRSI-1
```

in MSYS.OPT and run MUTIL function 9.
Notes: Change the port numbers to what is appropriate to your system . The ports on the
DRSI card are referenced by the numbers after the AT $. The INT 7 is taken as comments only (but INT something is required).
SPEED of the DRSI ports is really set on the DEVICE=HAPNDRSI.SYS statement in CONFIG.SYS . SPEED value given on port statement is more or less only comments.
NAME must begin with *, can be something like *145.01 if you prefer .

Before you start MSYS, you must run HAPNKISS.

That should get you started. For other configurations and options, read the appropriate documentation.

MSYS COMMAND SUMMARY

The following is a SUMMARY of the commands available at the LOCAL keyboard for MSYS. The commands may be abbreviated to the capital letters. This list does NOT include the BBS commands that are available only from the BBS. Most "LOCAL" commands may be executed from the BBS by prefixing them with a "/". Please refer to the MSYS SYSOP section for more detailed explanation of the commands.

```
*comment  Enables comment line from user to SYSOP
#msgs/use Displays total number of active msgs and known user on BBS
@home     Designates @ home bbs option
ABortfwd  Aborts forwarding as soon as possible
ADDUser   Allows sysop to add a user to bbs
APpend    Used to append a file to a message and other general uses
ARChive   Puts given @BBS in specified sub directory
ARP       Lists the known callsign/IP address
ARP add   Adds digipeaters to TCP/IP stations
ARP del   Removes TCP/IP stations from arp list
ATtended  Turn HF forwarding on and off
AUTODest  Add new DESts for known neighbors
AUTOHold  Hold messages that contain own call in R: lines
AUTOKill  Kill all messages after forwarding
AUTONode  Add new neighbors heard
AX25Backoff    Provides exponential delay to RNR messages.
AX25L2V2  Turns on Version 2 of the AX.25 protocol BY PORT #

BADuser   Enables the bad user list
BBs       Enter the BBS as a local user
BBSNode   If on and the node is enabled, the bbs will appear as a
          node to the network
BBSTimeout     Number of seconds of inactivity on bbs to disconnect
          user
BDigi     Specifies digipeaters for beacons
BElloff   Turns bells on or off.
BID       Manipulates bids in the BIDLIST.DAT file.
BINterval Nodes Broadcast interval
BMaxcon   Maximum BBS connects
BOOT      Reboots computer. (Remote sysop only)
BPorts    Used to indicate BBS ports only
BText     Specifies one line of beacon text
BUDCalls  Set or display the list of calls with SSID to be monitored
BUDList   For selective monitoring of BUDCalls
```

```
C#          Attempt to connect to station via digi list given on port #
CHAnstat    Displays status of a logical channel whose number is given
CHEck       Number of 1 second intervals of no activity to disconnect
CL B #      Change border color
CLrscrn     Clear window number given or * for clear all windows
CMdwindow   Specifies the window # to be used for commands (CM 4)
COPy sourcepathname destpathname   (include drive: if needed)
CONference  Used to enable/disable conference function in BBS
CTS         Allows specification for each port if hardware hand shaking
            should be used
CRetries    Sets number of retries for forward connect attempt
Cscript     Another form of the Connect command:

DEADletter      Used as default for unknown @BBS
DEBug           Used to display or modify memory
DELEfile        Specifies file name to delete
DELUDays        Specifies the number of days after which a user that
                has not connected will automatically be deleted from
                the system.
DELUSer         Deletes a user from the user list
DESt ADD, REP, DEL   Adds replaces or deletes destinations in
                nodelist.
DFree           Displays amount of disk space free
DIGipeats       Displays digipeater/gateway list
DIRectory       Displays directory for path specified Example:
                DIR mail\*.*
DISconnec       Forces a disconnect on logical channel specified
DOcmdfile       To execute a command file
DPorts          Allows sysop to selectively disable digipeater
DRoute          Callsign port# [digi1...digi8]
DTimedate       Displays current date and time
DUmpbbs         Disconnect any bbs that has connected and uses a
                command other than S or F.
DUPcheck        Checks R: lines for BID checks & title/from call check


EDit        Screen editor for files
EFile       Edits the specified file
EMsghdr     Edits the header of the specified message number in BBS
EUser       Edits specified user
```

F2 time Number of seconds to continue displaying monitored packets
         after F2 (function key) was last pressed.
FBBC   Sets the FBB Compatibility mode
FBBG   Sets the FBB forwarding group size
FBBStimes Allows specification of hours bbs is full function
FCall   Finds the logical channel(s) in use by given callsign
FFile   (find file) FF *.DAT C:\
FIle   Converts specified message number into file name given.
FMsg   Searches messages in MAIL.BAK for given pattern string.
FOrward Initiates forwarding immediately
FPactor Initiates PACTOR or GTOR forwarding
FRAck   Displays or sets time in seconds to retransmit a packet
FREE   Deallocates the logical channel specified
FSize   Displays the file size for the pathname specified
FTIme   Forward time
FTPgm   File Transfer Protocol
FUlldup Used on the specified ports if set to ON.

GEtmsgotd Reads and displays Message of the day file
GMtime   Sets GMT offset hours from local time for BBS use
GRep   Will search the given file for the specified pattern

HApnstat Displays HAPN/DRSI statistics.
HCAll   Allows you to properly specify your bbs hierarchical
         callsign.
HCLean   This command automatically deletes specified messages.
HEard   The J heard list recording is turned off if # > Dispatches
Help   Lists the names of the commands
HOLdlevel Parameter is a number which if the sum of word weights
         matched in message exceeds the value the message is held.
HOStname Hosts id name TCP/IP
HReplace Selects adding to end of existing hierarchical @BBS field
         or replace.

ID *   Forces transmission of an ID immediately (also mail for
list)
IGnore   Calls in budcalls list will not be monitored
IMport   Imports messages from given filename.
INitpr   Allows you to specify if printer should be initialized
         before each file is printed.
IPNCheck Inactivity timeout in seconds.
IPNFrack Retry timer in seconds.
IPNRetry Retries, default 5.
IPTtl   TCP/IP Time to live

```
JBbs      Lists (P)BBSs heard and their paths
JDigipeat Lists Digipeaters heard and their paths
JGateways Lists Gateways heard and their paths
JHeard    Lists stations heard and those connected to BBS
JKanodes  Lists KA Nodes heard and their paths
JMsys     List other MSYS systems heard
JNetrom   Lists NET/ROM nodes heard and their paths
JTcp/ip   Lists TCP/IP stations and their addresses

KEyboard  Assign keyboard to channel number specified
KILLproc  Abort specified process number (Can be hazardous!)
KPorts    Allows sysop to selectively disable k NODe

LCount    Counts number of lines, characters, and words in file.
LOCKkbd   Locks the keyboard.  OR unLOCKs keyboard if locked.
LOGclose  This command closes the log file, renames it to form of
          yymmdd.LOG and then opens a new log file.
LUsers    List known user calls that have connected to BBS
LU xxx    Selects: BADusers, BBSystems, EXPert, LIMited, NEW, LOCked,
          RESet, XFAst_set

          NOTE: Use only the first 3 letters of the item from the
          list for xxx!
LZhuf     Enables Huffman coding on PACTOR connects.

MAIlnew        Advises user of new mail.
MAKEPrivate    Sets/resets making messages private by default
MAStersrv      Specifies the master White Pages server (default is
               N6IYA.#NOCAL.CA.USA.NOAM)
MAXBullage     Used to discard bulletins older than a specified
               amount
MAXFOrward     Maximum # bytes to forward on specific port
MAxFrames      Display or change number of frames sent before ACK
               rcvd
MAXRead        Specifies maximum size message that can be read or
               file that can be downloaded
MCForward      Ports where forwarding is counted as bbs connects.
MCOn           Number of connects allowed to sysop keyboard
MDir           Creates given directory
MEMory         Display amount of memory free
MERge          Reads MSYSFWD.DEF and copies it to MSYS.FWD
MEXpert        If on, display the message of the day
MHclear        Clears various J heard lists
MIDchar        Message ID indicator (max 4 chars)
MINmem         Allows specification of the minimum free memory
MISsing        Reports number of times forwarding aborted due to
               missing text file
MONBframe      Enable/disable monitoring of Beacon (UI) frames
MONCC          Enables/disables monitoring of TCP/IP packets
MONCF          Enables/disables monitoring of NetRom packets
MONIFrame      Enable/disable monitoring of Information frames
MONITor        Enable/disable monitoring of incoming frames
```

```
MONOutgoing    Enable/disable display of outgoing frames
MONPorts       Select which ports to monitor [MASK]
MONSFrame      Enable/disable monitoring of Supervisory frames
MONUFrame      Enable/disable monitoring of Un-numbered frames
MONxx          Enable/disable monitoring of PIDs on screen F2
               monitoring
MSgtrailer     Enable/disable message trailer on messages read from
               bbs.
MYAlias        Display or change bbs alias callsign
MYBbs          Display or change bbs primary callsign
MYCall         Display or change Call/Answer callsign
MYIpaddr       Sets IP address that the system will respond to.
MYKnode        Display or change K NODe callsign
MYTcpcall      Callsign used when sending arp and TCP/IP frames

NBdcast        Enables Node Broadcasts on a per port basis
NDwindow #     Net default window size (#=frames 1-7)
NETBbs         Enable BBS cmd in net NODe
NETCon         Enable connect cmd in net NODe
NETFbbstime    Allows specification of hours NET is full function
NETFBBC #      NET/ROM FBB Compatibility
               0=No FBB Compatibility (Default)
               1=FBB Non-Compressed Compatibility
               2=FBB Compression Compatibility
NETFBBG        NetRom FBB forwarding group size (0-5)
NETMaxfwd      Maximum message size to forward through network
NETNode        Enable the Net Node
NETOnly        If on, only BBSs can connect to bbs via network
NETRead        Specifies maximum size message that can be read or
               file that can be downloaded through the Netnode
NETTalk        Enable Talk cmd in net NODe
NETXBatchsize  Sets Max batch size for RLI compression on network
NETXOption     If on, enables RLI compressed forwarding on network
NETYapp        If on, allows YAPP transfers via network
NEWUser #      Allows you to set the initial user flags
NInfo          Gives net NODe status
NNode          Takes you into the NODe as if you were connected to it
NOBEacons      If a given port is set to 1 then there will be no ID,
               mail for, or BText beacons
NOBUlls        When on, no bulletins will be forwarded during normal
               forwarding.
NODe SEnd, SAve, REStore, ADD, REP, DEL
NODETimeout #     Number of seconds of inactivity on network node
NPorts            Ports to which nodes broadcast
NRCall callsign   Sets net NODe callsign
NRId mnemonic     Sets net NODe mnemonic identifier
NTtl              Network time to live (max hops)

OCInit            Initial obsolescence count
OCMin             Minimum obsolescence to broadcast
OKfrom            Specifies which non-callsigns in from field.
```

```
PAClen    Specifies maximum number of bytes that are put in a packet
PATh      Display information about given user callsign
PC        Enables/disables callsign serverPForward Display
          information about given BBS callsign
PForward  Shows the forwarding path for messages @bbs
PIng      Determines Round Trip Time to destination
PMessage  Prints specified message
POrts     Display information about ports
PPersist  "Agressiveness" value (255 to 0)
PQuality  Quality for nodes heard on port
PRInt     Prints a file (on printer)
PROcesses Display status of processes
PRTmsgs   Enables/disables printing of messages
PScan     Enables Pactor Scan
PWindow   Sets window to be used by a process

Qmsg      Copies msg into editor for reply.
QUIT      Terminates execution of MSYS gracefully (more or less)
          Also done with Ctrl F4

RBid      Adds received BID/MID to R: Line
RDir      Removes the given directory
REadbackup     Allows you to read a message from the /MAIL.BAK
          directory
REMove DESt callsign     Deletes DESt through all neighbors
REName    Oldpathname newpathname (must both be on same drive)
REPlaces  Replaces incoming @BBS calls
RESPonstime    Number of seconds to wait before sending out a
          response
RESTore   Restores a message from the mail.bak directory
RETries   Displays or sets number of retries before disconnecting
REQDIR    Enables/Disables Request directory server
REQFIL    Enables/Disables Request File server
REQMaxfile     Used to set a limit on the size of files that may be
          requested from the REQFIL server
REQQTH    Enables/Disables Request QTH server
RLinetime Adjusts the R: line time for proper GMT offset.
RUn       Allows you a DOS program with many limitations

SCreensave     Enables/Disables screen saver
SEnd      Sends a line of text to specified channel: SE 1 <text>
SE-       Same as above with no "Message from Sysop": SE- 1 <text>
SHUtdown  If on, advises users of impending bbs shutdown
SLottime  Value (0 to 255)
SMTP      Simple Mail Transfer Protocol
SNow      Enables/disables changing screen only during retrace
SOrt      Will sort up to 200 eighty byte records
STatus    Displays some status information
SUspendtrace   Number of free bytes in input buffers below which
               monitoring is turned off
SYsop     If on, adds to SYSOP messages to You have...message count.
```

```
TABs      Enables/Disables tabs in the bbs
TACk      Transport ACK delay (RESPonsetime)
TBdelay   Transport busy delay
TElnet    Used to establish keyboard to keyboard connection
TERminal  A command which permits direct keyboard access to port
TFrames   Send test frames
TImezone  Allows entry of your time zone
TNatime   Transport no activity timeout time
TOdo      Causes the MSYSTODO.DAT file to be scanned immediately
TPorts    Enables/disables ports for TCP/IP
Trace25   Enables/disables trace of AX.25 state changes
TraceIP   Enables/disables trace of TCP/IP state changes
TREtries  Transport retries
TRUncate  filename Adds CR/LF to files as needed
TS (hhmmss)    TimeSet command - sets DOS time to given value (must
          be exactly 6 digits)
TTimeout  Transport timeout (Net FRACK)
TXDelay   Transmit Delay time value (0 to 255)
TXTail    Transmit Tail value (0 to 255)
TYpe      Allows you to type a file

Users <file_name> Writes out all users to file <file_name>

VAlidmsgtype   Permits control over second character in message type
VErifysysop Enables/Disables Sysop verification when connected to BBS
VIdeo     Displays or changes video monitor in use

WAtch     Toggles watching of specified channel number
WBackgrnd Sets background color for specified window
WColors   Gives list of colors and their numbers
WForegrnd Sets foreground color for specified window
WHO       Displays current users/messages
WHY       Displays why message was held
Window    Displays current window definitions
WPAge     Sets page number to display specified window
WPEvery   Specifies the number of days between sending out WP
          information
WQupdate  Worst quality DESt route to record
WRecord   Window record which sends all output from specified window
          to a disk file.  Useful for debugging
WSize     Specifies coordinates for specified window
W?        Displays window number in upper left of each window

XBatchsize     Sets max batch size in bytes for RLI compression
XOption   If on, enables RLI compressed forwarding on port

Yapp      If on, enables YAPP transfers on port
```

<Esc> Cancel partially typed command

Function key usage:

F1 through F4 Select page to display (0 to 3 respectively)

F6 Causes the following characters to the next return to be treated as a command even if keyboard currently connected to some other process (such as local bbs, etc.).

F8 key Used to blank the screen.

Alt-F1 Alt-F2 Alt-F3 keys work like DOS F1 F2 and F3 commands to retrieve/edit last sysop command typed.

Ctrl/F2 Suspends activity on the monitor screen (F2).
Ctrl/F4 Terminates MSYS as if QUIT command was used
Ctrl/F5 Same as the PRocess command
Ctrl/F6 Assigns keyboard to command processor

MOST COMMANDS GIVE INFORMATION ON THEIR USAGE IF YOU TYPE THE COMMAND NAME FOLLOWED BY A QUESTION MARK.

Example: WS ?

*comment
Not really a command as such, * may be used at the beginning of a line making anything else you type on the line a comment. Useful in MSYS.DO to temporarily remove commands.

#msgs/use
Displays total number of active msgs and known users on BBS . The number of messages that can be on the bbs is set by the Mutil Function 1 program. The maximum number of users is 500.

@home (ON/off)
If On, causes @bbs field of messages for registered users of the system to be replaced with the home BBS they gave with the NH command (the way it has been done in the past) . If Off, the @BBS field is not modified.

ABortfwd
Aborts forwarding as soon as possible (not always immediately). You may want to DISC the channel forwarding is using (particularly if its trying to connect) to speed things up.

ADDUser callsign
Adds a user callsign to the user file without that station having to connect to the BBS . This can be useful in conjunction with the automatic @BBS generation . After the new user has been added you automatically will EUsers for the callsign at which time you can fill in the desired fields.

APpend Sourcefile Destfile
This command causes the source file to be appended to the destination file. This command may be used to append a file to a message file, among other more general uses.

ARChive @BBS filename
This command will concatenate all current msgs with @BBS given to FILES/filename . No compression is done . Example: ARC RLIBBS RLINOTES/MSGS (This assumes you have a RLINOTES subdirectory in the FILES directory)

ARP
Lists known callsign/IP address correspondence used for TCP/IP connects. ARP stands for Address Resolution Protocol.

ARP CLEAR
Erase all of the ARP entries.

ARP ADD callsign port# ip-address[/#] [digi1 [digi2 ...[digi8]]]

Allows you to manually add (or put it in MSYS.DO) a station (and up to eight digipeaters) to the ARP list. Periods are optional in the IP address.

Examples:
arp add WA8BXN 0 44 70 4 8 NO8M
arp add N8HSP 2 44.70.4.10 WB8CQR-5 WB8APD

You can put a / and a number after the IP address . For example, you can say:
arp add W3AAA 1 44 80 1 1/24 K8AAA

This says that any IP frames going to an IP address with 44 80 1 in the left 24 bits should go to W3AAA (through K8AAA as a digi).
Thus the number after the / is the number of significant bits to keep in comparisons.

ARP DELete callsign

Removes a single entry from the ARP table.
Example: ARP DEL K8EIW

ARP REP callsign port# ip-addr[/#] [digi1 [digi2 ... [digi8]]]

Replaces existing arp entry for given callsign with new information specified.

ATtended [ON/off]

When set to ON all forwarding is allowed. If off, ports with radio speed of 300 will forward only those messages in which the from call is the idcall (messages that you originate) or the to station in the message is the station you would directly connect to for forwarding. HF reverse forwarding is also inhibited if ATtended is OFF. Note that the regulations appear to allow you to receive messages unattended . So if you don't have a STA for HF turn AT OFF when you aren't there and remember to turn it ON when you are. To be on the safe side, put ATtended OFF in your MSYS.DO file should the system reboot when you aren't present.

AUTODESt (on/OFF)

When on and network node is enabled allows automatic updating of network node destination tables when node broadcasts are heard from neighbor nodes.

AUTOHold (ON/off)

This command specifies what should be done with messages that contain in them R: lines with your Hierarchical BBS callsign (see HCA command). This normally indicates that the message has passed through your system before and probably is in a routing loop. Currently such messages are held . AUTO- Hold ON means to do this. If AUTOHold is OFF the messages will not be held.

AUTOKill [on/OFF]
     When on, non-bulletin messages are automatically killed after they have been forwarded.

AUTONode [on/OFF]
     If on and network node is enabled, node broadcasts heard from new neighbor nodes are automatically added to the list of neighbor nodes.

AX25Backoff [MASK]
     A 1 bit in a given port position in the mask causes the FRACK value for the port to be multiplied by the retry number to determine how long to wait before retransmitting packets that haven't been acked.

AX25L2v2 [port#] [on/OFF]
     AX.25 version 2 protocol: Version 2 can be enabled for all ports or specified ports. Version 1, the default, may be better for HF operation.

BADuser [on/OFF]
     When ON, the calls in the BUDCall list will be totally ignored by the system. Such stations will not be able to connect to the system in any way (AX.25 or TCP/IP) and will not be digipeated. Do NOT use SSIDs in this list so that the listed calls can be used to match any incoming SSID the station may use. Try to avoid using this facility unless really necessary as it does add some overhead in the processing of every packet received. Hopefully stations that misbehave will realize that they can be turned off and not persist in their activity.

     The BADuser command is more inclusive than the Bad User Bit which can be set for a given user. Setting the bad user bit only prevents that user from using the BBS. There is relatively little overhead in supporting the bad user bit since it is only checked when connects attempts to the BBS are made.
     Putting calls in the BUDCall list and turning on BADuser has a lot of overhead since every incoming packet has to be compared with each call in the BUDCall list.

BBs [window#]
     Enter the BBS as a local user this command gets you into the BBS as if you had connected to the BBS, with a few minor differences. Of course while you are using the BBS other stations can still connect as usual. When in the BBS you use the BBS command set (A,B,C,D,G,H,I,J,K,L,M,N,P,R,S,U,V,W,X,Y,?,*). If you use the L command you will see all messages including the private ones. Likewise you can Read and Kill all private messages.

You can include the contents of a file in a message you are sending by typing:

\+\ filename

at the point you wish to include the file. If you want to include the contents of a message file from the MAIL.BAK directory, type:

\+\#msgnumber

at the point you want to include the file. Example:

\+\#1538

You can specify a window number as an argument to the BB command. BB 0 will use the top half of the F1 screen for example. If you were in the BBS locally and leave it for some reason with it still active and use the BB command you will return to your previous session rather than starting a new one.

BBSNode [ON/off]

If on and the node is enabled, the BBS will appear as a node to the network (using the alias as the identifier and the BBS call as the call). This way connects may be made to the BBS directly from the network.

BBSTimeout #

Number of seconds of inactivity on BBS to disconnect user, default 300 seconds. Note that when version 2 of the AX.25 protocol is enabled, no activity for CHEck time does not causes a disconnect.

BDigi [digi1 [digi2 ... [digi8]]]

Lets you specify up to 8 digipeaters for beacons (id, mail, btext). The list of digipeaters can be different for each port. If only a port number is given, the digipeater list for that port is cleared.

Examples:

bd 4 digi1 digi2	sets digi1 and digi2 as digipeaters for port 4 only
bd digia digib	sets digia and digib for all ports
bd 5	clears digipeaters for port 5

BElloff (ON/off)

On	All bells are turned off.
OFF	All bells are sounded.
-1	Keyboard connects sound but not other connects.

BID

Manipulates bids in the BIDLIST.DAT file. The following operands may be specified:

+ bidstring	Adds given bidstring to bid file
- bidstring	Deletes given bidstring
= bidstring	Tells if given bidstring is present
#	Tells number of active bids/mids

BINterval number
Network node broadcast of known destination nodes is made every number seconds.

BMaxcon [port#] #
This command limits the maximum number of connects to the BBS on each port. BBS connects are exempted from the limit but count for other users. Normal users get a BUSY response when over limit. BM 1 allows 1 user per port. If a port is marked BBS ONLY (see BPorts) the BMax value specified limits the number of BBS connects. For non-BBS only ports, the BMax value does not effect number of BBS connects. See also the MCF command.

BOOT
Boots computer. You should have an autoexec.bat file set up that runs MUTIL 4 and then MSYS so that after you reboot the computer MSYS will be restarted. Note that this is an abrupt termination of MSYS. Files are not closed for example. The correct reply to the question asked by the boot command is Yes (CASE IS IMPORTANT).

BPorts [MASK]
This command is used to indicate which ports are BBS ONLY connects. A One in the corresponding bit position makes a port BBS ONLY. For example, BP 5 would only allow BBS connects on ports 0 and 2. The default is no BBS-only ports. If you do use this command to make BBS only ports, you better be sure to mark all the BBS stations that you want to forward to you on BBS-only ports with user flag of 2!

BText [port#] text
Lets you specify 1 line of up to 80 characters of your own beacon text. Can be different for each port. Use with great restraint!

BUDCalls [callsign1 [callsign2 ... [callsign10]]]
Set or display the list of calls with SSID to be monitored or ignored. This command is used in conjunction with the BUDLIST or BADuser commands. As operands you give a list of callsigns, with SSIDs, that are to be monitored or ignored. The calls should be separated by blanks. To list the current setting of BUDCalls, type the command with no operands.
Examples:
budc w1xx w2xx-1 w2xx
budcalls K9XXX
budc

Note: there can be up to 10 calls.

BUDList [on/OFF]

To enable only those packets with calls in the BUDCalls list to be monitored use the command BUDL ON. To monitor packets for all calls, use the command BUDL off. OFF is the default setting.

CHAnstat channel#

Displays status of a logical channel whose number is given. This command requires an active channel number as a parameter. Information about the channel will be displayed in response to this command. The information includes the Port number used, the call of the station connected to and a status byte in hex.

CHEck [number]

Number of 1 second intervals of no activity to disconnect if Version 1 of the AX.25 protocol is in use, or to send an RR Poll if Version 2 is enables (see AX25L command). To see what value is currently set, give the command with no parameters.

CL B number

Allows you to set the border color (used on all screens). Use the WC command to see the color list. Default color is black for the border. Any of the 16 color numbers (0-15) may be used.

CLrscrn [number/*]

Clear window number given or use * to clear all windows. If no parameter is given, then the command processor window is cleared.

CMdwindow number

This command is used to change which window is used by the command processor. The default command window is number 0 which by default is the upper half of display page 0. This command is useful when you want a larger window (to type a file for example) to be used by some command. The new window is cleared automatically by this command. You will then get the cmd: prompt. (page 0 =F1 page 1 = F2 etc.) A good windowk to use is 4 (CM 4) which is the one you see when you press F4.

COPy

Sourcepathname Destpathname (include drive: if needed) The copy command copies the specified files and will return an error message if it isn't able to copy all the records. Directories and wildcards can be used.

CRetries number

Sets number of retries that there will be for forward connect attempt. This value should probably be less than RETries.

C script-filename

In addition to the usual C# form (like C0) there is now a connect command in which you do not specify a port number with the C, nor do you give the call of the station to which you want to connect, but rather give the name of a file that contains a connect script to reach that station. The files must be in the /FWD directory and can have any name. The connect script is the same format used in the forward file except that the first line begins with C rather the other letters used in the forward file. You can thus use the command C HOME in which case in the FWD directory you would have a file called HOME. An example of the contents of this file is

```
c0 wa8bxn
@olvgn
#120
+to
.c akron
#180
+to
.c cle220
#300
+to
```

This example can be interpreted as follows. In the first line the C means this is a connect script, port 0 is to be used for the initial connect, and the ultimate destination is WA8BXN. The second line says to make a connect to OLVGN (which is a network node). The following lines are commands sent to the node, the # lines are times to wait for responses, and the +to lines indicate the responses to look for that indicate success. See the descriptions for forwarding scripts for more information on these various options. Another simple example of what you could put in a connect script file is

```
C0 WA8BXN
@WA8BXN V TOMSYS
```

That's the entire file; it is a simple way to put the digipeater (TOMSYS) needed to reach the destination station (WA8BXN).

CTS [port#] [on/OFF]

Allows specification for each port if hardware handshaking should be used with the TNC or Modem. If on, MSYS will suspend sending data to the TNC or Modem when Clear to Send is not true. This will be quite useful with AEA TNCs which have problems when MSYS sends packets too rapidly to them. Also useful with most Modems. Make sure you have the proper wires in the cable between the computer and TNC or Modem, and that you have your TNC or Modem configured properly if you try to use this option.

C# callsign [Via digi1 [digi2 ... [digi8]]]

This command is used to connect to another station. The letter C must be followed by a digit indicating the port number to be used for the connection. This is followed by a space. Next comes an optional window number to be used for the conversation. Next comes the callsign of the station you want to connect to, with SSID if needed. Following this can come Via and a list of digipeaters to use, separated by commas.

Examples:

```
c1 w1xxx
c0 w2xxx v k2xxx
c2 k3xxx-2 v n4xxx-3,a5xxx
```

When you issue this command the screen to be used for the conversation will be cleared and you will see a line "Enter converse for channel # port #". When connection is made you will get a message "*** Connection established to callsign". If the attempt retries out (determined by the RETries value), you will get the message "Quit converse for channel # because no connect!". In this case the default CALL process window may show "*** callsign busy" if this is why the connection could not be made. Pressing any key before getting one of these messages will abort the connect attempt and return you to the current command processor window. When you are connected to a station, you type lines to be sent to the other station and will see lines typed by the other station. Once you begin typing a line it will not be interrupted by lines from the other station until you press return. To terminate the connection, type ^C, the letter d (^Cd) and press return.

If you want to save what you are getting on disk, press the \ key three times and then press return. You will get the message "*** Capture file opened ***". All lines you receive will be displayed as usual as well as being written to the file CAPTURE.DAT. When you turn capture on with the \\\ at the beginning of a line, you can also specify a filename to be used in place of the default CAPTURE.DAT. For example you can type \\\ junk.abc to start capturing your conversation into the file JUNK.ABC. When you have already turned capture on during a connection, if you type something like \\\morejunk.xyz then the current capture file will be closed and now your conversation will be saved in the file MOREJUNK.XYZ. Three \followed by return will alternately suspend and resume the capturing of lines to disk. When you disconnect (or get disconnected) the capture file will be closed. Note: That the default capture file name is always CAPTURE.DAT. These files can be viewed by using the TYpe command or edited with the EFile command You could for example connect to some other BBS, read a message found there after turning on capture and then include the capture file in one of your own messages. This procedure is explained in more detail for the BBs command above.

You can also send any file on your system as if you typed it on your keyboard to the station you are connected to by typing
\+\filename
at the beginning of a line. Example:
\+\ files\stuff
would send the file called stuff from the files subdirectory.

DEADletter (bbscall)
This command specifies a BBS call that will be used for private messages entered on your system without @BBS and whose home bbs is not known. This permits you to forward the messages to a WP server or another station that enjoys tracking down unknown souls.

DEBug
This command is primarily used during development of MSYS. Its results will change from release to release and are not further documented. Although it should not do any damage it will probably be of little use to the normal Sysop.

DELEfile pathname
This command is used to delete a file. You must give as a parameter a single file path name with or without wildcards. The parameter can contain a drive specification and/or directory names for the given file.
Example: DELE mail.bin*.*

DELUDays number
Specifies the number of days after which a user that has not connected will automatically be deleted from the system. Default is 100 days. Locked users (see bit list in EUser cmd), BBS call signs and bad users are not automatically deleted.

DELUSer callsign
This command removes a user call from the BBS user list. The callsign given as a parameter must match the a call seen on the list produced by LUsers exactly (including case). Use this command to REMove users that haven't been active lately when you are getting near the capacity of the list (currently 500 users). The last time a given station connected to the BBS may be obtained by using the P command while in the BBS. If you give * for the callsign (as in DELUS *) then most user entries that contain garbage callsigns or ? for the user's name. This option is pretty safe to use, but if in doubt make a copy of the MSYS.USR file if you wish.

DESt ADD [id:]destcall neighborcall neighborport obs# quality
Adds new network destination node which is routed through given neighborcall using specified port. The obsolescence count and quality values are stored as well.

DESt REP [id:]destcall neighborcall neighborport obs# quality
Changes obs# and quality for specified destination node

DESt DEL [id:]destcall neighborcall neighborport

Deletes route to specified destination node through given neighbor. When all routes have been deleted, the destination itself is also removed from the network node tables. See also the REMove command.

DFree [driveletter:]

Displays the number of bytes free on a disk. If no operands are given the space is for the default disk (normally C:). If a letter is given as a parameter, the number of free bytes will be given for that drive. Be sure the drive is ready before using this command. If its not and you get the DOS error message, type I for ignore.

DIGipeaters

This command displays the list of digipeater definitions and related information. The output of this command is the same as using the ID command in the BBS.

DIRectory [path]

Displays directory for path specified Example DIR mail*.* This command displays a sorted directory listing. Only the names of the files (or directories) are given, not the sizes. To get the size of a file, use the FSize command described below. Names of directories are enclosed in <>. If no argument is given, you will get a directory for the current default directory (normally C:\MSYS). To list some other directory, give the path to that directory and a file name of *.*. Other file specifications can be given for a selective directory list. If you include a drive specification in the pathname, be sure the drive is ready before using this command. Examples:

```
dir mail\*.*
dir a:*.*
dir files\*.c
```

DISconnect channel#

This command can be used to force a disconnect on the specified active channel. If the channel you specify was in the process of disconnecting when you use this command, then the channel will be immediately disconnected. Otherwise the channel will begin disconnecting when you use this command. Use it a second time for that channel if you want to speed up the disconnect.

The disconnect command can be useful when you want to shut down the system. Another good use is force a disconnect from the bbs you are forwarding to after you have used the ABort command to abort forwarding. The disconnect command also will properly terminate a modem or a pactor connection.

DOcmdfile filename

This command is used to execute the sysop commands found in the specified file. A command file will contain the same things you would normally type as commands. To execute a command file, type DO followed by the command file name. Example: DO cmds.dat

DPorts [MASK)

A one in the mask for a given port permits digipeat on that port.

DRoute callsign port# [digi1 ... digi8]

DRoute DELete

The first form of this command allows you to specify the TCP/IP station to which IP frames are routed by default (Default Route). You specify as parameters the callsign of that station (with SSID) and the port number used to reach that station. Optionally up to 8 digipeaters can also be specified to be used to reach that station. The second form of this command allows you to turn off default routing of IP frames. If you do have a default route specified, then all incoming IP frames that are not otherwise handled by existing ARP table entries will be sent to the default station. This station will also be used for any server functions you initiate (telnet for example) to IP addresses that are not currently in your ARP table. Care must be taken in deciding if this command is appropriate to use in your particular situation.

DTimedate

This command displays what the computer thinks the current date and time is. The time can be changed with the TS command. To change the date you must exit MSYS and use the DOS date command.

DUmpbbs [ON/off]

Disconnect any BBS that has connected and uses a command other than S or F. A BBS is defined as a station that sends a SID ([...]) line. This is useful in cases where the two BBB's get out of sync and message titles/text are being interpreted as commands.

DUPcheck [arg]

0 - Disables R: line BID checks & title/from call check
1 - Holds messages that fails these checks.

EDit filename

Screen editor for files. Escape gets to command mode, do help for command list. See section on EDITVIEW later in this document for more information.

EFile filename

This command is used to edit an ASCII file with re- cords <= 80 bytes in length. At least one operand is required which is the pathname to the file you want to edit. A second operand may optionally be given that gives the size of the work area in lines to be allocated (default is 200). See the separate documentation for EDITFILE for more information on the editor.

EMsghdr msg#

This command allows you to edit the fields of a message header. One operand is required which is the message number to edit. This command also automatically sets the message length to the actual size of the associated message file. This is useful when you have replaced a message file with some other file as described above. This command is also useful to change the status of messages to ALL (or others) that have not been marked as being read (but actually have been) so that they don't appear in the MAIL FOR list sent out with each ID. If you press Esc when editing you will skip the current and remaining fields and see the header again. This allows you to bypass having to press return for all the fields you don't want to change.

If you place a string of characters that begins with a period after the message number, it will be appended to the existing @BBS field of the given message. Thus if the @BBS of message 123 was WA8BXN then:
EM 123 .NTSOH
would cause the @BBS field of message 123 to become WA8BXN.NTSOH.

If you type a blank for the BID field the Bid is cleared entirely (you still need to get rid of the $ in the type).

EUser callsign

Edits user information. Also displays the meanings of the possible user flags bits. The EUser command will give a display similar to the following:

```
cmd: eu k8eiw
K8EIW is Don located at Cuyahoga Falls, OH  Last new message 36457
Last connected Sun Nov 21 23:29:36 1994 on port -1 Flags=$1A53
Lines=19
Postal code:44223 Home BBS: WB8BII Last WP Update: Jul 15 01:00:03
1990
Path to K8EIW is <Local Console>
User flags:
$8000 New user
$2000 No auto LC in bbs          $1000 Autoreply
 $800 YAPP upload authorized      $400 User password required
 $200 File upload authorized      $100 Lock user
  $80 Can't use S cmds             $40 Multiline packets (XF)
  $20 Need to send WP Update       $10 Remote sysop
  $08 Limited function BBS         $04 Bad user
  $02 BBS                          $01 Expert user
```

The flags add hexidecimally per character position. For example, if a user has both File upload and YAPP upload privileges, the flag for that position would be "A".

To prevent WP messages from being issued on a particular call, such as V1SIT, set the flag to $100 at a minimum.

The No Auto LC, Autoreply, Multiline Packets, and Expert User flags can be set by users with the appropriate XC, XR, XF, or X commands.

In the above example, the flags 1A53 mean this user has Autoreply turned on, YAPP and File upload permitted, Multiline packets will be sent, is a remote sysop, and is a bbs and an expert user.

F2time number

The number of seconds to continue displaying monitored packets after F2 (function key) was last pressed. The default is 600 seconds (10 minutes). Thus 10 minutes after you last press F2 monitoring will be turned off until you press F2 again. If you set F2 to 0 this function will be turned off. There is a fair amount of overhead in formatting all the information displayed on the F2 screen so on slower systems it is a good idea to allow the display to be suspended when you aren't actively looking at it!

FBBCompat [port#] [0, 1 or 2]

This command determines the level of compatibility with FBB style forwarding. FBBC 0 means MSYS will forward non-compressed to all stations as before. FBBC 1 means that MSYS detects the "F" in the SID, it will forward non-compressed, but in FBB style groups. FBBC 2 Means that if a station identifies itself with a "B" and "F" in the SID, it will forward compressed in groups per the FBB style format. It should NEVER be specified for the modem port due to the high probability of buffer overflows.

FBBGroupsize # [1-5]

This command determines the maximum number of messages that may be forwarded out in one group using FBB compliant protocol. The maximum group size is 5. In FBB systems, the maximum group size is 10K bytes, and the number of messages proposed in the group for forwarding will be limited by this value. MSYS does not have this limitation.

FBBStimes [Port#] [time-range]

Allows specification of hours BBS is full function for each port. Default is 0-23 which is all the time. The format of the time range is the same as used on the ! lines in the forwarding file.

FCall callsign

This command finds the channel associated with a callsign. The single required operand is a callsign without SSID. All channels in use by that callsign (with any SSID) will be displayed giving the Port, process number and name and complete callsign (with SSID). This can be useful when you were in the middle of a conversation with some station (and still connected) but switched the keyboard to some other process (maybe the command processor) and now want to get back to that conversation. Use the KEyboard command with the appropriate process number to do so.

FFile filenamepattern [starting-path]

When you give a file name pattern and optionally a starting path location this command will list all files that match the pattern. The following would find all files that have the extension .DAT on your C: disk

FF *.DAT C:/

The output from this command gives the date, time, size and complete path names for all files that match the search specification.

FIle msg# filename

Converts specified message into a file with name given. This command removes the R: lines at the beginning of the file it creates. If you really want to keep them, use an appropriate rename command instead.

FMsg pattern-string

Searches all messages in MAIL.BAK directory for given pattern string. For example, to find messages that refer to hurricanes you might type:
FM hurricane

FOrward [callsign]

This command causes forwarding to begin immediately if it is not currently going on. You might want to use this command when you have just entered a new message with an @BBS specification that you want sent out now. FO with a callsign of a single station begins forwarding to that station. This must match the callsign (including SSID) found on the F/R/P line in the forward file.

FRAck number

This commands allows you to set or display the number of seconds to wait for an acknowledgment to arrive for a packet that is sent out. If it doesn't arrive within the number of seconds specified the packet will be retransmitted. The retry count for the channel will also be incremented. Making this parameter to small will congest the channel and cause unwarranted disconnects. The default is 6 seconds. If in doubt, error on the high side! (port number optional) All IP (and IP through netrom) frame retry timing uses a dynamic FRACK value.

FREE chan#

This command allows you to immediately release an allocated channel. The process using the channel will see this as a disconnect. This command will not be needed very often. Disconnecting which frees the channel will automatically occur after the CHEck interval has expired with no activity. Sometimes unpredictable results can occur with this command. It is best to permit the DISC command do it's thing.

FSize filename

This command can be used to determine the size of a disk file. One operand is required which is the pathname to the file you want the size of. It may include a drive specification and directory paths in addition to the file name. Wild cards are not allowed. Be sure the drive containing the file is ready before using this command.

FTIme number

This command sets the time in minutes after the hour that forwarding will start. If time is negative number such as -5 then you will forward every 5 minutes.

FTPgm host-id

Takes a host-id as an argument (it can be either an IP address in the square brackets or a symbolic IP address). It establishes a connection to the FTP server of the specified system. FTP is the File Transfer Protocol used in TCP/IP.

FUllduplex [port#] [on/OFF]

Enables full duplex operation in the TNC on selected ports. This may be useful for satellite operation or with full duplex repeaters. Examples:

```
FU ON     turns full duplex on for all ports
FU 2 OFF  turns full duplex off for port 2
```

GEtmsgotd

This command causes the Message of the day file (MSYS.OTD) to be read and displayed. The Message of the day is sent to all stations connecting to the BBS. You can create and/or change this file using the EFile command. Keep the contents short! See also MEXpert command.

GMtime hours

This sets the GMT offset from the local time. The computer can be set for local time. GMtime will then offset MSYS time to the proper GMT time. If the TImezone GMT option is set, MSYS will insert the proper GMT in R: lines. Mutually exclusive with RLinetime.

GRep filename search-string

Will search the given file for the specified pattern, displaying lines that contain the given string. Case is ignored in the search. Wild cards are NOT allowed.

HApnstat

Provides HAPN/DRSI interface statistics.

HCAll string

This command allows you to properly specify your BBS hierarchical callsign. This is the callsign put in the R: lines of messages as they pass through your system. If you don't give this a value, your ID callsign will be used. There is a maximum of 49 characters for this entry. Hint: if you run more than one copy of MSYS for some reason and pass messages between the two systems, make your HCAll slightly different on each. I might use for example WA8BXN.OH.USA on one and WA8BXN.OH.USA.NA on the other. This will prevent unnecessary holding (or deleting, see OLdbids) of messages validly passed from one system to the other.

HCLean [number]

This command allows you to automatically delete specified messages every day at specified time (see MSYS.HCL in index). If no operand is given HCL shows time in minutes after 00:00 (midnight) that housecleaning happens. If a number is given as an operand it sets the time for housecleaning in minutes after 00:00 (midnight).

HCLean NOW

Manually initiates housecleaning.

HEard number

The J heard list recording is turned off if the number of dispatches per second in the last minute is less than the number given. Default value is 3100 which j heard recording. J Heard list recording takes quite a bit of computation for every packet heard. This command was added to speed things up on slower computers. (you might want to use a value of 10) Help This command lists the names of the available commands with the required part of the command name in capital letters.

HOLdlevel number

Parameter is a number which if the sum of word weights matched in message exceeds the value, the message is held. Default is 0, which will hold if any words are matched. See Automatic Message Holding

HOStname string

Sets TCP/IP hostname. Maximum of 49 characters.

HReplace [ON/off]

This command selects adding to end of existing hierarchical @BBS field info from BBSTONTS.DAT file (if HRep is off) or replacing anything past the first part with the information from the BBSTONTS.DAT file (if HRep is ON). Note that if some MSYS knows how to forward from some existing part of the @BBS field, no changes are made to it.

ID [*]

This command will send out an ID on all the ports immediately. You might use it to see if all your radios go into transmit as they should. It also will send out the MAIL FOR list. The text of the ID is not displayed at this time, but is shown once during initialization. Nothing is displayed by the automatic IDs sent out (on the F1 screen). To see what mail is outstanding for local users, type ID. To really force an ID to be sent, type:

```
 ID *
```

IGNORE [on/OFF]
When on calls in the budcalls list will be ignored.

IMport filename
Imports messages from the given file. The file must follow the form of the files in the mail.bak directory.

INitpr [on/OFF]
Command allows you to specify if printer should be initialized before each file is printed. Set it to ON if you need the initialization. If you need to press buttons (to select draft or letter quality mode for example) that you don't want cleared before each file, set it to OFF.

IPNCheck number
Inactivity timeout in seconds for TCP/IP connections, default 600.

IPNFrack number
Retry timer in seconds for TCP/IP connections, default 30.

IPNRetry number
Retries for TCP/IP connections, default 5.

IPTtl number
Sets TCP/IP time to live value. This is the number of times a TCP/IP frame will get relayed before it gets discarded. Default is 16.

JBbs [port#]
Lists BBSs heard and their paths for given port or on all ports if no port number is given.

JDigipeat [port#]
Lists Digipeaters heard and their paths for given port or on all ports if no port number is given. JD command screens out invalid callsigns.

JGateways [port#]
Lists Gateways heard and their paths for given port or on all ports if no port number is given.

JHeard [port#]
Lists stations heard and those connected to BBS or PBBS This command gives a list of the last 10 stations that were heard on the ports or connected to the BBS or PBBS. If a port number is given only stations heard on that port are listed. An asterisk after a callsign indicates that the station was heard through a digipeater.

JKanodes [port#]
Lists KANode beacons heard and their paths for given port or on all ports if no port number is given.

JMsys [port#]
Lists other MSYS systems' beacons heard for given port or on all ports if no port number is given.

JNetrom [port#]
Lists NET/ROM node beacons heard and their paths for given port or on all ports if no port number is given.

NOTE for all the J commands: These commands give the PBBS systems, KANodes and other beacons that have been heard on the ports. A BBS is a system that sends out a beacon that contains /B after what looks something like a callsign. A KANODe is identified by /N in its beacon to ID. For both of these commands the date, time, callsign and path are given. This pair of commands is useful to identify the PBBS that goes with a given KANODe. The Net nodes are also included. The real call for such nodes are given in [ ]. Real NetRoms get the real call in { }.

JTcp/ip [port#]
List TCP/IP stations heard and their addresses for given port or on all ports if no port number is given.

KEyboard chan#
The KEyboard command allows you to manually reassign the keyboard. As a parameter you give the channel number of the process you want to have the keyboard. You can get the channel number from the lower right part of the F1 screen. To get the keyboard back to the command processor so you can issue this command either press ^F6 to assign the keyboard to the command processor or press F6 to temporarily send keyboard input to the command process up to and including the next return key. When you press a key, it is placed in the input buffer of the process to which the keyboard is logically connected. Initially this is the command process. When you give certain commands (such as BB, PB, C) the keyboard will be assigned to the new process created for the command. When the process is terminated, the keyboard is automatically assigned back to the command scheduler.

KILLproc process#
Abort specified process number (Can be hazardous!). Never use this command to kill a permanent process. Better yet, just don't use this command! Process numbers can be viewed using the PROcess command.

KPorts [MASK]
This command allows you to selectively disable the K-node on any ports.
A one bit for a given port number enables the K-node on that port.

LCount filename
This command counts the number of lines, characters and words in a given file. Note: This is NOT the LC command found in the bbs.

LOCKkbd
This command "locks" the keyboard so that any keys pressed are ignored. I put it in for two reasons. First, I am getting RF into my keyboard from HF radios and it generates random characters. Second, my 3 year old son sometimes likes to press the keys. To "unlock" the keyboard, simply type "lock" again (no CR needed this time). When unlocking the keyboard, "lock" must be all LOWER case.

LOGclose
This command closes the log file, renames it to form of yymmdd.LOG and then opens a new log file. Caution: Do NOT use more than once per day!

LUsers
Use this command to see who has connected and more importantly how many users are in the user file since it can hold only 500 users currently. LU takes an optional argument to list users flagged as BAD, EXPert, LIMited, LOCked, XFAst, NEW and BBS system. Example: LU EXP
LU RESET clears all new user flags.

MAKEPrivate [ON/OFF]
Sets/resets making personal messages private by default.

MAStersrv hierarchical-address
Specifies the master WP server (default is N6IYA.#NOCAL.CA.USA.NOAM and probably shouldn't be changed). If MAS is set to NONE, no WP messages will be sent.

MAXBullage number
This command may be used to discard bulletins older than a specified amount as soon as they are received. Default age is 32000. A reasonable value might be 21 days. Bulletin age is determined by the date on the last R: line found in the bulletin. If MAXBullage is set to a negative number, bulletins older than the number given (in days) will be held rather than discarded.

MAXFOrward [Port#] # bytes

This command permits you to set the largest message size that will be forwarded on a port. This may be useful for HF forwarding.

MAXFrames number

Display or change number of frames (1 to 7) sent before waiting for and ACK (RR) to be received. A value of 4 is generally good for VHF ports. Use 1 or 2 on HF ports. For very good vhf paths and RS-232 wire links, 7 may be a good value.

MAXRead [port#] number

Specifies maximum size message that can be read or file that can be downloaded. If no port number is specified value applies to all ports.

MCForward [MASK]

Allows you to control Multiple Connections during Forwarding. The operand for this command is a port bit mask. Default is all 1's in the mask which allows forwarding to go on as in previous versions. If a particular port has its bit set to 0, then forwarding on that port is counted as a BBS connect on that port. Thus if you have BMaxcon set to 1 for that port and you are forwarding on that port, a BBS connect will not be allowed (since you are already using the port). Also, if there are already BMaxcon BBS connections for that port forwarding will be bypassed for that port. Careful use of these commands should reduce the congestion on HF forwarding as you now have complete control over the level of multiconnection for both BBS and forwarding activities combined. In the [MASK] argument you put 1's in the ports on which you want forwarding to be counted as a BBS connect. Using MCF along with BM (and BP if you have marked any stations as BBSs), allows you to limit a given port to a single BBS connect or forward at a time. If you want this on port 0, use the following:

```
MCF $FE
BM 0 1
BP $01
```

MCOn [number]

Specifies number of connects allowed to the SYSOP keyboard at one time if MCon 1, only allows an incoming call to the key board to be answered if there are no other keyboard connections. MCOn 0 prevents all incoming keyboard requests.

MDir directoryname

Makes specified directory.

MEMory

Displays amount of memory free. This number may not include the amount of memory available in the "holes" created by stations disconnecting.

MERge

This command reads MSYSFWD.DEF and copies it to MSYS.FWD, expanding any include files ($filename). A new file is also produced, MSYS.BBS that is a sorted list of all possible BBSs that can be forwarded to (including things like ARRL, 44*, etc.). This command was added for 2 reasons: previously, if there were any include files in the MSYS.FWD files they did not get read when a PF command was used and secondly, the new file is needed for hierarchical forwarding. The leftmost part of the hierarchical route only is used for forwarding and the contents of the new file were needed for efficient processing. So, the bottom line is rename your existing MSYS.FWD file to MSYSFWD.DEF and then use the MERge command. Any future changes you make to your forward file must be done to MSYSFWD.DEF followed by use of the MERge command. Its called MERge because it merges in the include files.

MEXpert [on/OFF]

If on, the message of the day will be displayed to all non-BBS marked stations. If off, it will only be displayed to non-expert, non-BBS stations.

MHclear [letter [port or B]]

Clears the various J heard lists. Letters are HBKDN or G (Heard, BBSs, K Nodes, Digi's, NetRoms). Examples:

```
MH        clears everything.
MH D      clears all Digi's heard
MH D 1    clears Digi's heard on port 1
```

MIDchar string

Message ID indicator (max 4 chars) to be sent in SID. If non-blank, MIDS will be sent for private messages when forwarding to systems that have the same string in their SID (System IDentifier, the stuff in [ ] when a connect is made to a BBS).

MINmem number

Allows specification of the minimum free memory as reported by the MEM for which a new connect request will be accepted. Default is 32000. Values less than 16000 probably will be disastrous. A value of 32000 or more is strongly recommended, particularly if you are using compressed forwarding.

MISsing

Reports number of times forwarding aborted due to missing text file.

MONBframe [ON/off]

Enable/disable monitoring of Beacon (UI) frames

MONCC [ON/off] - enables/disables monitoring of TCP/IP packets

MONCF [ON/off] - enables/disables monitoring of NetRom packets

MONIFrame [ON/off]
Enable/disable monitoring of Information frames

MONITor [ON/off]
This is the master command for monitoring packets heard. If this is set to OFF then there will be no monitoring. If set to ON then the types of packets selected will be monitored for calls as selected by BUDCalls on the ports set by MONPorts.

MONOutgoing [ON/off]
Enable/disable display of outgoing frames

MONPorts [MASK]
This command is used to enable or disable monitoring of the various ports on the F2 screen. See the section on MASK on how to set the mask for the desired ports.

MONSframe [ON/off]
Enable/disable monitoring of Supervisory frames (RR, RNR, REJ).

MONUframe [ON/off]
Enable/disable monitoring of unnumbered frames (SABM, UA, DISC, DM)

MONxx
Specifies any hex value for PIDs to ignore on screen 2 monitoring. Example: MONCC ON would enable monitoring of frames with PID of CC
Note: You must specify exactly two hex digits

MSgtrailer [ON/off]
If on when a message is read on the BBS (except with RN or RE) a line is sent at the end of the message indicating its number and who it was from. (Default is ON)

MYAlias string
Display or change BBS alias callsign. Must be 1 to 6 characters, no SSID.

MYBbs callsign
Display or change BBS primary callsign. Can have SSID.

MYCall callsign
This command displays your keyboard callsign if there is no argument, or sets the keyboard callsign (used to connect and respond to connects) to the call given as an argument (including SSID).

MYIpaddr ip-addr
     Takes four integers each in the range 0 to 255. It sets the IP address that the system will respond to. Example: MYI 44 70 4 6 If MYIpaddress is not specified, or set to all zeros, the TCP/IP support will be inactive.

MYKnode callsign
     Display or change K-node callsign. Should have SSID.

MYTcpcall callsign
     Display or change a callsign (with SSID) that will be used when sending ARP and TCP/IP frames. It should be the same as the network node call in most cases.

NBdcast [port#] [ON/off]
     If OFF, node broadcasts will not go out on port, but the node will otherwise operate on the port (assuming it is enabled using NP and other commands).

NDwindow number
     Network default window size in frames. Default is 4. Note that his has nothing to do with windows on the screen.

NETBbs (ON/off)
     Enable/Disable BBS command in network node.

NETCon (ON/off)
     Enable/Disable Connect command in network node.

NETFBBC number NET/ROM FBB Compatibility Flag
     0 = No FBB style forwarding (Default)
     1 = Uncompressed FBB forwarding protocol
     2 = Compressed FBB forwarding protocol
     This option only has meaning when connected to FBB compatible systems through the network node.

NETFBBG number
     Sets maximum number of messages included in a group when forwarded using network (1-5 messages per forward group)
     This option only has meaning when connected to FBB compatible systems through the network node.

NETFBBStime [hourlist]
     Allows specification of hours BBS is full function when accessed through the network. Default is 0-23 which is all the time.
     Hour list has same format as that used in the ! lines in the forwarding file.

NETNode [ON/OFF]
     Enables the Network Node

NETOnly [on/OFF]
This command is used to indicate if BBSs only or any stations can connect to the BBS via the network. If ON, only BBSs are allowed to connect via the network. If OFF, then any station can connect to BBS via network.

NETRead number
Specifies maximum size message that can be read or file that can be downloaded when connected to the BBS through the network.

NETTalk (ON/off)
Enable Talk command in the network node.

NEWUser number
Allows you to set the initial user flags for all new users (see EUuser command for the bit position meanings). Thus you can choose to make new users limited (or I suppose for a closed board, "bad users") until they identify themselves somehow.

NETXBatchsize
Sets the number of uncompressed bytes available for RLI compressed forwarding via the network port.

NETYapp [on/OFF]
Enables YAPP (Yet Another Packet Protocol used for binary file transfer between user and BBS) for connections made to the BBS through the network. Users must be enabled with user bit $800 to be able to upload files.

NInfo
Gives network node status display. Same output as U * command from in the network node.

NNode [window#]
This command takes you into the NODe as if you had connected to it. When in the NODe you can issue SYSOP commands if you need to by putting \ or / in front of them as you can in the BBS. Particularly useful are the NODe, DEStination and REMove commands when in the NODe to modify neighbor nodes and DEStination nodes. You can optionally specify a window number to use (4 might be a good one).

NODETimeout number
Number of seconds of inactivity on network node or K node at command prompt to disconnect user. Default is 300 seconds (5 minutes).

NOBEacons [mask]
If a given port is set to 1 then there will be no ID, mail for, or BText beacons.

NOBUlls (on/OFF)
     When on, no bulletins will be forwarded during normal
     forwarding. Useful in clearing backlog of messages,
     particularly during earthquakes and hurricanes!

NODe ADD [nodeid:]call port quality [digi1 [digi2 ... [digi8]]]
     Adds neighbor network node using given port and quality

NODe DEL call port
     Deletes neighbor nodes & its DESts

NODe REP [nodeid:]call port quality [digi1 [digi2 ... [digi8]]]
     Changes quality of neighbor network node

NODe REStore
     Read net NODe Database from disk. If you are using the network
     node, you would probably put this command in your MSYS.DO file.

NODe SAve
     Write network node Database to disk.

NODe SEnd
     Force known network node destination broadcast.

NPorts [mask]
     Ports to which network node broadcasts are made. Default is 0.

NRCall [callsign]
     Sets net work mode callsign. It should have an SSID. Example:
     NRC WA8BXN-2

NRId [mnemonic]
     Sets net NODe mnemonic identifier for network node. Example:
     NRI 460406

NTtl [number]
     Network node time to live (max hops). Default is 64.

OCInit [number]
     Initial obsolescence count for network node. Default is 6.

OCMin [port#] [number]
     Minimum obsolescence to broadcast a destination node in network
     node broadcast. OCM is settable for each port. Default 10

OKfrom List of acceptable non-calls for originating station (Maximum
     80 characters). Default is: WP REQWP.

PAClen [port#] [number]
Specifies maximum number of bytes that are put in a packet before it is sent. Possible range of values is 1 to 255. If packets will go through a network node on their way to their destination 236 should be used. On HF a value of 40 is generally good.

PATh callsign
Displays information about given user callsign (same as P on BBS)

PC [on/OFF]
This command enables or disables the on-line Callsign Server.

PForward callsign
Displays information about given BBS callsign (same as PF on BBS)

PIng destination
Where destination is either an IP address like [44.70.4.6.] or a mnemonic defined in MSYSHOST.NET. The reply comes in window 2 (where the connect and disconnect messages appear)and gives the round trip time in seconds.

PMessage message#
This will print the given message. If you have MAIL.BAK as a directory then the file MAIL.BAK/MSG###.DAT will be printed. This will look nice. If you don't, then MAIL/MSG###.DAT will be printed. This will be only the message text. This command just makes it easier to print a message than using the PRInt command.

POrts [port#] [ON/OFF]
Various information is given by this command if no operands are specified. Interesting to look at but may not be very useful. POrt can have operand on/off. When off a port will not respond to the id callsign (with any SSID) and will no longer digipeat or be usable from the KANODe. BBS will not id unless you use it. Id will be very short if it has to id (no text for the UI frame).If you call a station with the port off this will still work because incoming packets from that station don't include a connect packet.If you give a PO [port#] with no operand, the port will be turned off.

The POrts command will take a single parameter (ON or OFF) in addition to an optional port number. With just the single parameter, all ports are turned on or off. Useful when you want to shut down the system. PO OFF allows the current users to finish while preventing further connects.

PPersist [port#] value

Sends number given to TNC 255 to 0 as persistence value. Basically the higher the number the more likely the TNC will actually transmit a frame (if it has one to transmit) when it hears the channel is clear. Better overall channel throughput will result if all stations use a moderate value of persistence (50 to 100). A value of 255 says to transmit as soon as the channel is clear.

PQuality [port#] number

Default is 192. Sets quality for network nodes heard on port.

PRInt [filename]

Prints a file (on printer) The argument is the pathname to the file to print. Make sure the drive is ready before using this command. Make sure the printer is ready (and exists) before using this command. Printing is done to LPT1: via BIOS calls. This command will queue up to 25 print requests. PRInt with no operands lists the files in the queue.

PROcesses

This command displays for each active process its number, program running for that process, an integer parameter passed to the process (usually a port or channel number), the least amount of memory left in the stack for the process (labeled SPmin), and the window used for output by the process. Of particular importance is SPmin if the system does bizarre things. If this number is less than say 100 the stack is too small. If its a negative number, you have problems for sure!

PRTmsgs (on/OFF)

Enables printing of all incoming messages as they are received.

PScan [port#] |[freq]

Enables the PACTOR scanning routines

PWindow number number

If given one parameter this command displays the name and window used by the given process number. If two numbers are given as parameters, the first is the process and the second number is the window number to assign to that process.

Qmsg msg#

This command copies the given message file to a file called QMSG, stripping off the R: lines, and inserting > at the beginning of each line. It then takes you into the editor to edit the file QMSG. You may now respond to the quoted message text adding and deleting lines as desired. Editor help is available with Esc HELP. When you are done editing, save the file (Esc SAVE). Next go into the bbs and use the REP command to begin replying to the message. Instead of typing your reply, use the \+\QMSG to include the edited quoted message. When it has been read, type the ^Z or /EX to signal the end of your message. If the command is done from within the bbs by using the form /QM, the bbs will prompt your entries for sending the message.

QUIT

After you use QUIT command (or Ctrl/F4) the screen is cleared. JK etc. information is saved when you do a QUIT and reloaded when BBS is started again.

RBid [on/OFF]

When on, will take the received BID/MID ($:xxxxxx) and insert it in the current R: line.

RDir directoryname

REMove specified directory

Readbackup number

This command allows you to read a message in the MAIL.BAK directory (if it exists). Example: RE 12345

REMove DESt callsign

Deletes specified destination network node through all neighbors.

REName Oldpathname newpathname

Renames a file. Both name must be on same drive.

REPlaces

Reads BBS call/@BBS replace file and stores it in memory. Use this command every time you modify the MSYS.REP file while MSYS is running.

REQDIR [on/OFF]

Enables Request directory server in the BBS.

REQFIL [on/OFF]

Enables Request File server in the BBS.

REQMaxfile number
Used to set a limit on the size of files that may be requested from the REQFIL server and also the files that will be displayed in REQDIR requests. The default is 5000 which is probably too high.

REQQTH [on/OFF]
Enables Request QTH server in the BBS.

RESPonsetime # Where # is the number of seconds to wait before sending out a response. This is the T2 timer value in the AX.25 spec. If you set RESP to 0 it will work the old way, always sending out a response to each I frame received as soon as possible. With non-zero values for RESP, the responses are held, the latest one replacing any one being held, until no I frames have been received for the time given. It then sends out its response. RESPonstime Can be set for individual ports.

RESTore <arguement> Restores a message from the MAIL.BAK dir to the active mail directory specified by the arguement. IT DOES NOT maintain the old number, but assigns a new MESSAGE number. It WILL maintain the original BID (if any).
Possible arguements:

msg#	restores a given message number
msg# msg#	restores all messges in given range of numbers
Missing	restores messages that exist in MAIL directory but don't have headers.

NOTE: This command may now be done as a SYSOP command OR from within the BBS. BBS Example: /REST 12345

RETries number
Displays or sets number of retries before disconnecting. A value of 5 is good for VHF, 10 can be used on HF.

RLinetime [#]

Given number of hours are added to the computer clock to generate date/time in R: lines added to messages. Used to correct R: line time to "Z" time. MSYS remains on local/computer time. Mutually exclusive with GMTimeoffset.

RUn programname

Allows you a DOS program with many limitations. For those of you who have asked for some sort of DOS shell capability you may recall I have said it can't be done. But some of you said all you needed to do was output 8 bits to some port or something like that. Well if you have something VERY SIMPLE and VERY SMALL and very QUICK here is something that should work for you. Note: You must follow all of the rules given here or very strange things might happen for which I claim no responsibility. If you use the RUn command and it doesn't work it is clearly you who have made the error! Here are the rules for programs you might consider using with the run command (if you don't understand these rules or aren't sure your program complies with them, don't run it!):

1. Program must be small (I'd say under 10K). If you get a message that says not enough memory, the program is too big.
2. Program must be in default MSYS directory.
3. Program must be fast. If runs much over 1 minute, the system will reboot. While it is running no packets of any kind will be transmitted.
4. Programs should not use any of the "standard" pre-opened files. Console I/O must be done using ROM BIOS only. If you open any files, you must close them. If you use any memory, you must free it. If you change anything in the computer you must put it back for the most part.
5. Any screen output will be put in a file called $POOL. The TYpe command may be used to view the file.
6. You should write your programs in either assembly language or C. Here is an example of a C program that is suitable for framing (I mean execution using the RUn command):

```
#include <stdio.h>
#include <stdlib.h>
main(int argc, char *argv[])
{
 int I;
 char *path;
 path = getenv("PATH");
 for (i=0;i<argc;i++)
   cprintf("argv[%d] is '%s'\n",i,argv[i]);
 if (path) cprintf("Path = '%s'\n",path);
 exit(1);/* You didn't expect this to do anything useful I
hope*/
 }
```

SCreensave number
     If the number given is zero (which is the default) the screen will not be blanked automatically. The method used to blank the screen will work with all display cards (the screen content is saved and then blanks are written to the screen). When you press a key to unblank the screen you will always get to screen 1 (the one you get with the F1 key).

SEnd <chan#> <message>
     Parameters are an active channel number and the text of the message to send. The text is prefixed with "MESSAGE FROM SYSOP AT id-callsign". Use this to give helpful hints to users you see that are in trouble with the commands they are trying to use. If you wish to NOT send the "MESSAGE.FROM..." verbage, use the form: SE- <chan#> <message>

SHUTdown [on/OFF]
     When on, will send shutdown in progress message to users. Does NOT affect bbs connects or forwarding. Forwarding should be ABorted and incoming bbs connects and network connects should be DISConnected on a per channel basis.

SLottime [port#] value
     Sends the given value (0 to 255) to the TNC. This value is used in conjunction with PPersist. For more information consult your TNC manual.

SMTP
     Allows local access to the SMTP server (mainly for test purposes). The data channel port number for SMTP transfers is not correctly displayed. It does not affect operation and may be fixed in a future release.

SNow [on/OFF]
     If you have a CGA adapter and there is a lot of snow during screen changes (particularly when changing display pages with F1-F4) and you don't like it, set SNow to ON. Otherwise leave it off and the display will run faster.

SOrt filename
     Useful to sort your forward include files in the /fwd directory so you can find things there more easily. Will sort up to 200 eighty byte records.

STatus <option>

With no option, it displays the status of active channels, buffer space/errors on serial ports and dispatches per second.

OPTIONS:

RESET	Clears counters, sets dispatch values to last minute value.
Channel	Displays only information on active channels
Ints	Displays information about interrupts received
Stats	Displays only buffer space/errors on serial ports and dispatches per second info.

I put this one in for my own debugging (like some of the other information displays). FREE is what is currently free in the buffer used to get characters from the TNC (an interrupt driven routine is used). LEAST is the smallest amount of space that was available. If least is small (say less than 100) you've got some real problems. I don't think I've ever seen it go below 3000. If it is small it means that characters are coming from the TNC faster than they are being processed. You might try a lower baud rate between the computer and TNC but this probably won't really help the situation much. Dispatches per second is how many times per second each process was given control of the CPU. If this goes to zero I'd worry a lot! Note: will only display information for real ports.

SUspendtrace number

Where number is the number of free bytes in the input buffers below which the monitoring is turned off. There is 1 buffer of 4096 bytes for each serial port (tnc). The amount free at any time is given by the STatus command. When the free space gets less than the SU value, F2 monitoring is turned off to allow things to get processed more quickly. Default value is 2000 bytes. Again, F2 displays take a lot of time and when the system gets behind it is a good idea to turn off the updating of the display. When the free space in all the buffers goes above the SU value tracing is turned back on (unless otherwise suspended). This function can be effectively disabled by setting SU to 0 and then it will work as before.

SYsop [ON/off]

If on, messages to SYSOP will be included inthe "You have ... messages" on the F1 screen display.

TABs [ON/off]

when on tabs are expanded on local display; when off a tab shows up as ^I. Default is ON.

TACk [number]

Transport ack delay in seconds for network node. Default is 3. Similar in function to RESPonsetime but for network node.

TBdelay [number]

Transport busy delay in seconds for network node. Default is 180.

TElnet host-id [server#]

Takes an IP address (or host mnemonic) and optional server number as parameters.

Examples:

```
TE [44.70.4.10]
te eiw    (eiw would have to be defined in MSYSHOST.NET)
```

NOTE: To allow Telnet access to your BBS, you must modify the entries in your MSYSHOST.NET file. Basically you must add the call (without SSID) in square brackets of those stations that are to be allowed BBS access. Example:

```
44.70.4.5 eiw K8EIW      #this IP address would not have access
                         to the BBS
44.70.4.32 [k8eiw] don   #this IP address would have access to
                         the BBS
```

Note: You can use any case within the [ ], do NOT use any SSID's!

TERM [port#]

This command connects the keyboard directly to the port so that you may access a tnc or modem. It may be useful for resetting parameters in the modem or tnc. There are two F-keys that may be used while in this command. The F9 key will send a sequence which should take a tnc out of the KISS mode. The F10 key exits the term mode. When leaving the term mode, make sure that tnc's are in the KISS mode, or <PACTOR/GTOR STANDBY> mode. Modems should be left in the auto answer mode. While in the TERM command, the keyboard can not be used for other functions within MSYS (F6 key), nor will input from the port be visible on the F2 screen. There is no recording or sending of included files (\+\)and there are no plans to add them.

TFrames Port#

Send test frames on a given port each time enter is pressed. Use Esc key to terminate.

TImezone [string]

Allows entry of your time zone. Set timezone GMT to all caps to get lower case z on message time R: lines.

TNatime [number]

Network node Transport no activity timeout time in seconds. Default is 600.

TOdo
Causes the MSYSTODO.DAT file to be scanned immediately, doing anything it is time to do and also setting the time of the next thing to do.

NOTE: Be sure to use this command every time you change MSYSTODO.DAT while MSYS is running!

TPorts [MASK]
This number is used the same way as other similar commands: When the bit is set to 1 for a given port, that port is available for TCP/IP use (particularly ARP broadcasts).

Trace25 [ON/OFF]
Enables/disables traces of AX.25 packets. If you set it on you will get more stuff that you can't understand faster than you ever wanted to see it! Mainly used for debugging of AX.25 protocol problems.

TraceIP [ON/OFF]
Enables/disables trace of TCP/IP problems. Again mainly for debugging purposes!

TREtries number
Network node Transport retries. Default is 3.

TRUncate filename
Adds CR/LF to files as needed to make lines no longer than 80 characters so it can be edited with ED command.

TS HHMMSS
TimeSet command - sets DOS time to given value (must be exactly 6 digits)

TTimeout number
Network node Transport timeout in seconds (FRACK for the network). Default is 60 seconds.

TXDelay [port#] number
Keyup delay (usually in 10 mS increments) number given can be between 0 and 255 (its best to err on the high side). This number is sent to the TNC. For more information, check your TNC book.

TXTail [port#] number
Keydown delay - how long to wait after sending a packet to unkey the transmitter. The number can be between 0 and 255. This command should be used with TNC-2 clone TNCs (any tnc that uses hardware switches to set baud rate going to radio) when on HF. Without this parameter the TNC keys up on the transmitter before the complete packet is sent. Try a value of 20 as a starting point.

TYpe filename
     Allows you to type a file. Use spacebar to stop/start and a control Z or A to abort.

Users filename
     Writes out all users info to formatted ASCII file

VAlidmsgtype [arg]
     The argument is a list of letters that may follow S in send commands in the bbs. Defaults are BTPW.

VErifysysop [ON/OFF]
     If on then whenever any variation of the id call is used to connect to the BBS (except from the local keyboard) then 4 characters from the REMOTE SYSOP phrase must be entered.) NOTE: THIS MAY NEED TO BE SET TO PREVENT UNWANTED ACCESS BY LESS-THAN-HONEST (AB)USERS. The REMOTE SYSOP phrase is in a file called PASSWORD.RMT in your FWD directory that is the same as the MSYS.RMT file on the other system.

VIdeo   [COLOR/MONOCHROME]
     You can change the monitor you want to use while the system is running (using COLOR or MONOCHROME as a parameter). I've got both connected to my systems here (at the same time) so it is meaningful to switch back and forth on rare occasions (like to see if it works). You probably ought to specify the one you want to use in your MSYS.OPT file and not otherwise use this command.

WAtch chan#
     This command allows you to see all that occurs in both directions for the channel you specify. Watching is turned off when the channel is closed or you use the WAtch command a second time for that same channel. The lines that go to the screen are not labeled in any way, they are just sent to the screen. If you WAtch more than one channel at a time it will work but you may wonder which line came from which channel. Using watch may have some side effects like resetting the window being used by the process using the channel you have selected. You should probably not "watch" a forwarding channel. That's a "feature" of this program and will not be corrected!

WBackgrnd number
     This command requires two parameters to set back- ground color: Window# and Color#. The color number should be between 0 and 7.

WColors
     Gives list of color names and their numbers

WForegrnd number
This command requires two parameters to set fore- ground color: Window# and Color#. The color number should be between 0 and 15.

WHO
Displays current users and messages.

Window
Use of this command is a good way to see what numbers mean in the various window definition commands.

WHY msg#
This command will search for the words in wordfile or hold file in the given message#. Useful when you do a LH and want to see why the message was held.

WPAge number number
This command requires two parameters to set page number: Window# and Page#. The page number must currently be between 0 and 3. Basically this command defines what page a window will appear in. Example: WPA 5 3
would assign window 5 to page 3 (the one you see when you press the F4 key). This example assignment, by the way, will not be particularly useful!

WPEvery number
Specifies the number of days between sending out WP information for users who haven't changed their information. Default is 60 days.

WQupdate number
Worst quality network node destination node route to record. Default is one.

WRecord [number]
Window record sends all output to a disk file. The first time use the command for a given window it begins recording everything sent to that window in a file called window.dat where N is the window number. The second time you use the command for the same window it turns off recording for that window. Using the command with no operands displays the status of recording. NOTE: This function produces a lot of disk output quickly. It can fill up your disk. It also slows down the system a fair amount due to all the disk writes. This command was added for debugging purposes and should not be used as a general purpose capture command. You can determine the window numbers by using the W? command.

WSize number number number number number
     Specifies coordinates for specified window This command requires 5 parameters. The first is the window# being defined. The remaining parameters are two pairs of numbers giving the upper left and lower right corners of the window. The pair is column number (1,80) then row number (1 to 25). Overlapping windows may produce amusing results. To specify window 3 as being the entire screen the following command may be used: WS 3 1 1 80 25

W?
     Use of this command is the easiest way to find out which window is which! It displays the window number of each window in the upper left hand corner of that window.

XBatchsize [port] #
     Sets number of bytes of uncompressed data per batch for RLI compression on port. The value defaults to 0, which means only one message will be forwarded per batch.

XOption [port#] [on/OFF] If on, enables RLI compressed forwarding
          on port

Yapp [port#] [on/OFF]
     Enables use of Yapp (binary file transfer between users and the BBS) on selected ports.

<Esc>
     Cancel partially typed command

Function key usage:

F1 through F4 select page to display (0 to 3 respectively)

F6 causes the following characters to the next return to be treated as a command even if keyboard currently connected to some other process (such as local BBS, etc.).

F8 key can be used to blank the screen. Pressing any other key will unblank the screen. A new SYSOP command is also added that allows you to specify the number of minutes after last keystroke to blank the screen automatically:

Alt-F1 Alt-F2 Alt-F3 keys work like DOS F1 F2 and F3 commands to retrieve/edit last SYSOP command typed.

Ctrl/F2 suspends activity on the monitor screen. ^F2 restores normal operation. This is probably as close as being able to scroll the F2 screen that I will be implementing.

Ctrl/F4 terminates MSYS as if QUIT command was used

Ctrl/F5 is the same as the PROcesses command

Ctrl/F6 assigns keyboard to command processor

MOST COMMANDS GIVE INFORMATION ON THEIR
USAGE IF YOU TYPE COMMAND NAME FOLLOWED BY A
QUESTION MARK.

Example: WS ?

MOST COMMANDS GIVE THE CURRENT VALUE SET IF
YOU TYPE JUST THE COMMAND NAME.

Example: FRAck

MSYS BBS USER COMMANDS

A - Abort	P - Path to station
B - Bye	PC - Call Sign Server
C - Conference	R - Read message
D - Download	S - Send Message
G - Search file	T - Talk to SYSOP
H - Help	U - Current users
I - Information	V - Version
J - Calls heard	W - What files
K - Kill message	X - Expert-Mode
L - List Messages	Y - YAPP Command
LC - List Catagories	?x - Info about command
M - Message of the day	* - Comment line
N - Enter name/qth	

The A (abort) command to the BBS. It can be used to abort output from most BBS commands, like Download, List and Read.

The Bye command disconnects you from the BBS. Use it when you are done!

The C command enters the user into the Conference. Following lines typed will be sent to all stations in the local conference. Control-ZQ (^ZQ) or /EX may be used to exit the conference. When a station enters or exits the conference, a message noting the event is sent to the stations in the conference. Other Conference commands are:

^ZH produces the help text
^ZU Lists the current BBS users
^ZA # Asks the user on the given channel number to join the conference

The CC command allows sending a "Carbon Copy" of a given message to other stations. Only the originator of the message, the destination station, or the SYSOP can do this for a given message. Syntax is: CC msg# call1@BBS1; call2@bbs2 ... The @bbs parts are optional if the calls are users of your system. Example: cc 4725 k8eiw@wb8bii; no8m An alternate syntax for this command is: CC msg# <filename Where filename does not include a file type (.DIS is automatically appended). This file contains tocalls (and optionally @BBS) of the stations that should get the copies. There is one call per line of the file. Example:

```
CC 3195 <locals
The file LOCALS.DIS could contain:
k8eiw @ wb8bii
NO8M
```

The Download command is used to read a file stored on the system. The format of the command is D filename where filename is the name of the file to download. To see what files are available for downloading, use W or WN command. To get a file from a subdirectory type:

```
 d <directory_name>/<filename)>
```

The G command (for grep, a Unix utility) allows you to search a downloadable file for a specified string of characters, printing all lines that contain the string of characters. Case is ignored in the comparison. No wild cards (like ? or *) are recognized. When you use this command you will be asked for a file name and then a search string. For file name enter the name that you would use if you were downloading the entire file. If you do not give a file name, Database.0 will be assumed. When the command is used, the user will be asked for the file name to search and then the search pattern.

The default directory FILES\ will be prefixed to the file name the user gives UNLESS the DOS SET command has been used for filesdir. Thus downloadable files may be pattern searched, and in the directory that the sysop chooses to make available.

The G command is not available when the BBS is in limited service mode.

The GX command permits search on the directory specified by the DOS SET msys:grepdir (default is FILES).

SEE ADVANCED USERS SECTION for details.

The H command will give you a short description of the commands. For more information about a particular command, type ?x where x is the letter of the command.

The I command by itself gives hardware configuration of this system. (the information displayed comes from HELP\INFO.HLP)

For example:

This system runs on an IBM PC type computer and is written in C.
It uses TNCs running in KISS mode which allows all the fancy stuff it does.

Location is Port Huron, Michigan
TNCs KPC-4
PK-232
Frequency Radio Power Antenna
145.03 MOCOM 70 60 W Isopole at 80 ft
145.09 TR-7400A 30 W Isopole at 30 ft
If you have any questions please leave a message.

ID gives a list of the ports and digipeaters/gateways available.

JH command lists stations recently heard on the various ports and stations that recently connected. Use the P command for path to station that have connected at some time. For the following commands the sponsor's callsign is given in {curly brackets}. This is normally NOT the call you would use. This callsign is given to help identify some of the "mystery" callsigns used for nodes and digipeaters such as "MYDIGI".

JBbscommand gives a list of BBS's that have been heard along with date and time they were heard and the path to them.

JDigipeat command gives a list of the Digipeaters that have been heard along with the date and time they were heard and the path to them.

JGateways command gives a list of the Gateways that have been heard along with the date and time they were heard and the path to them.

JKanodes command gives a list of the KAnodes that have been heard along with the date and time they were heard and the path to them.

JMsys command List other MSYS systems that have been heard.

JNetrom command gives a list of the NET/ROM nodes that have been heard along with the date and time they were heard, the real callsign of the NODe and the path to them (most are direct).

You can specify a port number as an argument to limit the display to just that port. For example:
JK 1 would display the KA Nodes heard on port 1 only.

The K command is used to kill (delete) old messages from the system. You can kill only those messages that are to or from your station. The format of the command is K, a space, and then the number of the message to delete. Multiple message numbers separated by a space may also be killed.

Example: k 123 6789 1123.

KM - To delete all messages TO you.

KF - Kill messages that have been forwarded (sysop only).

KT - Msg# to kill NTS traffic you are going to deliver.

The following "K" Commands Are SYSOP only:

K* - Goes through ALL messages, allowing you to kill, keep or hold any of them. You can also read a given message (it must be in the mail.bak directory). When reading a message this way, use the space bar to pause the display.

K@ - kills all messages with given @BBS. only compares the leftmost part of the @BBS field (the normally visible part of a hierarchical address) instead of any part which caused unexpected deletions.
Example: K@ ARRL

K> <call> or K< <call> will kill all messages to or from a callsign or "group".

L - The List command lists selected message headers. The following formats are available:
L - List messages since you last used the B command
LB - List bulletins (all of them, use with care!)
LC - List Catagories ("TO" fields)
LM - List messages to or from you (List Mine)
LN - List messages with type of N (List Not read or forwarded)
LO - Lists msgs older then date given as YYMMDD Ex.:
LO 910204
LT - List NTS Traffic messages
LL # - List the last # messages Example: LL 10
LU - Lists unread messages to you
L< callsign - List messages from callsign Example:
L< WA8BXN
L> callsign - List messages to callsign Example:
L> WA8BXN
L@ callsign - List messages being sent to BBS call Example:
L@ WA8BXN
L # - List messages >= given number Example: L 827
L # # - List messages from # to # Example: L 50 75
L"string" - Lists messages with given string in title (case insensitive)
L'string' - Lists messages with given string in title (case insensitive) Examples:
L'Pfb'
L"ID's"
l"Modem users"

Note: When asked for msg#'s to read, always use the indicated number of digits. Use leading zero's if needed to get the requested number of digits! Example for a 5 digit message number: 00123

The following are sysop only:
LF - Messages that have been forwarded but not deleted
LH - Messages marked as Held (invisible to all users)
LY - Messages that have been read but not deleted
L$ - Messages with type $
L$$ - Messages with type $ (also gives their BIDs)
L$ # # - Messages in range also giving their BIDs
LL$ # - Last # messages, showing BIDs.
L# - Bulletins that have been completely forwarded(type #)
L? - Messages that MSYS doesn't know how to route

The L< and L> commands will also take wildcards as arguments.

Note: Messages that are Private are shown only to Sender, Addressee or SYSOP.

If the listing from the L command exceeds the page limit you have set with the X command, it will permit you to enter the last 2,3, or 4 numbers of the message numbers that you wish to read. Upon completion of the displaying of the desired messages, it will continue with the listing of messages.

The M command may be used to display the Message of the day.
Normally this message (if not empty) is displayed when you connect unless you have selected Expert mode (?X for more info on Expert mode). If you have selected Expert mode, using the M command is the only way to see any Message of the day. If there is no message of the day, you will simply get the next command prompt. The message of the day will advise you of new features and other important information. Example: M

N command can be used to register your name or QTH. You should enter both of these. To enter your name type N your name.
Example: N Mike

To enter your QTH, use the command NQ your qth.
Example: NQ Kirtland, OH

To enter your ZIP or Postal Code, use NZ code.
Example: NZ 44026

To enter the primary BBS that you look for your mail on use NH callsign.
Example: NH WA8BXN

The P command can be used to find the path last used by a station to connect to this system. The format is P callsign.
Example: P W1AW
The information given comes from the station's use of the N commands and information recorded by the system. Try the P command with your own call to check the information the system has about you.

PC <callsign> Will return the callsign information from the callsign database (if enabled).

The PF (bbs_callsign) command can be used to find out if this system knows how to forward mail to a particular BBS (what you would enter as @bbs in the S command). The port number and call of the BBS to which the message would be sent is displayed along with the VIA list used for the connect. If NetRom or KA Nodes are used to connect to the BBS to which the message would be forwarded to the fact is noted but the details are not given. If this system does not contain the requested BBS in its forward file you get a message that the callsign is unknown.

Example:

```
PF WA8BXN
```

The R command is used to read active messages in the bbs. The following are variations of the R command:

R # # ... Reads the given message numbers Ex: R 12734 11521
RM Reads your messages (Read Mine)
R catagory Reads the messages in a given TO" field. Example: R SALE
R> callsign Reads msgs with given "TO" field
R< callsign Reads msgs from given callsign
R@ callsign Reads msgs with given @BBS Example: R@ ARRL
RE # Read Export This causes the given message(s) to be displayed with a S command line, followed by title, message text and then /EX in place of the normal way messages are displayed. If this form of output is saved using the output to file command [< filename] you can easily export messages to an ASCII file.
RH <msg#> Gives complete forwarding information including R: header lines.
RP Read Preview. It does a normal read operation but does not mark the message as having been read so it will show up if you do a LU command.
RN <msg#> This is like the R command but no headers are printed, just the message text. Good for marginal connections when you otherwise know the message number you want to read.
REM Read Export Mine. Reads all messages to you in export form

The following R commands are SYSOP only:

RS Reads all mail for SYSOP. Note that when you read mail for SYSOP (ALL CAPS) it gets changed to mail for Sysop so that it won't continue to show up as a new msg.

RES Read Export Sysop. Reads all messages to SYSOP in export form)
R* It allows you to read all held messages. After each is read you are prompted to enter H to continue to holding the message, K to kill, N to release as private, or $ to release as a bulletin.
R+ Goes through all the held msgs using the hold file trying to identify why each was held and allowing you to continue to hold, kill, or release as private or bulletin each message.
R- This is an abbreviated form of the R+ command, and is useful on HF or slow network channels.

Typically, after a screenfull of lines is displayed on your terminal, you will be asked:

More? [Y]es, No or Continuous

Just press return for next screenfull, N to quit reading this message or category), or C to display all the rest without pausing. The number of lines per screen is set by the X command.

REPly

The REPly command is used to generate a reply to a message without having to do a S command. The advantage of using the REPly command is that it will usually pick up the correct @BBS (often with proper H route) from the R: lines in the message being replied to. You can use either:
REPly msg# (example: REP 1234) or
REPly msg#. (example: REP 1234.).

In the first form you will have to supply the title. In the second form, a title is automatically generated of the form RE: the-original-title Messages that passed through SMTP systems will probably not generate the correct reply address.

The SR command can be substituted for the REPLy command, and will perform the same function. If you put a period after the msg #, a title will automatically be made by preceeding the original title with RE:

S command Format: Sx tocall @bbs $BID#

Where x is message type (P=private, B=bulletin, T=traffic)
If the x is omitted, and the MAkePRivate is on, the message will automatically be made a type SP
Tocall is DEStination callsign (without - number/SSID)
@bbs is DEStination BBS (again without any - number) [OPTIONAL]
---> If @BBS part is omitted, if the station has entered a Home BBS (NH command) that BBS will be used. Otherwise the to station callsign will also be used as the @BBS.
$BID# is the Bulletin IDentification Number [Valid ONLY for type B - OPTIONAL - if omitted, MSYS will assign if needed.]

Examples:

SP call @ bbs.haddr	SP K8EIW @ WB8BII	Send a pvt. msg
ST zipcode @ NTSst	ST 44070 @ NTSOH	Send traffic msg
SB catagory @ route	SB NEED @ ALLOH	Send a bulletin
SR msg#	SR 12723 or SR 12723.	Send reply.

You will next be asked for a title for the message (except for the SR 12723. form). After this you type the message, keeping all lines less than 80 characters. When you are done with the message type ^Z or /EX on a new line.

Never use SSID's in callsigns (ex: don't use WA8BXN-2, just use WA8BXN)

Hierarchical addresses (haddr) are of the form:
state.country.continent.
Other sub-areas may be used by BBS in a particular area, such as: #NEOH.OH.USA.NOAM. Generally, the bbs will be aware of the full hierarchical address of a bbs (from the BBSLIST.DAT file).

When sending personal messages to a distant bbs, it is a good idea to put the destination city and state in the title.

If you type ^A in place of ^Z when you are done typing a message, the message will not be stored.

Users who send messages to stations without giving any @BBS or giving unknown @BBS are warned of the fact.

Use SB to send bulletins, ST to send traffic, etc. All messages sent to a specific callsign will be treated as if you used SP (send private). To make messages to a specific callsign readable by everyone use SB instead of just S! Or make it to a non-callsign (such as ALL). If you are sending a bulletin that may also be entered on other BBS, use the form:

SB <to> @<bbs> $<BID#>

so that a proper BID# will be assigned.

The SR <msg#> command functions identical to the REPLy <msg#> command.

The ST Command will NOT add a BID to messages.

Notes about the @BBS:

Automatic generation of @BBS field for S command in BBS: If the S command contains an @BBS entry it is left untouched. If there is no @BBS as a parameter for the S command, the following happens. If the to station is found in the user file then the home bbs of that station is substituted for the given @BBS. If the to station is not a known user then the to station callsign is placed in the @BBS field.

Any messages entered into the system with no @BBS that don't get a @BBS from the user file will have ==== for the @BBS. Only messages with second type character of N will cause the user file to be searched for @BBS. Thus bulletins will never get their to call sign (typically ALL) looked up in the user file.

In the bbs, if a S command has an SSID in the @BBS filed it is removed in the leftmost part of the hierarchical address. Thus S W1XYZ @ K1XXX-5 is taken as S W1XYZ @ K1XXX.

T The T command (talk to sysop). When a bbs user types T you will get the connect music and then have 60 seconds to type something or they will be put back to the bbs. If you do type to them then when you are done with your conversation and do ^Cd then they will go back to the bbs. The keyboard call can still be used for incoming connects; note that it is accessible to all stations, even on closed bbs ports. Outgoing connects are the same as they have always been. When an incoming keyboard connect occurs, you will automatically be taken to the F1 screen If you wish to prevent users from ringing your bell, set MCON 0.

U The U command gives the current users of the bbs plus the number of active messages and users and what they are doing.

UPload

The UPload command To use it, the user must be authorized (see user flags). To do an upload, the user types UP on the BBS. MSYS will then ask for the file name to be uploaded. If no file name is entered, or the file exists (in the FILES directory) the upload is terminated. If the filename is accepted, the user then sends the ASCII file, ending with a line that contains only ^Z. Once the transfer is started, it may be aborted by sending a line with ^A.

Version

The V command gives the version of this software and date and time MSYS was started.

W The W command lists the available files for downloading.
Examples:

```
W    Lists all files in C:\MSYS\FILES
W (Dir1)  Lists all files in the (Dir1) subdirectory of
          C:\MSYS\FILES\(Dir1).
```

The directory list provided includes the dates & sizes of the files.

X Command can be used to set or reset Expert status and also to set the number of lines sent before pausing during an R command. If no argument is given, X simply toggles and displays the new Expert status. When Expert status is set, the initial connect messages are replaced by a compatibility identifier line ([MSYS-1.14-BFH$]) and the command prompt is simply >. If a number is given after X, this is the number of lines to display for a R command before asking More? To continue displaying the current message when More? is sent, press return. To abort the message, type N and then return. This facility has been added to prevent a long message from scrolling off your screen faster than you can read it. Experiment with different values until you find one that pleases you. To not use this facility, set the number of lines to zero. Examples: X 20 (pause after every 20 lines) X 0 (don't pause ever). When you get More? in R cmd output (as a result of using X cmd to set lines between pauses), you can type C for continue without any more stops.

XC

Toggles the automatic display of msg catagories when you connect to the bbs and are using non-expert mode. Default for new user is ON

XF and XS Commands

XF sets "fast" mode, XS sets "slow" mode. Fast mode means that multiple lines are placed in packets for some commands (like D, R and L). Slow mode means one line per packet as it has been (the default). XF is faster for users with good paths. XS is better for users with poor paths (or busy channels!).

XR

Toggles the automatic asking if you want to reply to a msg after you read it. Can be useful if you get a lot of personal mail. Default for new user is OFF.

YD pathname filename

YAPP download of the given file. Users must start their YAPP function within 30 seconds of issuing the command.

YU pathname filename

YAPP upload of the given file. Users can not delete or replace existing files. Users must be enabled with user bit $800 to be able to upload files.

YW List files in the default YAPP directory. (See section on DOS SET for further details on defaults.)

YW xxx List the contents of the xxx directory under the YAPP directory.

* Putting * at the beginning of a line makes it a comment. It also suppresses the next command prompt (but the system will be waiting for another command). * is useful to answer the SYSOP if you get a: MESSAGE FROM SYSOP

< filename (sysop only)
Causes the commands found in the given file to be processed as if they were typed from the keyboard. This may be used as a message import function if messages are placed in a file. A sample is:

```
SB ALL @ ARRL < W1AW $XYZ27
Sample title
This is the text of the message. It is short.
/EX
```

You can of course have many messages in the file. When end of file is found, input is taken again from the keyboard.

> filename and >> filename
Both of these commands cause output normally sent to the screen to be also saved in the indicated file. Using just one > causes a new file to be created (replacing any existing file by that name). Use of two > before the filename causes output to be appended to the end of the filename given (if it exists). To terminate writing to the file, use either of these commands without a file name. While output is being saved to a file, there will be NO PROMPT FOR NEXT COMMAND.

/<cmd> To be used for sysop commands while in the BBS (see remote sysop) or:
\<cmd> To be used for sysop commands while in the BBS (see remote sysop) Either symbol works!

A set of REQxxx servers are available in the BBS they are: REQDIR, REQFIL and REQQTH. Messages sent to your system to REQanything will be processed. If a server is not available for the requested function, a reply message will be generated stating that the server is not available. The servers named above are enabled by new commands of the same names. That is, using REQDIR ON will enable the REQDIR server, etc. Enabling the REQQTH server also requires that a file name containing the Buckmaster CD ROM ASCII callbook file (normally S:HAM0\HAMCALL.129) be specified with the CAllfile command.

The REQDIR server responds with a message to the requester containing either the FILES directory listing or specified FILES directory subdirectories given in the message title. An @BBS to which the message is to be returned may optionally be given in the title if the reply is to go to a bbs other than the one at which the original message was entered. Examples:

```
SP REQDIR @WA8BXN
   (blank title)
^Z
```
would return a listing of the FILES directory to the bbs used to originate this message.

```
SP REQDIR @ WA8BXN @NT8V  (title)
^Z
```
would return a listing of the FILES directory to the sending station @ NT8V.

Subdirectories may be given in the title, proceeded by a slash and separated with blanks. The optional @BBS may also be specified. Example:

```
SP REQDIR @ WA8BXN
/MODS /PGMS/BASIC
^Z
```
The file REQDIR.HLP included on the distribution disk should be placed in the HELP directory. The file REQFIL.HLP should be placed in the FILES directory.

The REQFIL server responds by sending the requested file. The file name is specified in the title of the message. Subdirectory names may be given if needed (don't include the leading /). An @BBS can be specified in the title as described for REQDIR above. Example:

```
SP REQFIL @ WA8BXN
mods/kam.1 mods/kam.2
^Z
```

The REQQTH server responds with callbook information for calls given in the title. Currently only US hams are in the callbook file. Calls must be separated by blanks. An @BBS may be specified as described above.
Example:

```
SP REQQTH @ WA8BXN
nt8v no8m K8EIW
^Z
```

CALLSIGN SERVER

The callbook server has split out of MSYS. There are three servers available, one for the Buckmaster CD ROM, one for the RT Systems SAM database, and one for the j-com disk files. Note that the CAllfile command has been removed from MSYS. In its place you specify the name of the file (or directory in case of j-com files) to be used when you start the appropriate call server TSR.

To use either the Buckmaster CD ROM callsign files, or those available from j-com, and SAM (c) RT Systems, you must load the appropriate TSR program before starting MSYS. Once this is done, accessing the call server is the same as it has been in the past (using either the PC command in the BBS or sending a message to REQQTH @ the bbs call).

The TSR for the CD ROM files is CALLSRV1. It optionally takes one pararmeter, the name of the callsign file. If none is given, the default is S:\HAM0\HAMCALL.129.

The TSR for the j-com files is CALLSRV2. It optionally takes one parameter, the name of the directory in which you installed the several j-com files. If none is given, the default is F:HAMBASE.

The TSR for the SAM (c) files is CALLSRV3. It optionally takes one parameter, the name of the directory in which you installed the several SAM (c) files. If none is given, the default is F:HAMBASE.

THINGS NEEDED TO MAKE THE CD-ROM DATA BASE CALLSIGN SERVERS WORK:

Put the following in your CONFIG.SYS and reboot system.

```
LASTDRIVE=W
DEVICE=SONY_CDU.SYS /D:MSCD210 /U:1 /B:340 /M:P /V /L:S /Q:*
/T:*
```

Run the following before MSYS (probably put these statements in AUTOEXEC.BAT):

```
MSCDEX /D:MSCD210 /M:4 /L:S /V
MSCDINFO
CALLSRV1 S:\HAMO\HAMCALL.129
```

Put the following in MSYS.DO:

```
     REQQTH ON
```

Notes: You may have to make changes in the above statements if you have different device drivers for your CD ROM drive or set up the hardware options differently. Running without interrupts works pretty well on the CD ROM drive and if you are like me you don't have any interrupts to give to the drive anyway. If you don't want to respond to REQQTH queries but only enable the PC command on the BBS, set REQQTH OFF.

For all the servers, the title must be less than 39 characters.

THINGS NEEDED TO MAKE THE SAM (c) DATA BASE CALLSIGN SERVER WORK:

Put the following in your AUTOEXEC.BAT file.

```
          E:\sam\api\samapi  E:\sam     This installs the SAM
               (c)                      API interface.
          C:\msys\callsrv3.exe E:\sam   This installs the MSYS
                                        interface to SAM (c)
```

Of course, this presupposes that you have already installed the SAM (c) data base on your E: drive.

Put the following in MSYS.DO:

```
     REQQTH ON
     PC ON
```

MSYS now supports the SAM optional County data base as well.

MODEM SUPPORT

MSYS will support one or more modem ports. The modems must be capable of supporting true carrier detect (CD) operation. Both the CTS and the DTR line must be connected!

MSYS modem forwarding protocol

When MSYS attempts to forward to another BBS, the following protocol is used for the initial logon. When MSYS sees the other system send "Call:" or "Callsign :" it will respond with two lines. The first is "echo off" and the second is the callsign indicated in the forward file. Next, when it sees "Password:" or "Password :" it responds with a line that is the password from the forward file. It then proceeds using the normal packet protocol lines.

Incoming Call Protocol

When carrier is detected from the modem, MSYS sends the line:

MSYS Modem Interface (c) Copyright 1994 Hub Computers Inc.

It then attempts to open MSYSMOD.MSG. If the file is found, its contents are sent to the modem. Next MSYS prompts with "Call:". At this point a human operator would enter their callsign value. An MSYS connecting for purposes of forwarding would reply with "echo off" which disables echoing by the called MSYS system and then the callsign value would be send on a second line. MSYS next prompts with "Password:" and expects the password to be entered that corresponds to the previously given call sign. These two words are then used to scan the MSYSPASS.MOD file to find a match. If a match is found, the third word on the line in the MSYSPASS.MOD file is used as the apparent callsign with which the user will appear to have connected to the system.

Here is an example of a normal user connecting into MSYS via modem:

```
MSYS                    User
-----                   --------
MSYS Modem Interface ...
[contents of MSYSMOD.MSG]
Call:
                        NO8M
Password:
                        camping
[from this point on it looks just like a packet connection was made]
```

Here is an example of a MSYS system forwarding into another MSYS system via modem:

```
MSYS                      Calling MSYS system
-----                     -------------------
MSYS Modem Interface ...[contents of MSYSMOD.MSG]
Call:
                          echo off
                          WA8BXN
Password:
                          BINGO
[normal packet protocol follows]
```

Use ONLY FBBC 0 and XO OFF on modem ports. Very Important!!!

The use of FBB style forwarding and/or compression on the modem port will usually result in buffer overflows, and cause serious consequences to the continued high quality performance of the system.

Note: To accommodate exchanges with FBB systems, use of control characters in the call and/or password should not be needed. The callsign must be prefixed with a period to turn off echoing from the FBB system. Here is an example MSYSFWD.DEF file entry

```
     F2 K8EIW ATDP5551212 .WA8BXN MSYS
```

The F2 could be R2 or P2 to ask for reverse forwarding or to always poll respectively. The 2 can be replaced by whatever port number your modem is connected to.

K8EIW is the call of the bbs we are forwarding to.

ATDP5551212 is the telephone number to call. The P could be replaced by T if tone instead of pulse dialing is desired.

.WA8BXN specifies the "callsign" value to be give. The period says to FBB systems to turn off echoing. The actual callsign value used by FBB is just the WA8BXN part.

MSYS is the "password" value that will be sent to the FBB system.

This example is for forwarding to FBB systems. When forwarding to a MSYS system the only difference is that the period is NOT used in the "callsign" field.

MSYS.OPT FILES

The name of the port MUST be (in capitals) "MODEM" Radiospeed must be set to 1200. Port speed must be set to the speed of the modem for a 300 to 4800 baud modem. For modems that can lock the port speed it should be set to at least 19.2k for a 9.6 or 14.4k modem.

Here is an example:

```
port 5 at $260 int 5 speed 19200 name MODEM radiospeed 1200 tnctype 2
```

MSYS.DO

The CTS option MUST be turned on in the MSYS.DO file. The following shows the proper command for turning CTS on for port 5:

```
CTS 5 ON
```

The CTS line must be connected in the EIA cable from MSYS to the modem!

There are a number of ways to set up the modem. Here is a file for a Zoom 2400 baud modem:

```
atz ---------- reset the modem to the power on defaults
atq1 --------- don't send responses
atm1 --------- speaker will be active until connect
ats0=1 ------- answer on one ring
```

The only problem that might show up with these files is that the "ats0=1" *MUST* be last in all the files. If it is another location and someone is calling in at the time of the initialization, the modem will answer and then ignore all the commands that follow and come up with an improper initialization.

One 2400 baud modem was found with a unique need. Register 27 on a small plug-in was found to control the V22.bis routine. If 27 was set to one (the default), then the modem would *NOT* initiate a V22.bis error correcting connect. The factory default was that 27 was set to never make an error correcting connect! The line "ats27=0" cured the problem. AND ... it was found on a very generic 2400 baud modem that this same register, which was not mentioned in the documentation, worked the same way. Check for register 27 on your 2400 baud modem!

Here is the MSYS.OPT string used with a V32.bis Zoom modem which is able to connect from 300 to 14.4k (with automatic fallback):

```
port 5 at $260 int 5 speed 19200 name MODEM radiospeed 1200 tnctype 2
```

Here is the MSYSTNC.2 file that then sets it up:

```
atz ........ reset
at&f ....... load factory defaults
atl0 ....... speaker volume low
at&c1 ...... DCD follows carrier
at&d2 ...... DTR signal setting
atq1 ....... no responses
atm1 ....... speaker on until carrier
ats0=1 ..... answer in one ring
```

This may or may not work with your modem. A book and a whole bushel basket of experimentation may be necessary. Without the book for your modem, your work will be cut out for you.

One of the HAYES commands is AT&V. This command will permit you to see the setup for the different registers in your modem. The following is the display from a working Hayes Ultra 96 modem:

```
ACTIVE PROFILE:
B16 B1 B41 B60 E0 L1 M1 N1 Q1 T V1 W0 X4 Y0 &C1 &D2 &G0 &J0 &K3 &Q5
&R0 &S2 &T4
&U0 &X0 &Y0
S00:001 S01:000 S02:043 S03:013 S04:010 S05:008 S06:002 S07:099
S08:002 S09:006
S10:014 S11:095 S12:050 S18:000 S25:005 S26:001 S36:007 S37:009
S38:020 S44:003
S46:002 S48:007 S49:008 S50:250 S97:030
```

It has been reported to us that the following string works well with the US Robotics Sportster modem:

```
AT&F&B1&H1&R2Q1M1S19=2X4S0=1
```

The detail the individual commands is as follows:

```
AT&F      Load factory (rom) settings into RAM
AT&B1     Fixed serial port speed
AT&H1     Hardware Clear to Send
AT&R2     Received data hardware (RTS) flow control
ATQ1      Supressed result codes
ATS19=5   Inactivity timer set to 5 minutes
ATX4      Dial tone detect, busy/no answer detect, etc
ATS0=1    Answer on one ring
```

DIP switch positions 1-7 off, and 8 on.

It should be noted that some of the above commands do not exactly follow the Hayes Command set.

There does not seem to be a problem initializing most modems with one long string as opposed to individual "AT" commands.

Use at your own risk! READ YOUR MODEM MANUAL!

MODEM USAGE AT 4.8, 9.6 AND 14.4k

Hardware flow control between a modem using 9.6k or above is critical. MSYS now hosts full CTS support. This means that the modem's port speed can be set to a speed higher than the maximum connect speed. When a connect is achieved, the speed between the modem and the computer remains at that higher speed no matter what the speed of the modem to the remote person connected. If a person checks in with an old 300 baud modem, the speed between the computer and the modem remains at what it was sent as. The modem will accept data until its buffer is full. It will then signal the computer, via CTS control, that it is full and will await the buffer emptying before it tells the computer to send more data A full RS-232 cable is necessary. The three wire cables that work in some TNCs will not work here.

NETNode USAGE

Access to a network node from the modem is possible. It is accomplished by adding the capitalized string "NETNODE" to the appropriate line in the MSYSPASS.MOD. USE THIS OPTION WITH CAUTION! An example follows:

wombat poobear WA8BXN NETNODE

A user designated with the NETNODE parameter will first be put into the MSYS NETNODE and will have FULL capability to access the packet network. For such a user to access the BBS, he would give the normal BBS command.

FORWARDING

This is the area of MSYS that requires the most study to properly establish.

MSYS can forward in any of several ways:

1. Standard AX.25 Non-compressed
2. FBB non-compressed style
3. FBB compressed style
4. RLI compressed style
5. SMTP - non-compressed
6. Via Modem - NON-COMPRESSED
7. Via Attended HF GTOR or PACTOR as well as AX.25 PACKET
8. To (or from) a file

MSYS can forward based on several parameters:

1. On hierarchical address of a BBS
2. On a unique "flood" route
3. On a limited basis to a component of a hierarchical address used as a bulletin distribution area
4. On a limited basis on the "TO" field as well as the "@BBS" field
5. On ZIP code for NTS messages

MSYS can forward at a specific time, or on timed intervals.

FILE SETUP

There are several files that must be set up for forwarding to happen.

The first file that needs to be set up is the MSYSFWD.DEF. This file is the script that MSYS uses to establish the connection to the distant bbs. It is also the file that is merged with the specific lists of catagories to be forwarded, found in the <bbs_call>.FWD or <bbs_call>.REV files.

Next, before any bulletins can be forwarded, a file called MSYSBBSB.DAT must be created. That file consists of a maximum of 32 lines (0-31) starting with a number on each line, followed by the callsign of the bbs that you will forward bulletins to. Please refer to the example in the distribution.

FORWARD FILE DESCRIPTION

NOTE: TO FORWARD BULLETINS YOU MUST SET UP MSYSBBS.DAT!

The MSYS forward file (MSYSFWD.DEF) is an ASCII file that can be prepared with any program editor. The file consists of groups of lines separated by a line beginning with at least 3 dashes. Each group of lines specifies what should be sent to a given BBS along with other information.

The first line has the following format:

Position	Content
1	F (The letter) If you substitute R for F (F0 becomes R0 WA8BXN) then it enables reverse forwarding request. P0 Forces a reverse forward poll - EXCEPT IN FBBC MODE!
2	0-6 The port number to use (a single digit)
3	A blank
4to N	Callsign of BBS (with SSID) optional:
N+1	A blank
N+2 to ?	Via list of digipeaters

Examples:

```
F0 W1XX V W2XX,W1XX-1
R2 WA0XYZ-2
```

When no nodes are being used, this first line is used as if it were a connect command to the desired system . When a NODe is used, the Via digipeater list will have no effect.

If a NODe (KA NODe or NET/ROM) is to be used, the next set of lines are a script of how to use the NODe(s). This script may be up to 25 lines long. Its format will be described below.

Next comes a list of callsigns of BBS systems that should be forwarded to the system given on the first line (the one beginning with F). There is one call per line. No SSID should be given. The callsign of the system given on the first line should normally be given in this list. Example forward file that doesn't use any nodes:

```
F0 W1XX
W1XX
W2ABC
W2XXX
----------
F0 W1XX V W1XYZ
W1XX
W2ABC
W2XXX
---------
F0 W1XX V K1XXX,W1XXX
W1XX
W2ABC
W2XXX
--------------
F0 W8XXX-12
W8XXX
WA8XXX
K7XXX
------------
```

Note: In the previous example we try three different paths to connect to W1XX. If we get the messages forwarded the first way the other connections will not be attempted.

```
F0 W1XX
!01-04,12,21-23          This will forward during the times indicated
W1XX                     (Use of this line is optional, but if used,
W2ABC                    must begin with a ! If you wish to
W2XXX                    temporarily not forward to a station, or if
----------               you wish to have bulletins not marked as
                         forwarded unless they are polled, use !25)
```

In the list of callsigns given indicating what BBS's msgs can be forwarded to a particular BBS if a second entry is typed on the line then the second entry must match some part of a message title to allow it to be forwarded. For example,

```
F1 W1XX
ARRL DX
----------
```

@ARRL will be forwarded only if DX appears in their title. Case of letters is ignored in the comparison.

You can use * in the forward file. If you have NTS* then all msgs with NTS as the first 3 characters of the @BBS will be forwarded. For example:

```
F1 W1XX
NTS*
----------
```

* used alone is treated specially. Only messages that have been on the BBS for more than about 2 days will be forwarded. This gives time for them to be forwarded by other specified connections.

IMPORTANT NOTE: In the forwarding file, use of * for the BBS names cannot be the first call and the first bbs in the list of BBB's that can be forwarded to through the given bbs MUST begin with a letter.

Example:

```
F0 WA8BXN
*         <---- won't work
WA8BXN
-----------
F0 WA8BXN
44*       <----- won't work
WA8BXN
-----------
F0 WA8BXN
WA8BXN    <----- will work
*
44*
-----------
```

Note: # is a wild card character. If you want to put #NEOH as an entry in your forward file, you will have to use "#NEOH (the " is an escape character that says the character that follows must match exactly and is not treated as a wildcard character as it normally would be).

(callsign).FWD/.REV Files

One way to reasonably manage large forwarding lists is to create a file for each BBS that you forward to. The file will take its name from the BBS that you will be forwarding to. For example, to establish the list of @BBS items to forward to WB8BII, you would create a file called WB8BII.FWD. In this case, the extension name is not important, but to simplify our explanation, we will use the extension .FWD. To help manage the lists, you would to set up a C:/MSYS/FWD directory. In this example, the file will be: C:/MSYS/FWD/WB8BII.FWD. In this file, you will list all of the different @BBS and hierarchical addresses you wish to forward.

If you wish the other station to also be able to poll you for the catagories, the file_name.ext must be <callsign>.REV.

Example:

File entry	Comments (NOT IN THE FILE!)
WB8BII	Mail with an @BBS of WB8BII
KA8Z	Mail with an @BBS of KA8Z
OKIPN	Mail with an @BBS of OKIPN
NEOH	Mail with an @BBS of NEOH
NTSKY	Mail with an @BBS of NTSKY
ALL*	Mail with an @BBS of ALL(anything)
"#NEOH	Mail with a HIERARCHICAL ADDRESS of .#NEOH
OH	Mail with an @BBS OR HIERARCHICAL ADDRESS of OH
[OH	Mail with a @BBS of OH
]OH	Mail with a HIERARCHICAL ADDRESS of .OH
EU	Mail with a HIERARCHICAL ADDRESS of EU (Europe)
442*	Mail to ZIP CODE 442xx

The last entry in the table MUST have a carriage return after it. There should be NO blank lines in the file, INCLUDING AT THE END OF THE FILE!

It is suggested that perhaps @BBS and hierarchical entries be listed in the file ahead of the more general ALL-type entries. In the event a logical connection is lost during a forward cycle, at least some of the personal mail will be forwarded ahead of the less urgent "for sale-rusty tower" messages.

To implement these (callsign).FWD/.REV Files files, the entry in MSYSFWD.DEF would look like this:

```
---------
F0 WA8BXN
$C:\MSYS\FWD\WA8BXN.FWD
---------
```

The dollar sign ($) at the beginning of the line is very important as it tells MSYS to include the contents of the file defined on that line. At the time of the automatic merge, or whenever the MERge command is given, the contents of the WA8BXN.FWD file will be included in the MSYSFWD.DEF file to give the final MSYS.FWD file. The included file may be changed anytime, but the results will not take effect until a MERge command has been executed. Due to the extra overhead of opening, reading, and closing the file, this should only be used for more than just a few call signs.

REVERSE FORWARD

MSYS can respond with messages to a connected BBS when it gives F>. If you wish another bbs to be able to REVERSE FORWARD (i.e. connect TO you and pull the messages listed, rename the file to: <callsign>.REV. This will permit MSYS then to examine the file and to forward any messages and bulletins to the other bbs upon request. Don't forget, before any bulletins can be forwarded to the other bbs, it must be assigned a number in the MSYSBBSB.DAT file!

So you may want to set up your forward file as follows:

```
[file MSYSFWD.DEF]:
F0 W8XYZ
$c:\msys\fwd\w8xyz.rev
--------
F0 K8XXX
$fwd/k8xxx.dat
--------
```

The file C:\MSYS\W8XYZ.REV could look like this:

```
W8XYZ
ARRL
ALLUS
```

The file C:\MSYS\FWD\K8XXX.DAT could look like this:

```
K8XXX
K9XYZ
ARRL
```

With these example files, reverse forwarding would be enabled for W8XYZ but not K8XXX. Recall that $ lines in the forward file MSYSFWD.DEF cause the given file to be included at that point.

When you run the MERge command, the contents of the C:/MSYS/FWD/W8XYZ.REV file will be added to the contents of the MSYSFWD.DEF file to get a combined file listing the routing to W8XYZ and all of the types of messages to be forwarded. Because the file is in the /FWD directory, and because it has a .REV extension, if W8XYZ should connect to you, your station will reverse forward, or send those items in response to a poll as well.

Reverse forwarding is very beneficial to a busy hub bbs as it will help move the mail. It is not uncommon in a metropolitan area to be forwarding to one bbs while several others are reverse forwarding from you, provided the network capacity is available to handle the traffic.

Also remember to be sure to use the MERge command whenever you make ANY changes to the forward files! Reverse forwarding of a given message is not possible until there has been at least one normal forwarding (either forced manually with the FOrward command or at your forward time). The other half of reverse forwarding (where MSYS would send F> when it is done forwarding messages to a particular station) is enabled with the R in the first line of the forwarding file (F0 becomes R0 WA8BXN).

When done reverse forwarding, MSYS just disconnects and does not send *** done.

Reverse forwarding checks if the bbs is currently connected. If so MSYS will not request reverse forwarding.

Reverse forwarding POLL (connects and requests reverse forwarding even if you don't have anything yourself to forward) can be implemented. Simply place P where you had R or F before. Thus: F0 WA8BXN becomes P0 WA8BXN

A check is made to see if asking for another message to be reverse forwarded will overfill the maximum number of messages.

Reverse forwarding will check if the required number of attempts at forwarding a message have occurred before reverse forwarding. Thus if we have OH 25 in the reverse forwarding file, a message that hasn't undergone 25 forwarding attempts will not be reverse forwarded by this entry.

SELECTIVE FORWARDING

Selective forwarding by title contents is limited to ONE selection per route. Thus if you have both ARRL LETTER and ARRL PFB in your forward file for a given bbs, only bulletins addressed @ARRL with LETTER in their titles will be forwarded. No other bulletins with @ARRL (including those with PFB in their titles) will be forwarded.

In addition to selective forwarding title comparisons, two additional selection criteria may be specified: number of forward attempts that must be made before using this path or maximum message size to be forwarded using this path. Here some examples:

```
F0 KA0FPO
ARRL
NEOH  DX
KQ8M  10
NEOH  '15
OKIPN  <3000
OKIPN  '<I
-------
```

This set of lines will cause the following to happen. All @ARRL messages will be forwarded. Messages with @NEOH will be forwarded if they contain DX (upper and/or lower case) in the title. Messages with @KQ8M will be forwarded if type is PN and there have been at least 10 forwarding cycles with the message present on the BBS. Messages with @NEOH will be forwarded if there is 15 somewhere in the title. The single quote mark is used to indicate that the number following is not the number of attempts that must be made to forward the message by other means. Messages with @OKIPN that are less than 3000 bytes long will be forwarded. And finally messages with @OKIPN and <I in the title (again, case ignored) will be forwarded. The ability to not use a given forwarding possibility until there have been the specified number of forwarding attempts is included so that preference can be given to favored paths while still allowing alternate paths to be automatically used if the favored ones fail. Limits on message size have been implemented for those who may forward to mini-BBS systems with limited storage capacity.

TO FIELD FORWARDING

New with MSYS 1.18 is the limited ability to selective forward bulletins based on the TO field as well as the @BBS field. If, for example, you have a bbs in your area that specializes in satellite items, you may wish to have an entry:

```
ALLUS >AMSAT KEPS OSCAR SATELL SPACE MOON
```

in the included file. The right arrow indicates to MSYS that the following TO fields @ ALLUS will be forwarded to that particular BBS. The limitation is that there can be only as many different catagories as will fit on one 80 character line. You may NOT have two lines of ALLUS >. If more than one line is needed, the bbs should probably get everything anyway. The TO-FIELD must be specified for each @BBS field it may occur with. In the above example, TO OSCAR will only be forwarded if it has an @BBS of ALLUS.

You may, however have a line for ALLUS and a second one for WW as in the following example:

```
ALLUS >AMSAT KEPS OSCAR SATELL SPACE MOON
WW >AMSAT KEPS OSCAR SATELL SPACE MOON
```

MERge:

This sysop command is automatically done at the startup of MSYS and may be executed by the Sysop while MSYS is running. If forwarding is in progress, it will execute at the beginning of the next forward cycle. The command causes MSYS to read the MSYSFWD.DEF file and copies it to MSYS.FWD, expanding any include files ($filename). A file is also produced, MSYS.BBS that is a sorted list of all possible BBSs that can be forwarded to (including things like ARRL, 44*, etc.). Any changes you make to your forward files must be done to MSYSFWD.DEF and/or the include files, followed by use of the MERge command (IN THE BBS).

FORWARDING NOTES

Make sure you don't have any blank lines in your various forward files (MSYSFWD.DEF and the files in the \FWD directory).

Executing Commands From Within the Forwarding File.

In the forwarding file, at the beginning of a set of forwarding lines, (i.e., at the beginning of the file or after a ------ line) you can place any sysop command(s) by simply putting \ in front of them.

The MSYS.BBS file contains a number after each BBS that the system knows how to forward to. This is the ORing of all the BBS Masks that can be used to forward the particular @BBS. This is used to determine when bulletins have been sent to all stations that should be getting them. when such a bulletin is found, it gets its type changed from $ to # so then it is no longer considered for forwarding.

Within the forward file a !timestring line may be placed before any callsign to limit forwarding hours for that callsign (or bulletin route). The first ! line as used before limits all times for a bbs entry you forward to. The following ! lines remain in effect until changed by another ! line. When you start another bbs forward entry group the forwarding hour defaults to 0-23.

```
For Example:   (the comments and blank lines would NOT really be in
               the forward file):
F0 WA8BXN      this is the beginning of a forward group
!0-16,19-23    don't initiate forwarding during hours of 5 or 6 P.M.
WA8BXN         forward the @BBS any time we connect
!0-16,22-23    don't forward the following during early evening
NTSOH
OH
!1-4           forward following only in early morning
ARRL
ALLOH
-----
```

Also remember to be sure to use the MERge command whenever you make ANY changes to the forward files! Reverse forwarding of a given message is not possible until there has been at least one normal forwarding (either forced manually with the FOrward command or at your forward time). The other half of reverse forwarding (where MSYS would send F> when it is done forwarding messages to a particular station) is enabled with the R in the first line of the forwarding file (F0 becomes R0 WA8BXN).

When done forwarding MSYS just disconnects and does not send ***Done.

**** BE SURE TO INCLUDE MSYS in your forward files for all other stations you forward to so I can send bulletins about MSYS to all of you. The list of MSYS operators keeps growing!

FORWARDING DISPLAY

As each of the BBSs to which msgs can be forwarded is scanned for a given bbs you will see it. When a message is being forwarded you will see a line that gives its number and the to station and @bbs of the message. In parenthesis you will see the entry in the forward file that is being used. Thus if the forwarding file contained.

```
F0 WA8BXN
WA8BXN
44*
---------
```

and there was a message for K8ZZZ@44123 being forwarded you would see something like:

```
 Fwding msg # 123 K8ZZZ@44123(44*). QSLDX @BBS:
```

Messages that arrive with @BBS of IDCALL get the @BBS field changed to --- . This makes them distinctive

Messages sent to REQQTH, REQDIR and REQFIL are left private if they arrive that way.

*** There can only be up to 400 different bbs names in the forwarding file. To check this do a LCount on the file MSYS.BBS.

BIDS

Messages that are entered on the system addressed to other than what looks like a callsign automatically get a BID generated for them.

Mail forwarded to BBSs identifying with either [MBL or [.. - $] get the BID included in the S command line.

BIDs may also be added to the message R: line with the RB on command.

Wildcards

Wildcards (where you can use *) have been enhanced to allow more than just using *. Based upon the recommendations of AA4RE and others, the following have been implemented:

```
* Matches 0 or more occurrences of anything
@ a..z, A..Z
# 0..9
+ a..z, A..Z, 0..9
? a..f, A..F, 0..9 (hex digits)
< a..z
> A..Z
$ =@#+?<>$     (any one of those characters, but I don't have any
guesses as to why this group was included)
= any one character
" the escape character: the next character must match exactly
( beginning of group . (abc) matches a b or c.

Examples: 9v1A2q   matches #@#@#@
      95===  matches anything 5 characters long beginning
             with 95
     (WK)6   matches anything starting with W6 or K6 (case
             is important)
      95###   matches any 5 digit number beginning with 95
```

MISSING TEXT TO Message

If the text for a message is not available when it is being forwarded, the forwarding is aborted instead of sending out a missing text message for the text. The STatus command will now indicate how many times this has occurred and the last message number for which it happened. In addition, an entry will be made in the event.log file. The MISsing command may also be used to determine how many times this occurred without generating all of the other STatus output.

The script consists of several sets of lines. The first line begins with @ followed immediately by the call (and Via digipeater list if needed) of the NODe to initially connect to. The port that will be used is taken from the digit after the F on the first line in the group. If the connection is established to the NODe, the following sets of lines are used to talk to the first NODe and possibly other nodes called through it.

Each set of lines for a given NODe begins with a line that has a period in position one. The following characters on the line are sent to the NODe as a command. The next line in the set is optional and if used has # in the first position.

Following this is the number of seconds to wait before abandoning use of this NODe. If this line is not given then 60 seconds will be assumed.

The next line is required and begins with + in the first position. The following characters are what are looked for in lines that come from the NODe to indicate that connection has been established to the station called (another NODe or the desired BBS). An exact match to the characters given on the line (including case) must be found somewhere in a line received to continue.

Next come two lines beginning with ? in the first position. These lines contain strings of characters that if found anywhere in a line that comes from the NODe that the connection was not made. Thus after we issue the connect command (given on the line beginning with .) one of three things can happen: a line from the NODe that matches the +string is found and we continue with the script or talk to the BBS, a line from the NODe matches a ?string which means we give up on making the connection, or a time-out occurs waiting for a match which again means we will give up the connection.

Example scripts:

These strings are useful for KA Nodes.

```
 @ANOD    (issue connect command to ANOD)
 .C       (issue connect to desired call)
 #25      (time out after 25 seconds if no match)
 +MADE    (if MADE comes from NODe, continue)
 ?BUSY    (if BUSY comes from NODe, abandon attempt)
 ?RETRIED (if RETRIED comes from NODe, abandon attempt)
```

For a NET/ROM NODe the following would be useful:

```
+to
?Failure
?Busy
```

Here is another example of a complete forward file:

```
 F1 W8XXX
 W8XXX
 W5XYZ
 ---------
 F0 W4XYZ
 @W7XXX
 .C W9XXX
 +to
 ?Busy
 ?Failure
 .C W4XXX
 #120
 +MADE
 ?RETRIED
 ?BUSY
 W4XYZ
 W4XXX
 -------
 F1 W9XXX-2 VIA K9XXX-3,KA9XXX
 W9XXX
 W1XXX
 W3XXX
 K0XXX
 ----------
```

NOTE: The forward connect script for a given bbs is limited to 25 lines.

MSYS NODe Forwarding

You can forward directly through your own MSYS node (assuming it is properly activated). Here are some examples how to do it:

```
F0 W8XYZ  This is the station we are forwarding to, the number
%W8XYZ    after F is not important.  Then do a network connect to
          W8XYZ - assuming  W8XYZ is in our local dest list
W8XYZ     forward @ W8XYZ stuff to this bbs
-------
F0 W8XYZ
%W8XXX    {connect to node W8XXX which is in our local dest list}
+to
?busy
W8XYZ     stuff to forward to this bbs
-------
          NOTE:  There should be NO BLANK LINES in this file!
```

FORWARDING TO JNOS SYSTEMS

TO forward to a JNOS TCP/IP mailbox the file may look like the following.

```
---------------
F0 W8UPD
.
+NET (may not be needed)
W8UPD
(MORE @BBS CALLS)
---------------
```

Another way that also seems to work:

```
---------------
f0 w8upd
%UAKRON
$c:\msys\fwd\w8upd.rev
---------------
```

Please note that some JNOS systems may require you to make the final connect to the ALIAS rather than to the callsign of the bbs. There is a command in JNOS that probably should be used: mbox jumpstart to permit the JNOS system to start the handshake sequence without awaiting the extra carriage return.

HF FORWARDING

HF Forwarding regulations do not allow transmitting 3rd party traffic unless there is a control operator present, the station is under remote control, or a STA has been issued. To aid in compliance with the regulations, the ATtended command has been added. When set to ON all forwarding is allowed. If off, ports with radio speed of 300 will forward only those messages in which the from call is the idcall (messages that you originate) or the to station in the message is the station you would directly connect to for forwarding. HF reverse forwarding is also inhibited if ATtended is OFF. Note that the regulations appear to allow you to receive messages unattended. So if you don't have a STA for HF turn AT OFF when you aren't there and remember to turn it ON when you are. To be on the safe side, put ATtended OFF in your MSYS.DO file should the system reboot when you aren't present.

PACTOR FORWARDING
GTOR FORWARDING

The commands for PACTOR and GTOR are essentially the same, with the obvious difference in the mode of the TNC. Automated forwarding should not be done using PACTOR/GTOR. However, reverse forwarding can move traffic off a remote PBBS. A MSYSFWD.DEF-like file is made with a filename of your choice. This will serve as a manual forwarding file.

The following is an example of a pactor forward file:

```
Filename = NO8M.PAC
     p0 no8m
     wa8bxn
     wb8bii
     k8eiw
     -----
```

Only callsigns or routes can be in this list. You can NOT have the $filename.ext type of include file.

The above causes a connect to NO8M and sends traffic for the three listed PBBS stations. A reverse forward prompt then results in traffic for the connecting station to be reverse forwarded.

When you wish to manually forward, enter the following command from the sysop keyboard:

```
     FP filename.ext
```

or in the previous example:

```
     FP NO8M.PAC
```

MSYS then acts on your file and forwards via the PACTOR/ GTOR port. MSYS can have a "normal" forwarding session in progress at the same time as it is forwarding via PACTOR/GTOR.

DO NOT ENABLE DIGI ON YOUR PACTOR/GTOR PORT!!

DON'T FORGET TO ENABLE CW ID IN YOUR PACTOR/GTOR TNC!!

MSYS does not yet support the new Kantronics TOR mode automatic switching.

Sample entry in MSYSFWD.DEF:

```
T 44 70 241 1   (DO NOT USE DOTS IN THE ADDRESS)
W8UPD
-----
```

Note that no port number is specified, it does an arp if needed. Also note no digis (put them in the arp table if you have to have them). No nodes. Just T and the IP address of the DEStination station as shown. This corresponds to the line like F0 W8UPD. Next comes the @BBS values to send to this station. There can be more than one. Then the ------ that ends this entry. The same forward file is used for both normal forwarding and tcp/ip stuff. I have also made some changes to the SMTP command. After you use it, if you type PRompt it will prompt you for To: (enter call@bbs) Reply-to: (just hit return if you like the default given) Subject: (this is in the body of the text, not the title) Then you enter your text and end with. (not ^Z) as instructed. The result is a message that contains all the fancy extra lines often found in tcp/ip messages.

Bulletins may be forwarded using SMTP. IMPORTANT: SEE NOTE BELOW To do so you must assign the SMTP station an entry in the MSYSBBSB.DAT file like you would for normal forwarding (don't use any SSIDs). Also add the callsign of the station in the T line in the forward file after the IP address. Example: T 44 70 242 6 WB8BII This should be done in the forwarding file for all SMTP forwarding.

Bulletins are accepted from SMTP (a bulletin goes to a non-callsign). BIDs are made up based on your message number and callsign.

Hierarchical addresses are added to msgs that come in via SMTP.

NOTE: Since BIDs are not supported in this implementation of SMTP, bulletins can either be sent to or received from TCP/IP systems but not both ways.

MSYS will accept <RCPT TO>: lines in either the format a%b@c or b!a@c where c is your bbs call, a is the destination stations call and b is the destinations bbs. Both of these get transformed into a@b on the MSYS message list.

FBB COMPATIBLE FORWARDING

MSYS follows the standard established by F6FBB in his version 5.14 and 5.15 of his BBS software.

As described in the FBB manual, once the F letter in the SID (System type IDentifier) has been detected, all command lines must start in the first column with the 'F' character. All command lines are terminated by a return (CR) character. When the connection to another bbs using the FBB protocol has been made and the 'F' flag has been received, the SID of the local station is sent with the first proposal. The proposal may look something like this:

```
FB P WA8BXN WB8BII K8EIW 12345_WA8BXN 765
F>
```

Where:

FB	Type of command (proposal)
P	Type of message (P = personal, B = bulletin)
WA8BXN	Sender (From field)
WB8BII	BBS of recipient (@BBS)
K8EIW	Recipient (To field)
12345_WA8BXN	BID or MID
765	Size of message in bytes
F>	End of proposal

In the case of multiple message proposals, the 'FB' line is sent for each message proposed followed by one F>.

Upon receiving the proposals, the other BBS will accept or reject the message with the command line:

```
FS -+=
```

where

- `-` means No, already have that message
- `+` means Yes, send the message
- `=` assumed by MSYS to be No, message being forwarded by another system.

The FS line must have as many +,- signs as FB lines in the proposal.

Each message is sent with the title on the first line, the text, and a control Z in the last line. There are NO blank lines between messages.

When the other BBS has received all the messages in the group, it acknowledges by sending its proposal and the direction of forwarding is reversed.

If the receiving bbs has no further messages to send, it responds with a: FF
This line must NOT be followed by a F>.

If the sending BBS has no further messages, it responds with a:

FQ

and then disconnects.

FBB COMPATIBLE COMPRESSED FORWARDING

The protocol utilized for the transfer of compressed messages is an extension to the standard FBB protocol. Compressed forwarding is validated by the presence of the letter B in the SID [MSYS-1.14-BFHM$].

In the case of compressed forwarding, the proposal line begins with a:

FA

FA means that the transfer will be an ASCII compressed message.

The proposal to send a message is in the form:

FA P WA8BXN WB8BII K8EIW 12345_WA8BXN 6789

Compressed message transfer is done in a binary mode, which is derived from the YAPP protocol. All transfers consist of a header, a block of data, an end of message and a checksum. Each transfer is equivalent to the transfer of one message of the standard protocol, but is not followed by a control Z. The end of file specifier is defined by the transmission of a EOT character followed by a 1 byte checksum..

Format of the header for a compressed message:

<SOH> 1 byte = 01 hex
Length of the header 1 byte = Length from the title,
Title of the message 1 to 80 bytes
<NUL> 1 byte = 00 hex
Offset 1 to 6 bytes
<NUL> 1 byte = 00 hex

To follow the FBB format, the title of the message is transmitted in ASCII, not compressed.

The offset is also transmitted in ASCII and specifies the offset at which the data should be inserted in the file (in case of a fragmented file). A data block contains from one to 256 bytes. It begins by two bytes which specify the format:

<STX> 1 byte = 02 hex
Number of data 1 byte = 00 to FF hex. (00 if length = 256 bytes).
Data bytes 1 to 256 bytes

The last data block is followed by the end of transmission character (EOT) and the checksum in the format:

<EOT> 1 byte = 04 hex
Checksum 1 byte = 00 to FF hex

The checksum is equal to the sum of all the data bytes of the transmitted file, modulo 256 (8 bits) and then two's complemented.

The checking of the checksum is very simple:

The sum of the data from the file and the checksum received modulo 256 (anded with FF) shall be equal to zero.

In case of a checksum error, the message is not accepted and the system sends the comment

*** Checksum error

and issues a disconnect request.

The transmission sequence is repeated for each message in the group.

W0RLI STANDARD FOR DATA COMPRESSION DURING THE FORWARDING OF MESSAGES
IN THE AMATEUR RADIO BBS NETWORK

ihave/iwant variable batch size protocol
Feature letter X, commands SS, SX, SY
LZH compression

Command Definitions and BNF description.

```
N               = INTEGER, size 32 bits.
protocol        = TBD
batch_size      = N
file_size       = N
msg_count       = <0-25>

id              = BID | MID EOL
ids             = id | ids id

byte            = <0x00 - 0xff>
bytes           = byte | bytes byte

ihave = "SX" msg_count [" " protocol] [" " batch_size] EOL [ids]
iwant = "SY" msg_count EOL [ids]
isend = "SS" file_size EOL bytes
```

For ihave:
Protocol and batch_size not used in initial implementation.
Default protocol is : LZH <C source provided>

For iwant:
SY means want only there.

The basic interchange (example):

---

bn are BIDs (or MID if message has no BID).
m: is master, s: is slave.

```
m: SX 5
m: b1
m: b2
m: b3
m: b4
m: b5

s: SY 3
s: b1
s: b3
s: b5
```

m: SS 26545
m: <sends 26545 byte file>

In the case of any error, the station noticing the error must disconnect.

With more detail of one implementation (MB/SERVER processing)

From Sending side:

MB:
Scan messages to find batch not larger than max. batch size.
Keep record numbers and cc: id in list.
Send ihave.
Receive iwant.
Mark all messages in ihave and not in iwant as forwarded (dup bid)
Pass iwant list to SERVER in file.
Wait for server acknowledge.

SERVER:
Create export file.
Compress export file.
Signal MB that compressed file is ready to send.

MB:
Send isend, send the file.
Receive prompt.
Mark all messages in iwant as forwarded.

From the receiving side:

MB:
Receive ihave.
Check BIDs in ihave.
for all non-dup BIDs, make iwant list.
Send iwant.
Receive compressed file.
Signal SERVER import file available.
Send prompt.

SERVER:
Decompress file, import messages.

FORWARDING ON THE MODEM PORT

LET'S PAUSE FOR A MOMENT ... DO NOT FORWARD WITH MSYS ON A TELEPHONE LINE UNLESS YOU ARE ABLE TO INITIATE A ERROR CORRECTING CONNECT WITH THE MODEM. A NON-ERROR CORRECTING CONNECT WILL MAKE YOUR STATION RESPONSIBLE FOR GARBLED, JUMBLED AND MESSED UP MESSAGES.

ALSO: DO NOT ATTEMPT FBB COMPATABLE OR COMPRESSED FORWARDING VIA THE MODEM PORT. YOU WILL OVERRUN THE AVAILABLE BUFFERS!

The FWD.DEF file defines the script MSYS uses to establish a forwarding session.

In the FWD.DEF file, put a F#, R#, or P# line followed by the @BBS items to forward. The format of the F, R and P lines are identical except for the letter. This format is different from what would be used for packet forwarding and is as follows:

```
F# callsign phone# callword password
$fwd/<call.rev>
```

where #	is the port number,
callsign	is the call of the station you are forwarding to,
phone#	is the string to send to the modem to dial,
callword	is the reply to give to Call:, and
password	is the reply to give to Password: (questions asked by the destination MSYS system)
$fwd/<call.rev>	is the directory and file which contains the forwarding information

Example:

```
F1 WA8BXN ATDP5551212 swiss normal
$fwd/wa8bxn.rev
```

NOTE: The log-in callword and password are converted to UPPER CASE. Thus the station you are calling must have UPPER CASE values for your callword and password. The station you are calling must be running MSYS 1.11 or higher.

NOTE: The included file in the above example will be in the MSYS/FWD directory.

FBB ON THE PHONE

FBB style forwarding (compressed or not) should NOT be used on the modem. Compresssed forwarding just won't work and the non compressed mode has a high likelyhood of overrunning buffers on long messages because of the continuous transmission stream of a block of messages. Therefore, the FBBC parameter MUST be set to 0 for the modem port. "Normal" forwarding to and from FBB systems is possible however

Use ONLY FBBC 0 and XO OFF on modem ports. Very Important!!!

Use of control characters in the call and/or password should not be needed. The callsign must be prefixed with a period to turn off echoing from the FBB system. Here is an example forward file entry

```
F2 K8EIW ATDP5551212 .WA8BXN MSYS
```

The F2 could be R2 or P2 to ask for reverse forwarding or to always poll respectively. The 2 can be replaced by whatever port number your modem is connected to.

K8EIW is the call of the bbs to which we are forwarding.

ATDP5551212 is the telephone number to call. The P could be replaced by T if tone instead of pulse dialing is desired.

.WA8BXN specifies the "callsign" value to be give. The period says to FBB systems to turn off echoing. The actual callsign value used by FBB is just the WA8BXN part.

MSYS is the "password" value that will be sent to the FBB system.

This example is for forwarding to FBB systems. When forwarding to a MSYS system the only difference is that the period is NOT used in the "callsign" field.

FORWARDING TO AA4RE SYSTEMS

It has been reported that for MSYS to forward to a 4RE bbs via land line, the 4RE bbs needs to change the responses in the MESSAGE.BB file. First, change MESSAGE 95 N to: Call: and MESSAGE 96 N to: Password: These two changes have been tested between KI7HM (MSYS and KD7HP (4RE) and are reported to work ok. (Thankyou Gentlemen!)

ANONYMOUS CHECK-INS

It is possible to set up the modem port to take a call from an anonymous check-in. As long as all outgoing traffic is held, people interested in ham radio, awaiting their license or whatever can read and send traffic. As long as ALL outgoing anonymous traffic is held, there are no legal implications. MSYS is designed so that if someone uses the string "visit" for both the call and the password, and if visit is in the MSYSPASS.MOD, MSYS will allow a check in. Here is how the MSYSPASS.MOD should be set up:

```
VISIT VISIT V1SIT
Visit Visit V1SIT
visit visit V1SIT
(other calls follow)
```

By having all the various combinations of case, a anonymous check-in does not have to worry about the case that is used. Now, it is CRITICAL that MSYSHOLD.DAT have the lines "connected=v1sit" and "from=v1sit". All anonymous traffic will be held. It can then be routinely deleted or readdressed by the sysop (with the sysop's call being used for the sending party).

It is also necessary to use the "EU V1SIT" command to add the user to your system. Here is what the file will look like when done:

```
V1SIT is VISITOR located at VISITOR  Last new message 19485
Last connected Fri Mar 19 23:10:57 1994 on port 5
Flags=$0140 Lines=20
Postal code: VISITOR  Home BBS: NO8M  Last WP Update: Mar 10
08:41:01 1994
Path to V1SIT is Direct
```

Other calls than V1SIT can be used, however MSYS is programmed to disallow changes to the V1SIT callsign's information. An anonymous check-in can not go in and change V1SIT's home PBBS, name, etc.

Also, please note that the "FLAGS" for V1SIT are set at $140 to prevent erroneous WP messages from being generated!

It helps to have a help file that tells the anonymous users something about how to get help. Here is an example MSYSMOD.MSG:

```
****************************************
** Welcome to the NCARC MSYS Modem Port! **
****************************************

For information on the system, use "I".  To a list of messages on
     the board, use the command "L 0".  To see what is in the
     download section, use the command "W".  (Use "D FILES.LST" and
     "D README.DOC" for information on how to get those files.)  A
     MSYS User's Manual is available and how to get it is in the
     file USERMAN.INF.)  Please DO use this modem port for
     downloading the LONG files I have here ... we have PART 97 of
     the FCC Regulations and other very long files that are
     inappropriate for downloading by a RF link.

If you are interested in amateur radio and how to get your license
     leave me a message (SP NO8M or SP SYSOP) and I will get you in
     touch with someone that can help.

-----end of MSYSMOD.MSG
```

Anonymous use of the port will be in direct proportion to the advertising that is done outside of amateur circles. If you make up fliers and distribute them, you will see port usage rise.

ACOUSTIC COUPLED MODEMS

There are times that you may want to utilize an acoustically coupled modem port. This might be needed when you want to forward automatically from a pay phone. The MSYSFWD.DEF line in this set up would read like this:

```
-----
R0 NO8M ATX1D callword password
$FWD\NO8M.REV
-----
```

A callword and password that is a unique, pretend user is set up. If the callword and password were BOZO and CLOWN, the following line could be used in your MSYSPASS.MOD:

```
BOZO CLOWN N08M
```

The pretend callsign N08M would then be set up on the home PBBS to forward what you wanted. You would add the pretend call to your home MSYSFWD.DEF like this:

```
-----
r5 n08m
!25
$fwd\n08m.rev
-----
```

This would cause the home MSYS to keep all the traffic that you wanted for the acoustic system (and was listed in the N08M.REV) listed as PN or B$ and ready to forward into the remote system. Make sure you add the pretend call to the MSYSBBSB.DAT if you will be forwarding bulletins.

MISCELLANEOUS NOTES

ON USAGE If you have a user who ties up a RF port for hours reading each and every message, you may benefit by having him start up a MSYS check in on the phone port and get each and every message automatically. In that way, the user can read the stuff off-line at their own computer.

ON R: LINES: The above user may then try to automatically forward messages out on that phone MSYS that was set up. This will add a new R: line that will be unknown to your network. You can either begin supporting each and every PBBS that comes and goes or demand that the station send their message with NO R: line and that they send their messages manually.

REQUEST: Please forward problems, concerns and hints to Steve, NO8M@NO8M.#NEOH.OH.USA.NA.

MSYS.REP
CALLSIGN @ BBS REPLACEMENT

This feature is used to change the routing or direction of a message. It can be used to redirect traffic for a certain ZIP code to a specific bbs that has capabilities for handling it, or it may be used to redirect messages from one LAN to another.

Create a file called MSYS.REP. In it place lines with the following format

OLDCALL OLD@BBS NEWCALL NEW@BBS

For example:

```
ALL  ALLONT   ALL  NEOH
     (messages  to: ALL@ALLONT  become ALL@NEOH)
44* NTSOH @1 ---
     (messages to 44*@NTSOH    become originaltocall@ ---)
NTS* * @2 @1
     (messages coming in to NTS-anything get tocall and @bbs swapped)
```

*** IMPORTANT *** There can be no more than 150 lines in this file! The algorithm goes like this: When a message comes in, a check is made comparing the to call and @bbs of the message with the first two fields of the records in MSYS.REP. If a match is found (wild cards are allowed in MSYS.REP) then the last 2 fields are used to replace the tocall and @bbs of the message. @1 or @2 in the last two fields in MSYS.REP have a special meaning. @1 means the original to call and @2 means the original @bbs. Note that wildcards are not expanded in these last 2 fields. To support this facility a new command is also added: REPlace. If an argument of L is specified, the current replacement list is displayed. If no argument is given, the current contents of MSYS.REP are read and stored in memory. (This is done when MSYS is started also). NOTE: You MUST use the REP command after you have edited MSYS.REP to make the changes effective.

DATABASE HELP

Commands related to Database manipulation available on MSYS

MSYS allows BBS users to input information and search for information in simple data bases. The names of the databases are Database.0 through Database.999. These files are located in the normal FILES directory, which can be accessed by the D command (download) on the BBS. These files can also be searched by the G command on the BBS. To use this command you enter the command G (with no operands). You will then be asked for the file you want to search. Simply pressing return at this point will select Database.0 by default . You are then asked for the search string. This is a set of characters that when found in lines of the file cause those lines to be displayed. Case is ignored during the search. If a selected line ends with \ then the following line(s) are also selected for display. Data is entered into a data base by sending a message to MSYS with DBADD in the title. The text of the message will be added to the indicated Database (Database.0) by default. A header line is also inserted in the data base telling the station that made the entry, long with the date and time. Here are some examples:

```
SP MSYS   {this is a S command issued to the BBS}
DBADD 3   {the message title, Database.3 to be used}
wind damage mentor Ohio \
large tree down at 615 & 20 \
reported by mentor pd \
no injuries
^Z   {normal way to end message}
```

The text of this message would be added to Database.3. Now if you search Database 3 for either wind damage or mentor for example, you would get to see these lines, along with any other lines that have wind damage or mentor in them.

```
Example of using search command:
G               {BBS command to do search}
File search routine {Response from bbs}
Enter file name (or just press return for Database.0): Database.3
Enter search argument: Mentor
{lines from message above displayed}
>               {BBS ready for next command}

Another example:
Assume a file had the following lines:
Mary had a little lamb
its fleece was white as snow, \
and every where that Mary went
the lamb was sure to go.
It followed her to school one day
If you searched this for 'snow' you would get the following back:
its fleece was white as snow, \
and every where that Mary went
```

That's a brief outline of what can be done. Its simple yet very flexible. Let me know of any applications you can think of for these facilities. Particular Database numbers can be assigned to different uses. Formats for information can be suggested for each different Database.

HIERARCHICAL FORWARDING and MSYS

There are several features in MSYS that combined with hierarchical forwarding should almost completely automate routing of incoming messages. These features act at two separate times: first when a message comes in and second during forwarding.

First let's review the way hierarchical forwarding works. With hierarchical forwarding the @BBS field can contain more than just the callsign of the BBS that is the DEStination of the message. Routing information can also be placed in this field. The @BBS field thus has the syntax of part1.part2.part3. .... When it is time to forward a message, the @BBS field is inspected and the leftmost part that is found in the forwarding file is used to forward the message. Consider as an example @N8PIH.MI. If N8PIH is found in the forwarding file then it is forwarded that way and the MI part is ignored. On the other hand, if N8PIH is not found in the forwarding file, then if MI is found in the forward file then it is forwarded that way. If neither part is found in the forwarding file, then the message will not go anywhere. Such messages will show up (after at least one forwarding attempt after the last time MSYS was started) if you do a L? command.

If all messages had both the actual callsign of the DEStination BBS and the appropriate xx part in the @BBS field then the forwarding file could be greatly shortened. You would need to have the calls of the BBS stations in your state in the forward file, along with each of the xx designators for the remaining states. If there were say 50 BBS systems in your state then perhaps 99 different entries would be sufficient in your forward file (ignoring for the moment bulletin routes like ARRL).

NOTE that the limit of 400 different calls in the forward file is more than sufficient! Messages that are going to a BBS in your state get routed as specified in your forwarding file; those going to another state get routed to the BBS you have set up in the forwarding file to get stuff for that state. You could include a few explicit calls for BBS systems in nearby states that you route directly to, if you want to override the normal path for messages going to that state if you so desire.

Now for the second part, the action taken by MSYS when a new message arrives. When a message comes in, MSYS looks at the @BBS field. If it is absent, then if the to callsign field contains something that MSYS knows the routing for, the to field is copied to the @BBS field. Otherwise the @BBS field is set to ===== and the message won't go anywhere. It will show up as noted above in a L? command.

For forwarding NTS traffic, MSYS uses a file called BBSTONTS.BIN that contains BBS callsigns and the states they belong in (this information comes from several sources, more on this later). Next MSYS inspects the @BBS field to determine if it knows how to forward what is in the given @BBS field. If some part of the possibly hierarchical specification is found in the forwarding file, it is left alone and will be forwarded by that matching specification. If no match is found in the forwarding file, then the BBSTONTS.BIN file is searched to find some part of the given @BBS field. If a match is found, the corresponding NTSxx is appended to the existing @BBS field. Thus MSYS will attempt to lookup the right state to send the message to. Assuming the state is found, then forwarding will take place as described above (see the N8PIH.MI example). If no part of the @BBS field can be recognized, then the message will not be sent out, but will show up when you do a L? command as described above. The BBSTONTS.BIN look up will also handle zip codes given in the @BBS field, making @12345 into @12345.xx as is appropriate.

Thus far we have examined what happens when in the process of automating forwarding. In the event the proper routing cannot be determined by MSYS then you will have to use the Sysop EM (edit message) command to manually add the appropriate .xx designation, assuming you can guess it somehow. If you are lucky, the originator put useful information in the title of the message or may have hidden it in the text of the message. Maybe the callbook will be of some use to you. Or maybe you can just force it in the opposite direction from which it arrived.

IMPORTANT! Hierarchical parsing is left to right

It needs to be emphasized that MSYS will forward on the leftmost element matched in the @BBS ONLY. For example, if the @BBS of a message is KA8Z.#NEOH.OH.USA and there is an entry in the WB8BII.REV file for file for KA8Z, and the path to WB8BII is lost, the message will NOT alternate route to the #NEOH path.

Bulletin Forwarding Logic

New in MSYS 1.18 is the limited ability to forward bulletins based on hierarchical components normally used for specific bbs addressing, such as OH or USA. By the proper use of the new [and] symbols, it is possible to route bulletins as well as messages to specific bbs's based on specific portions of the given hierarchical address.

1. To determine which field of the @BBS will be used to forward a message, the @BBS components are examined from left to right until a component is found in the forward file that matches. For purposes of comparison, xxx in the forward file will match xxx in any position of the @BBS, [xxx in the forward file will match only the leftmost component of the @BBS, and]xxx in the forward file will match any component but the leftmost part of the @BBS. Item 7 below gives more detailed logic for determining which component of the @BBS is to be used for forwarding.

2. A flood bulletin is a bulletin that will be forwarded on its leftmost component to all bbs systems in the forward file that contain either the leftmost component or the leftmost component prefixed with a [. A non-flood bulletin is a bulletin that is forwarded on something other than its leftmost component in its @BBS field. An example of a non-flood bulletin would be @ALLNY.NY when passing through the state of Ohio on its way to New York. Such a bulletin would be forwarded on its NY component and sent to only one station that handles messages going to NY.

3. Each unique component found in the forward file contains an entry in the MSYS.BBS file. [OH, OH, and]OH are considered to be three distinct components. Components that begin with] cannot be used to flood bulletins.Those components in the MSYS.BBS file that can be used to flood bulletins will contain a number indicating which BBS forwarding group(s) they appeared in, the particular numbers being derived from the MSYSBBSB.DAT file which gives each BBS to which bulletins are forwarded a unique number. Components that cannot be used to flood bulletins or are only sent to BBS sytems that are not listed in MSYSBBSB.DAT will have zero for the number in the MSYS.BBS file.

4. Each message has a bits field that is used to keep track of which BBS ssystems have which message, either because their R: line was inthe message when it was received or because it was forwarded to the BBS. Only those BBS systems with entries in MSYSBBSB.DAT are recorded.

5. At the beginning of each forward cycle, each flood bulletin is inspected to determine if it has been forwarded to all the systems that should get it. Bulletins that have gone everywhere are changed fron type $ to type # which indicates they no longer need to be considered for forwarding and as an indication to humans that they have been forwarded everywhere. A flood bulletin is considered forwarded everywhere it should go if its bits are set for every BBS it should go to. Recall that these bits are set when the message is first received based on the R: lines as well as when the emssage is forwarded. The bits that must be sent for a bulletin to have been sent everywhere are the result of a logic OR between the bits indicated by the leftmost component of the message's @BBS as determined from the @BBS file and the bits associated with the leftmost component prefixed with [obtained from the MSYS.BBS file.

6. Non-flood bulletins should be marked with type F (and perhaps killed if autokill is on) the first time they are forwarded.

7. The following logic is used to determine which component of the @BBS field of a message will be used for forwarding. The first step that has success terminates the logic.
   MSYS.BBS is searched for
      (a) the leftmost component
      (b) the leftmost component prefixed with [
   For the remaining component, MSYS.BBS is searched for
      (c) the component
      (d) the component prefixed with]

For example, if the @BBS is X.Y.Z, MSYS.BBS would be searched for a match in this order:
X
[X
Y
]Y
Z
]Z

In summary, if you wish to forward only flood bulletins with an @bbs of OH, you would have an entry of [OH in your <bbs1>.REV file.

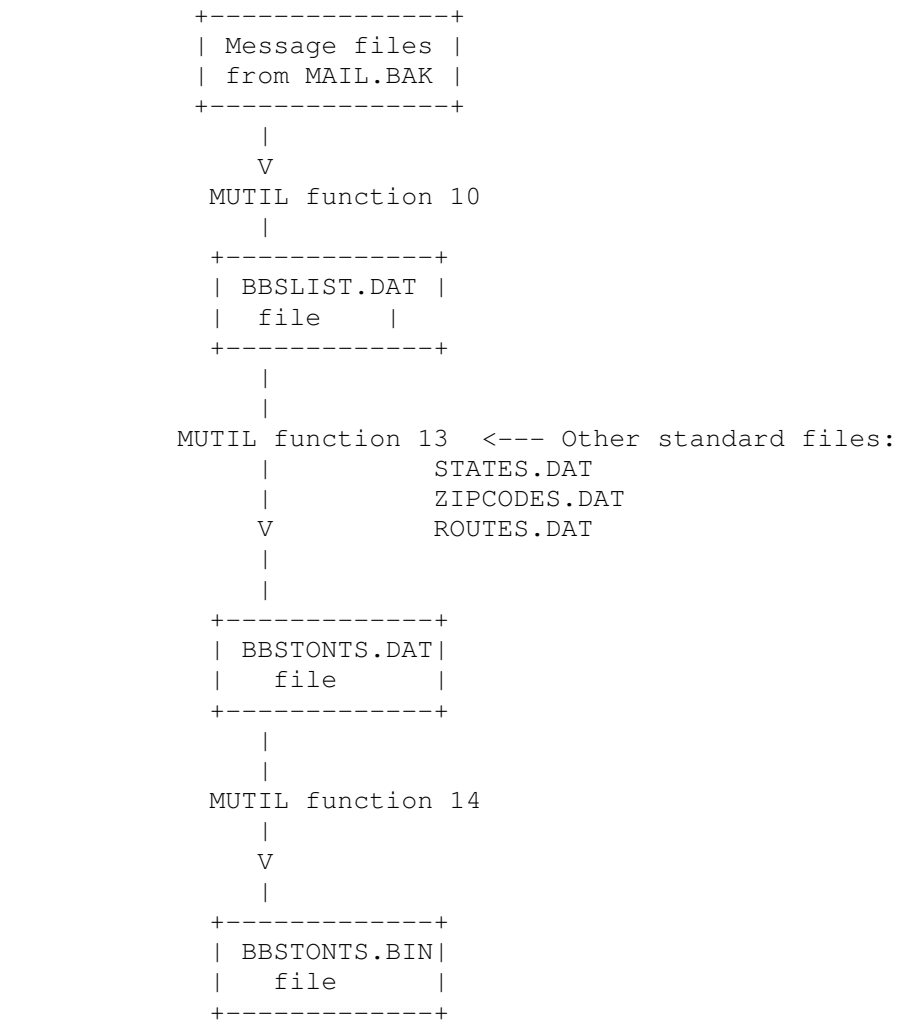
If you wish to forward messages with a hierarchical address of <bbs2>.OH, and not bulletins @OH, you would have an entry of]OH in the <bbs3>.REV file.

If you wish to forward both bulletins @OH and messages @<bbs>.OH, the proper entry in the <bbs4>.REV file would be: OH

Creating the BBSTONTS.BIN file.

This file contains binary data and cannot easily be created with a text editor. Mutil function 14 takes as its input a file called BBSTONTS.DAT and produces as output BBSTONTS.BIN. To create BBSTONTS.DAT, you run a function called BBSTONTS (function 13 in MUTIL) which takes as its input BBSLIST.DAT. BBSLIST.DAT is created and/or updated using data from message headers that have passed through your system and/or AD8I BBS lists. Function 10 of MUTIL is used to use messages as input; function 11 uses AD8I lists as input.

Lets look at a picture of these files and programs:



MSYS.HCL

You should create a file called MSYS.HCL into which you put lines describing classes of messages you want to eradicate. Each line should contain one or more of the following keywords:

AT= TO= FROM= AGE= TITLE= TYPE= KEEP= SYSOP=

So for example:

```
AT=ARRL AGE=30
```

would delete all messages with @BBS of ARRL that are over 30 days old. If specified, the values of AT TO and FROM must match exactly for a message to be deleted. TITLE and TYPE if specified must match some part of their corresponding message header fields; case is ignored.

KEEP can be used to specify a number of msgs that match the other values given that should be saved anyway.

Example:

```
TO=ALL AT=ARRL TITLE=PFB KEEP=2
```

Would keep the newest two messages to ALL@ARRL with PFB somewhere in their title. NOTE: The value of title given cannot contain blanks.

DIR is used to specify a Directory you wish to store a message into before killing. For Example:

```
TITLE=CD-ROM age=3 DIR=FILES\HARDWARE
```

Will store any file with "CD-ROM" in the title that is over 3 days old in the file MSYS/HARDWARE
The title of the message will be the BID number. This will probably require some Sysop editing to change the title to something more meaningful.

SYSOP=Y has to be specified in any line that you want to delete messages that are to the idcall of the BBS. Thus to delete all read messages (including one to you) you could have a line:

```
TYPE=PY SYSOP=Y
```

The following line deletes read messages, except those to you:

```
TYPE=PY
```

The effects of these lines can be far reaching. For example, if you put in the line:

```
KEEP=5
```

Then it will delete all but the 5 newest messages! All this action takes place when you type:

```
HCLean NOW
```

or at a time specified as a numeric parameter to HCLean. HCL 120 would cause it to happen at 02:00 (number given is minutes after 00:00 [midnight] to begin housecleaning).

Please note that BID= is NOT one of the items that is used to select messages to be killed in the MSYS.HCL file.

MSYS.HCL file as an example:

```
at=arrl age=21
at=okipn age=10 a
to=panet age=3 type=#
to=mods age=3 type=# dir=files/mods
at=alloh age=10 type=#
at=allbbs age=7 type=#
at=amsat age=7 title=schedule dir=files/amsat
at=amsat age=7 type=#
at=midnet age=3 type=#
at=arrl title=pfb keep=1 type=#
at=arrl title=dx keep=1 type=#
at=arrl title=kep keep=1
type=# title=sale age=2
type=# title=need age=3
type=# age=35
type=py keep=1
type=pf keep=2
type=tn age=21
type=pn age=60
to=sysop age=7 type=#
```

At the end of housecleaning, the file MSYSHCL.DO is done (as a DO file) if it exists.

MSYS NETWORK NODe INTRODUCTION

Some basic information concerning the implementation of the MSYS NODe interface will help in making decisions related to configuration. Two terms must be understood by the system administrator (SYSOP). These terms are neighbor NODe and DEStination NODe. A neighbor NODe is a compatible NODe to which there is a direct connection path. MSYS does not support use of digipeaters in the path to neighbor nodes. Normal AX.25 connections are made to neighbor nodes. The data exchanged by these connections consists of a mixture of supervisory and data frames that together support the network operation. DEStination nodes are also network nodes and are the nodes listed when one does a Nodes command. DEStination nodes are nodes that can be reached through the network either directly if they are also neighbor nodes, or indirectly through some neighbor NODe.

Routing

Routing of frames in the network is conceptually fairly simple, handling all the details is not! Each NODe does a "Nodes Broadcast" periodically (usually every 30 minutes) which is a list of nodes that it knows how to route to. This may be due to directly hearing a neighbor NODe or by hearing the nodes list from neighbor NODe that claims to have a route to each NODe it lists in its NODe broadcast. Information from the nodes broadcast is maintained by MSYS. When it receives a frame that is not for itself, MSYS looks to see which neighbor NODe can handle the DEStination indicated in the frame and simply passes the frame on to some neighbor NODe.

Data frames while they are inside the network as handled as what are termed "datagrams". What this means is that during a conversation between two end users there is no guarantee by the network that the data frames will always go by the same path or that they will make it through the network. Depending on conditions in the network the path taken by frames can change and at times frames will be arbitrarily discarded. The end nodes (the ones the users connect to) are responsible for error free end to end services. While the AX.25 protocol insures that transmission of frame between any two nodes is error free, the network protocol establishes error free operation between users.

MSYS network NODe functions include both the routing of frames within the network as well as the uplink/link logic to support the error free end to end services. Additionally, the TCP/IP servers can make use of the network as an alternative to the existing IP routing facilities. Connections from the network may be made directly to the BBS through the enhanced NODe command set.

An attempt has been made to make the MSYS network NODe user friendly while still retaining compatibility with other network nodes. To this end there is a help command and headings on output generated by various information listing commands. Since MSYS is generally run at an attended site, a Talk command is available to allow users to talk to the Sysop directly from the network.

Basic Decisions About the Network NODe

The really big question is should you enable the network NODe functions of MSYS? Fortunately, you have more options than just yes or no! So perhaps the question should be reworded as should you run the NODe in full feature, fully automated mode? The answer for most stations will be NO. The answer in your particular case will depend on the capabilities of your station and the locations of other compatible nodes. If you run a high profile station already (wide coverage as a digi/K-NODe) and have compatible network nodes at reasonable distances from you (maybe at least 30 miles away), then you may fill a need by running with everything enabled. On the other hand, if the only way you can talk to anything is through a digipeater (even when the batteries in your ht are fully charged) and on a hazy day you can still see the antennas of at least five network nodes, all still forwarding the same frame they received 2 years ago, then may I suggest you run in a very controlled minimal mode. Make friends, not enemies through these enhancements to MSYS!

Network NODe Quickstart

This section tells the minimal things you must do to get the NODe up and running in some (probably not optimal) fashion. You MUST define your network NODe callsign using the NRC command. If you are also active in TCP/IP you should set NRCall and MYTcpcall to be the same thing. You MUST use the NPorts Command to enable the ports that you want to be able to access the NODe. For example, NP 3 would allow access to the NODe on ports 0 and 1 but no others. You SHOULD define your network NODe mnemonic identifier using the NRI command. You MUST enable the NODe using the NETNode ON command. You should probably turn AUTONode and AUTODESt ON and set OCM to 3.

This should put your system into full service automatic mode as far as the network NODe is concerned. It probably will not be the way you will want to run but its a starting point.

To REMove a DEStination NODe completely, use

```
REMove DEStination nodecall
```

To REMove a DEStination for a given neighbor NODe, use

```
DESt DELete DEStcall neighborcall
```

To REMove a neighbor NODe (and all the DEStination routes through this neighbor NODe), use:

```
NODe DELete nodecall port#
```

To add a new neighbor NODe, use

```
NODe ADD nodecall port# quality [digicall1 .. .  digical8]
```

Note: The mnemonic identifier for the neighbor NODe will be obtained from the first nodes broadcast heard from the NODe.

Use of the digipeater calls is optional. Although you can specify up to 8 digipeaters, rarely will more than 1 or 2 be practical to use.

To add a new DEStination for a given neighbor NODe, use

```
DESt ADD [destid:]destcall neighborcall neighborport obs# quality#
```

Notes:

- The neighbor NODe callsign must be defined before any DEStinations can be added for it.
- destid or alias for the DEStination is optional.
- obs# is the obsolescence count, 0 locks the DESt in for the NODe.

To change the quality of an existing neighbor NODe, use:

```
NODe REPlace callsign port# newquality [digi1 ... digi8]
```

To change the obsolescence count and quality of a DEStination through some neighbor NODe, use:

```
DESt REPlace DEStcall neighborcall neighborport newobs# newquality
```

One Possible Network NODe Setup

A NETNode can be run in a moderate mode that is somewhere between fully automatic and low profile. Here is some basic info that chould be used: Computer is an old original IBM PC 4.77 MHz clock 640 K ram, two 10 M hard drives, 4 serial cards and a printer card.

Port	Freq.	
0	223.7	High profile (150 W, beam at 80 ft . aimed west)
1	145.01	Quite high profile (200 W omni ant at 90 ft.)
2	145.05	Low profile (10 W omni ant at 60 ft.)
3	80 M	
4	40 M	
5	20 M	
6	serial connection to another computer	

Nearby existing NetRom nodes:

Port	Call	Distance
0	WB8CQR-10	15 (Miles)
1	WB8CQR-1	15
1	K8EIW-1	45
1	W8QLY	75
1	W3LIF	80
2	KB8AST-1	25

Nearby NOS stations using NetRom interface:

Port	Call	Distance
1	N8HSP-4	10
1	WB8LYJ-4	15

All of these stations are usually reliable connections. Other NetRom nodes are usually heard but are not always reliable. During band openings many more are heard, some are reliable during the band opening.

NODE RELATED PARMS in MSYS.DO

Here are the network NODe related parameters I have in my MSYS.DO file:

OCM 3	Transmit DESt nodes heard that have obsolescence counts 3 to 6
NRI 460406	Right 3 bytes of my IP address [44.70.4.6] expressed in hex (quite useful this way)
NRC WA8BXN-2	Callsign net NODe responds to. MUST be the same as TCP/IP callsign (MYTcpcall)
NPorts $47	Network NODe uses ports 0,1,2 and 6 only
PQuality 100	Default port quality 100. A low value assigned to nodes not manually entered that connect to mine
NODe REStore	Read saved NODe file when MSYS starts
AUTONode OFF	Do NOT record NODe broadcasts from nodes not already in neighbor lists. This means that new nodes heard (during band openings for example) will not be automatically added, unless they do actually connect to my NODe.
AUTODESt ON	DEStination NODe broadcasts (from nodes already in my neighbor table) will be recorded.
NETNode ON	Enable network NODe

Here is the output I get from the JN Sysop command (or R command in the NODe):

Port	Neighbor NODe Call	Quality	DESts	Heard	Digipeater(s)
1	AKR:K8EIW-1	179	66	23:43	
1	CLE1:WB8CQR-1	180	67	23:41	
0	CLE220:WB8CQR-10	180	67	23:34	
2	PRYTWP:KB8AST-1	192	2	23:25	
1	YNG:W8QLY	180	7	23:36	

The general philosophy I am using is that I only want to attempt to connect to network nodes that I have a good path to. These are also the nodes that I broadcast myself (and the DEStinations they claim they can handle). I have manually entered these nodes into my neighbor NODe table (using NODe ADD). I see little sense in saying that I can really connect to a NODe I hear once because a meteor hit at the right time! This method does not particularly limit the number of DEStination nodes my NODe knows about because the nearby nodes claim they can handle everything they hear. I just let the nearby nodes worry about finding routes to everything they claim to hear. My system serves as a link from the one isolated NODe on 145.05 LAN frequency to the nodes on 145.01 which give access to the Ohio 4800 Baud UHF backbone frequency. I also serve as an IP switch to and from the network. I have adjusted the neighbor NODe qualities to favor nodes (primarily TCP/IP stations) to which I am a good route. I have set most of the normal NetRom nodes to a lower quality value so that they will talk directly to each other most of the time rather than trying to go through my NODe. Should they be unable to connect directly to each other, they will still be able to go through my system.

If you want to run very low profile in terms of getting used as a network NODe, leave the default of 10 set for OCMin. This means that your NODe broadcasts will only send out your own call. This will get the network to recognize you but not make it think you are a good relay to anything. Keep AUTOD ON so that your system will know how to reach nodes in the network. Its probably a good idea to set AUTON OFF and manually add the calls of the best nearby nodes you can connect to (one is all you need).

If you are totally confused about the network operations, try to consult with who ever runs your local NODe(s). Most of the parameters you have, they have (with some changes in their names).

Miscellaneous Comments on Network NODe

If you are low on SSIDs, you might consider the following. Set the IDCALL to your callsign without any SSID. Do not define a BBS, Keyboard, or K NODe callsign. Pick an SSID and use your callsign with it and set as the value for MYTcpcall, NRCall and FORWARD call To get to your BBS the network NODe BBS command can be used. To talk to your keyboard the BBS T command or network NODe T command can be used. If you want to make outgoing calls you will have to define a keyboard callsign.

Should you use the network NODe to route TCP/IP frames? I would say only if you have to! The normal IP routing is better than using the network NODe but in those cases where using the network is the only way to reach some distant IP station then do use the network.

Network node connects are allowed only on those ports enabled by NPorts command.

Kanode which lists KANodes that have been heard. An optional parameter may be given that specifies the port for which the list is desired.

Justheard which lists stations recently heard. Here again a port number may be given as an argument. The N command is now described as Netnodes (lists destination network nodes known).

In the network node, using the C# form of the connect command (example: C2 WA8BXN) will always result in a normal AX.25 connect, even if the destination station is a network node.

The network node does not record information in nodes broadcasts heard on ports that are not enabled for network node use (NP command).

NET NODe PARAMETERS SUMMARY

(PARM	Default	Comments)
PQuality [port#] #	192	Quality for nodes heard on port
BInterval seconds	3600	Nodes Broadcast interval
OCMin #	10	Minimum obsolescence to broadcast
OCInit #	6	Initial obsolescence count
WQupdate #	1	Worst quality DESt route to record
NTtl #	64	Network time to live (max hops)
TTimeout seconds	60	Transport timeout (Net FRACK)
TREtries #	3	Transport retries
TACK sec	3	Transport ACK delay (RESPonsetime)
TBdelay seconds	180	Transport busy delay
NBdcast [port#]	ON	Enable or disable node broadcasts on a per port basis
NDwindow #	4	Net default window size (frames)
NETCon	ON	Enable connect cmd in net NODe
NETBbs	ON	Enable BBS cmd in net NODe
NETTalk	ON	Enable Talk cmd in net NODe
NETXBatchsize [port] #		Sets max batch size for RLI compression on port
NETXOption [port#]	OFF	If on, enables RLI compressed forwarding on port
AUTODESt	OFF	Add new DESts for known neighbors
AUTONode	OFF	Add new neighbors heard
NODe SEnd		Force known DESt transmission
NODe SAve		Write net NODe Database to disk
NODe REStore		Read net NODe Database from disk
NODe ADD call port qual [digis]		Adds neighbor NODe
NODe REP call port qual [digis]		Changes quality of neighbor NODe
NODe DEL call port		Deletes neighbor nodes & its DESts
DESt ADD [id:] destcall neighbor call neighborport obs# qual		Adds new DEStination
DESt REP destcall neighborcall neighborport obs# quality		Changes obs & quality
DESt DEL destcall neighborcall neighborport		Deletes DESt through given neighbor
REMove DESt callsign		Deletes DESt through all neighbors
NInfo		Gives net NODe status
NRCall callsign		Sets net NODe callsign
NRId mnemonic		Sets net NODe mnemonic (alias) identifier
NPorts (MASK)		Ports to which nodes broadcast are made
TNatime seconds		Transport no activity timeout time
C BBS or C bbscallsign		Acts like BBS command taking you to the bbs. You also now get a "connected to" message with either of these commands.

A number of commands allow setting options for network connections. Here are the network command names and their AX.25 port related counterpart.

NETOnly	BPorts
NETRead	MAXRead
NETMax	BMaxcon
NETFBBStime	FBBStimes
NETFBBC	FBBC
NETFBBG	FBBG
NETXBatchsize	XBatchsize
NETXOption	XOption

PACTOR
GTOR
OVERVIEW

PACTOR operates somewhat like a enhanced AMTOR. A series of characters is received and an acknowledgement is sent by the receiving station. PACTOR has a full character set that make it look like a typical packet connect. In addition it has a number of features that enhance its performance such as dual speeds, internal compression and internal memories that enhance reception.

GTOR is the KANTRONICS implementation using the Golay error correcting code.

Neither PACTOR nor GTOR share a channel with other users. They can not be used for automated forwarding. Our own experience utilizing the modes while camping in very poor conditions shows that GTOR is a far superior mode of digital communication, with PACTOR as a close second choice.

HARDWARE

MSYS supports only the KAM Plus or KAMs that are upgraded to KAM Plus TNCs. The KAM is the only TNC that gives separate commands to allow you to seize the transmit direction and to go into receiving mode. Other TNCs provides only toggles that lead to uncertainty about the state of the function being toggled.

When used with MSYS, the KAM Plus can only be used as one port dedicated to PACTOR or GTOR. You can not do KISS or access the VHF side of it.

The CTS line must be present in the RS-232 cable. The radio must be capable of PACTOR/GTOR operation. Some older radios will not switch from transmit to receive fast enough. (Also some newer ones switching between vfos or modes!) Additional cooling will probably be necessary. In addition, a larger than normal power supply may be necessary. These are points which should not be overlooked.

500 Hertz receive filtering is useful. Our own experience has shown excellent results using minimal power into poor antennas when working home. 25 watts into a mobile whip is more than adequate to a home station running 100 watts to a dipole over a 400 mile path on 40 meters ... 24 hours a day.

PACTOR/GTOR SET UP

The set up change is minimal. The name of the port in the MSYS.OPT file must begin with "PACTOR" or "GTOR", depending on the mode desired. For example:

```
PACTOR_40
or
GTOR_HF
```

would be appropriate.

PACTOR TNC PARAMETERS

The following KAM commands must be given to set the KAM into the proper state:

```
MYCALL <YOUR_CALL_HERE>  This is the callsign that will be used on
                         the AIR.  The BBS call MAY be different!
                         Observe your License Class privileges!
MYPTCALL  <YOUR_CALL >   This is the call that the KAM will respond
MYGTCALL  <YOUR_CALL >   to on the air
CWPTT ON
CWSPEED 20
```

The MSYSTNC# (TNCTYPE) file for PACTOR should then contain:

```
control-c X
control-c d
control-c d
PBBS 0                   (The TNC will reset at this point)
INTFACE TERM
XFLOW OFF
FLOW OFF
CRSUP OFF
PREKEY 0
PTHUFF ON
PMODE PACTOR
DELETE 0
SPACE 2295
MARK 2095
SHIFT MODEM
ECHO OFF
PACTOR
```

For GTOR, the MSYSTNC# (TNCTYPE) file should contain:

```
control-c X
control-c d
control-c d
PBBS 0                    (The TNC will reset at this point)
INTFACE TERM
XFLOW OFF
FLOW OFF
CRSUP OFF
PREKEY 0
PTHUFF ON
PMODE GTOR
DELETE 0
ECHO OFF
GTERRS 80
SPACE 2295
MARK 2095
SHIFT MODEM
GTOR
```

(The KAM Plus does not need the PERM command to remember its settings.)

PACTOR MANUAL CONNECTS
GTOR MANUAL CONNECTS

To connect manually to another PACTOR/GTOR station, the standard command "C# callsign" is used. You will not have keyboard access until the connect is made or fails. The standard "^CD" is used to disconnect although if your connect accepts a "BYE" command or the like, that is preferable. After the connect is closed, there is a short delay (about 10 seconds) where you again lose keyboard access.

PACTOR FORWARDING
GTOR FORWARDING

Automated forwarding should not be done using PACTOR/GTOR because of the HF "attended" rules. However, reverse forwarding can move traffic off a remote PBBS. A MSYSFWD.DEF-like file is made with a filename of your choice. This will serve as a manual forwarding file. You can not have the $filename.ext type of include file. The following is an example:

```
p0 no8m
wa8bxn
wb8bii
k8eiw
-----
```

The above causes a connect to NO8M and sends traffic for the three listed PBBS stations. A reverse forward prompt then results in traffic for the connecting station to be reverse forwarded.

The following is another example:

```
p0 no8m
home
wife
wa8bxn
wb8bii
k8eiw
no8m
oh
usa
-----
```

The above file connects to NO8M from a remote version of the NO8M PBBS. It sends traffic for stations who use the home NO8M to the HOME route. It sends traffic for KA8ZVV to her computer which is attached via a RS-232 cable. The other forwarding is standard. When you wish to forward, you utilize the following command:

```
FP filename.ext
```

MSYS then acts on your file and forwards via the PACTOR or GTOR port.

USER'S VIEW WHEN CONNECTED

When connected to a MSYS PACTOR/GTOR port, the user may be confused by the fact that the link direction is controlled by MSYS. However, other than that, the user will see a port identical to a packet port. The user should not attempt to control the direction of the link.

PACTOR LIMITATIONS
GTOR LIMITATIONS

No YAPP is currently possible on a PACTOR/GTOR port. You also should not do FBB-type compressed forwarding. No particular advantage is seen when using the FBB group forwarding. DO NOT ENABLE DIGI ON THE PACTOR/GTOR PORT!

PACTOR TROUBLESHOOTING
GTOR TROUBLESHOOTING

A power supply of ample capacity for your radio is necessary. Some problems can be traced back to unstable or minimal power supplies. The use of the TERM command to go in and talk to the TNC is valuable. The TNC can be interrogated and connects can be established without regard to MSYS settings by using TERM.

One initially confusing aspect is the use of "ECHO OFF". MSYS requires echo to be off. When in the TERM command mode, you may wish to turn echo ON so you can see what you are typing. Before you leave the TERM command mode, you must remember to again set ECHO to OFF.

MSYS PACTOR SCANNING
MSYS GTOR SCANNING

By instituting scanning, your HF radio can be made to cover many more frequencies and, due to that, more area. The costs are minimal. A TTL-to-Serial level converter and a serial port is required. Although the serial port would be a lot to ask for most of us, the port is configured in such a way that there is no need for an interrupt or normal port address. The memory required is under 1k.

The port does not need an interrupt due to it never needing to read data. Data is sent to the radio but there is no need to read data from it.

It is possible to build a simple interface to allow the computer to choose one of four antenna lines.

IOBUS.DOC

A separate file is included with the MSYS archive that documents what lines on a serial card are used for what purpose. You may want to review that file as it will be referred to often. It will also help to have documentation for the board you are using.

INTERRUPTS

You can give it a interrupt if you can spare one. You will be wasting it, though, as it is not used. If you can not spare an interrupt, the first thing to do is to insure that no interrupt is enabled on the card. You may have to cut traces, set switches or do whatever you need to do to make sure the interrupt is not enabled.

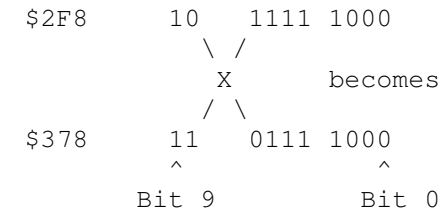
In most cards, you have two switches that connect either INT 3 or INT 4 to the electronics on the card. In some older cards, this is done with a jumper. In both cases, turning the switches off or pulling the interrupt jumper will cause it to be disabled.

There may be an instance where a interrupt is hard wired to the electronics. You will need to cut that trace using a knife or a hand held rotary grinder.

ADDRESSING

You will need to set the address jumpers or switches to address the board as COM2. This sets the address lines to the address $2f8. By cutting and soldering jumpers between two of the address lines, you can re-address the board to be $378. In this way, you can save the addressing space.

This is done in this way:



By cutting address bit 7 and address bit 6 and crossing them with a jumper, you have re-addressed the serial card to recognize address $378.

You need to cut two traces, the traces that lead to finger A23 and A24. Refer to the file IOBUS.DOC. Finger A23 is soldered, via a jumper, to the line that used to run from A24. A24 goes to the line that used to run from A23.

DSR LINE

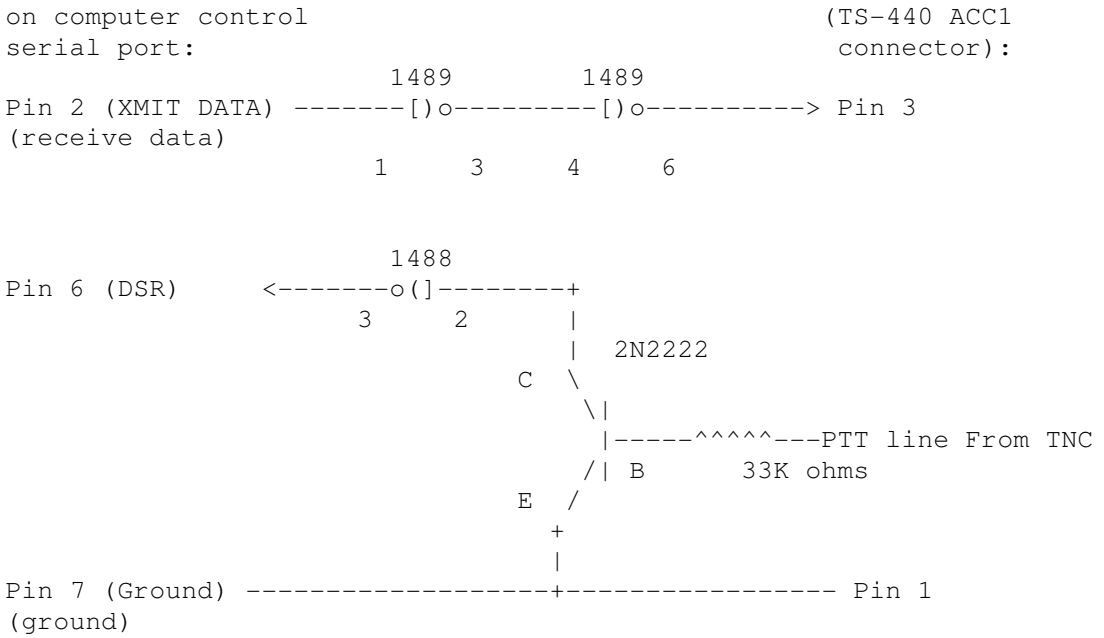
MSYS checks a line in the serial port called the DSR line. The DSR line is attached to the PTT line through level converters. This line will tell MSYS when the radio has transmitted. The radio transmitting will indicate that the scanning must stop.

The PTT line from the KAM+ TNC is near TTL levels(0 or +5 volts). It must be changed to RS-232 levels (+12 or -12 volts). This can be done via a level converter. When the TTL-to-RS-232 level converters are discussed, the additional wiring necessary will be covered.

We determined that many RS-232 adapters, such as an adapter to take a 9 pin plug to a 25 pin socket, are improperly wired when it comes to carrying the DSR line through the adapter. If the radio does not stop scanning, you may want to place a RS-232 LED box, such as Radio Shack 276-1401, in line to verify you are seeing the DSR change.

TTL TO RS-232 INTERFACE (1488/1489 version)

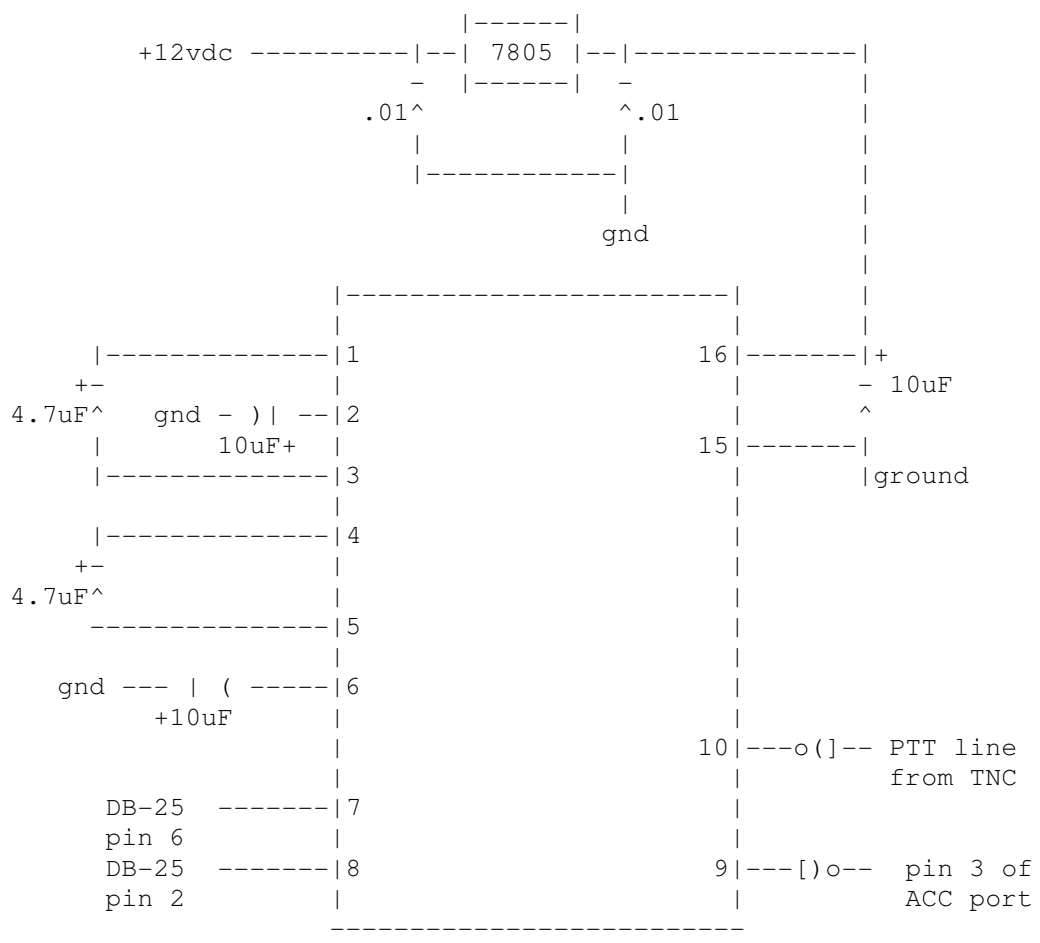
Supply voltages:	+5V	+12V	-12V	Ground
	1488	14	1	7
	1489	14		7



Note: []o is a level converting NAND gate
The arrow on the emitter of transistor is not shown.
Power supply design is up to you (not much current is needed).

TTL TO RS-232 INTERFACE (MAX232)

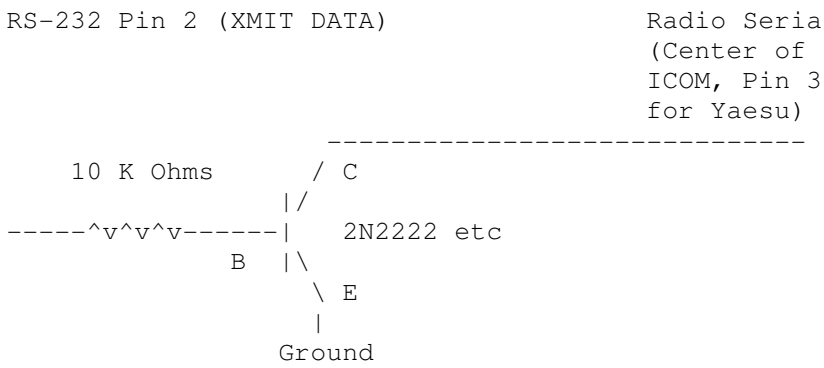
The MAX232 chip allows you to do the same level changes.



o(] and [)o are 7400 NAND gates

NOTE: JDR (800-538-5005) is advertising a newer version of the MAX232 which requires no capacitors.

For ICOM and Yaesu radios, the following circuit may be used from the computer to the radio:



You now need to set up the files that will be necessary. You need to run SCANDEF. SCANDEF will access a ASCII file called PORTn.DEF and modify the scanner. It will also define the scanning frequencies, the delay on each frequency before moving to another one, the antenna that will be used and the time that the frequency will be scanned.

The "n" in PORTn.DEF is a number you supply to define the port. If your port is to be number 6, you would call the file:

PORT6.DEF.

At least 4 lines are required in the PORTn.DEF file. In order, they are:

TSR xx	where xx is the interrupt number of the scan TSR, usually D2
PORT yyy	where yyy is the hex address of the control serial port
SPEED zzzz	where zzzz is the baud rate of the radio's computer interface

freq delay antenna starthour lasthour

The last line type may be repeated up to 25 times. It contains the frequency in MHz or KHz (14.235 or 14235 for example), the time to spend on this frequency listening for a connect attempt in seconds, the antenna number (0-3), the first hour to scan this frequency, and the last hour to scan this frequency. Only the frequency is required. Defaults for the other values are 5 seconds, antenna 0, start hour 0, and last hour 23. If you give any parameter beyond frequency, you must specify all parameters to the left of it.

Here is an example file (PORT1.DEF it might be called if port 1 is the PACTOR port):

```
TSR D2
port 378
speed 4800
3532.1
3737.1  10   1  0 8
3737.1  10   1  20 23
7169.6
7174.1
```

This would scan 3632.1 and the two 40 meter frequencies all day. 3737.1 would be scanned from 8 PM to 8 AM only, listening 10 seconds before going on to the next scan frequency (and using antenna 1). Note that these frequencies are NOT appropriate for PACTOR Operation!

TURNING IT ON AND OFF

To enable scanning type (or better yet put in MSYS.DO) "PS n ON" where n is the PACTOR/GTOR port. Repeat this for each additional port. This command reads the appropriate PORTn.BIN file storing the frequency parameters in memory and begins scanning.

To disable scanning type PS n OFF, where n again is the port number.

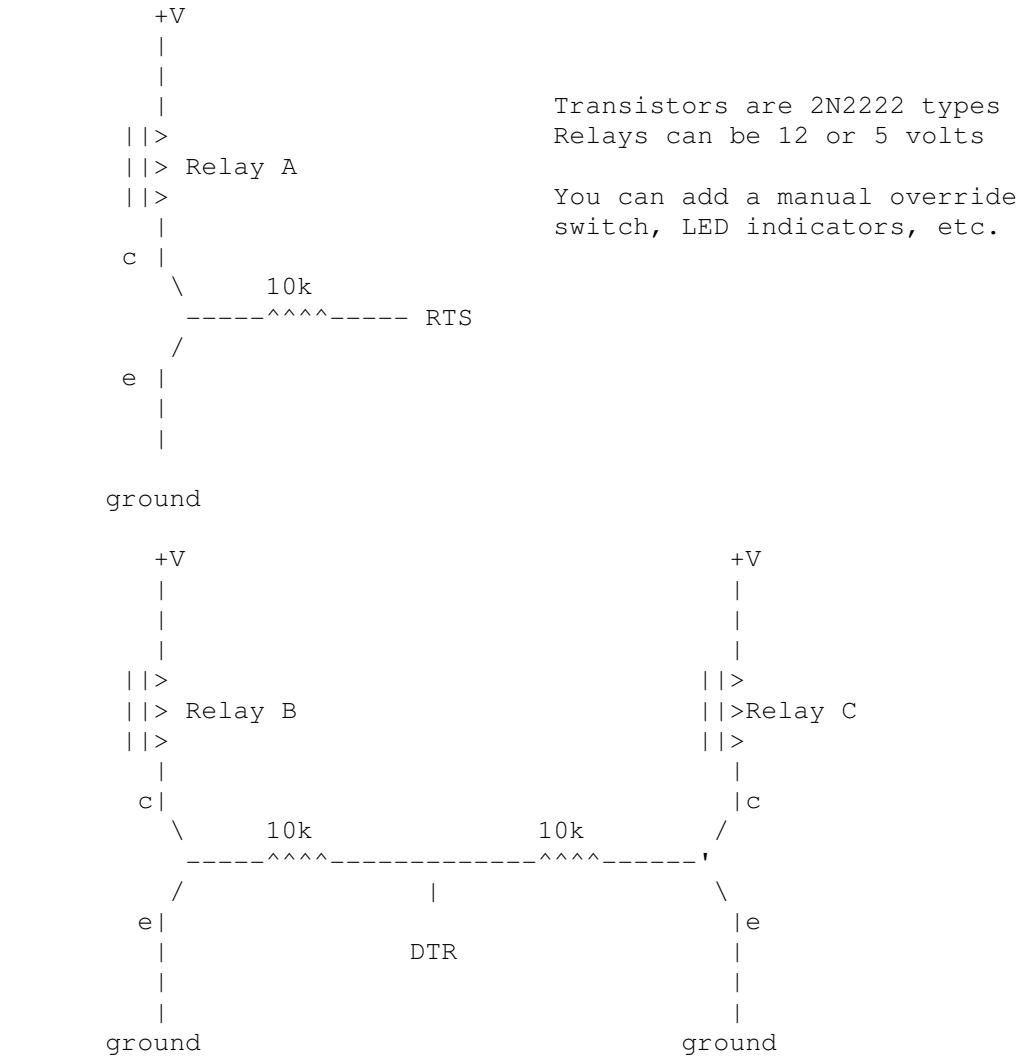
The PS command with no operands gives basic information about PACTOR scanning such as which ports are enabled, current frequency and number of frequencies being scanned. PS with a port number as a parameter gives a listing of the frequencies scanned on the given port.

The command PS n freq will tune the radio on the given port number n to the specified frequency. This frequency need not be in the scan list. Using this form of the command also disables scanning. The frequency can be given either in MHz (7.0991 for example) or KHz (7099.1). This form of the command is useful before making a manual connect (using the C# command) or initiating manual PACTOR or GTOR forwarding (using the FP command). Note that this command can be imbedded in the forward file (\ps 2 3541.2 for example). To resume scanning, use PS 2 ON, either manually or at the end of the forward file.

If you have enough memory, you can run the SCANDEF program described below using a command like "RUN SCANDEF PORT3" while MSYS is running. To see the output after the program runs, type "TY $POOL". (The RUN command will redirect any screen output to a file called "$POOL". You can read that file to determine the program's output.) Then use the "PS 3 ON" command to load the new frequency specifications. This could all be done as a remote Sysop.

RTS and DTR outputs from the serial port are set by the antenna number as follows:

Ant#	RTS	DTR
0	0	0
1	0	1
2	1	0
3	1	1



THE LAST STEP, THE TSR

Before running MSYS, an appropriate SCANTSR program must be run, such as SCAN440.COM. This is best done in your AUTOEXEC.BAT file. The SCAN440.COM program is designed to be used with the Kenwood TS-440. Other SCANTSR programs may be found on the distribution disks, including programs for the ICOM IC-735 and the Yaesu FT-757.

USING OTHER RADIOS

As distributed it is set up for a Kenwood TS-440. Source code is provided (SCANTSR.ASM) that can be modified for other radios. If you do modify it for some other radio and are sure of your modifications, send WA8BXN a copy so that it can be included in future releases. The SCANTSR program is called by MSYS when it needs to tell the radio to go to a new frequency. It is called as an interrupt passing the control serial port address in DX and the new frequency as a 32 bit number (long integer) in registers AX and BX. AX has the most significant 16 bits. The TSR simply formats whatever command the radio needs to go to the given frequency and then sends that data to the radio using the given control serial port address. The control serial port is already initialized.

EXAMPLE OF FP COMMAND FILE
Here is my "STEVE" file so that when I want to forward to you I just give the command "FP STEVE":

```
p0 no8m
no8m
usa
can
allus
---------
\ps 0 on
```

The scanner command will work only in FP files, not MSYSFWD.DEF. To add a scan file, you need to put a line of dashes prior to the command. The purpose of the above command is to cause scanning to resume at the end of forwarding. In addition, to reverse forward to anyone, like KP4GE above, two things are necessary. First, a REV file in the FWD directory. Second, the callsign of the station connecting must appear as an entry in the MSYSFWD.DEF. Should a sysop never want to forward to the station who will be reversed to, a dummy entry such as:

```
-----
P6 KP4GE
!25
$fwd\kp4ge.rev
-----
```

must appear before the station will be reversed to.

MSYS GENERAL INFORMATION

BBS ONLY ACCESS

Lets say you are running with ports 0 through 3 and want to make port 2 bbs only access with at most 1 connection allowed at a time (either another BBS forwarding to you or you forwarding but not both at the same time). You also don't want to generate MAIL FOR beacon on port 2. Use the following:

```
FB 2 25          turn off mail for beacon port 2
BP 4             port 2 bbs only
BM 2 1           max 1 connect port 2
MCF $B           no simultaneous forwarding on port 2
```

Note that BP and MCF both have port masks as arguments.

CONNECTS FROM NON-AMATEUR CALLSIGNS

Connects are no longer accepted from non-amateur callsigns (like PK-232 and NOCALL for example). You can still make connections to non-call signs.

COMMAND

Command lines beginning with # on the BBS are ignored (they sometimes come from KA Nodes).

CHANNEL NUMBERS

The channel numbers use a rotational assignment rather than always starting from 0

DEVICE ERRORS

Device errors are trapped by MSYS (you would previously get the Abort, Retry, Ignore? message). The first occurrence of the error is given a response of retry, the second ignore. This will normally be caught by particular function being performed which will terminate with an error message, but MSYS should continue operation.

DISK FULL MESSAGE

If the disk is full when a message is sent to the system it will give a message to the connection station that the disk is full and then disconnect.

DOS

If you are using MSDOS 3.3, you may need to put STACKS=0,0 in your config.sys file. MSYS has been run under MSDOS 5.0 and MSDOS 6.0 as well as DRDOS 6.0 with no known problems or special commands other than FILES=40 and BUFFERS=20. With MSDOS 5.0 and MSDOS 6.0, you may chose to use SMARTDRV.EXE instead of the buffers statement. It all depends on your personal preference and "RAM CRAM".

END OF LINE BELLS:

When you type past column 72 you will get a beep for each character. Keep those lines less than 80 characters!

EVENT.LOG

In the event that a problem arises with forwarding of a message or if MSYS is unable to open a file, an entry is made in a file EVENT.LOG showing the problem and the time of occurrance. The intent is to provide a clue should other strange things happen. This file may generally be deleted at any time.

HEADERS

To create an ASCII file containing all the active message headers, use the following sequence of commands on the bbs:

```
X 0       (turn off the More? message temporarily)
>msghdrs  (this is the name of the file to be created)
L$ 0      (or L 0 if you don't want to see the bids)
>         (close the file)
X 20      (restore the More? message)
```

KEYBOARD

When someone connects to your keyboard answer call you will get two musical tune to alert you to the connect. Any stations connecting to your keyboard call when you are already connected to another station will get a busy. The actual number of connects allowed before busy is given by the SYSOP command MCon. MCon 1, for example, only allows an incoming call to the keyboard to be answered if there are no other keyboard connections. MCon 0 prevents all incoming keyboard requests (they get busy). MCon N will allow N connections to the keyboard until busy is returned. When you type past column 72 you will get a beep for each character. Keep those lines less than 80 characters! When a station attempts to connect to your keyboard via the several ways possible but is sent a busy the bell will also beep to alert you to this.

***LINKED

A line of the form ***LINKED to callsign is recognized as a connection by the given callsign. This occurs with some versions of RLI systems when a user uses the C command on the RLI system to connect to the MSYS BBS. ***LINKED to callsign requires SYSOP verification if SYSOP call is used.

Messages sent, forwarded, and killed get logged to MSYS.LOG.

A new log file record is being generated. It is identical to the BBS record except it has BYE in place of BBS. It is generated when a user exits the bbs by any means (disconnect, timeout, B cmd). You may wish to put an entry in the MSYSTODO.DAT file to rename MSYS.LOG periodically (weekly?) as the log file CAN get very large, very fast!

LOCKUPS

If the system hangs,an attempt is made to reboot. Thus you should make sure you don't have a diskette in A: and that you have an appropriate AUTOEXEC.BAT file set up. It is strongly recommended that any automatic restarts include the MUTIL 4 command to insure that the mail file will be properly indexed. If the mail files are not properly indexed, MSYS will most likely hang on restart.

MAIL BEACON

Messages to your IDCALL and to SYSOP are not displayed in the MAIL FOR beacon. You will see them on the screen.

MAIL.BAK

To reduce disk read/write delays when a message is received and copied to the MAIL.BAK directory, subdirectories are automatically created in the MAIL.BAK directory. 100 messages are backed up in each directory. This can substantially reduce the time it takes to make the backup copy of incoming messages because much smaller number of file names must be looked at to add the new file name to the directory. The created subdirectory names are of the form MSGxxx where xxx is the message number divided by 100. Thus in MSG025 messages 2500 to 2599 would be found. The message /RESTore function from the BBS has been modified to understand these changes.

MAX channels AND PORTS

Maximum for MAXCHANS is 50 (MSYS.OPT)

Maximum number of ports is 7 (numbered 0 to 6)

MSYS supports all IRQs (0-15) for serial ports. I suggest the use of IRQs 2,3,4 & 7 on PC class computers and 3,4,5 & 7 on AT class and above computers. Use of 01 & 6 are not recommended but are supported in the code. Due to the use of these interrupts by other hardware devices/functions results will generally be bad!

MEMORY MANAGEMENT

Buffer space for logical channels is allocated when the channel is allocated (and released when the channel is released). The MEM command may or may not change results depending on position of the last user in the memory space. If you find that you are seeing ***MEMORY LOW BUSY messages, you probably should investigate the reasons why. There are several things that you may try to do to relieve the problem. The first is to use a upper memory manager to put as many of the TSRs as possible into upper memory. If you use Quarterdeck QEMM, you should run the OPTIMIZE program. When using QEMM, you may wish to NOT use DOS=HIGH in the CONFIG.SYS.

Another suggestion is to use the Quarterdeck program VIDRAM with the "ON" switch set. VIDRAM will utilize the memory just above the 640K boundary that is normally used for graphics and add it to the available program space.

While the default MINmem is 32000, it can be lowered slightly WITH CAUTION. If MINmem is set too low, you do run the risk of a user (INCLUDING THE SYSOP!!!) logging in and utilizing more memory resources than are available, causing immense grief and extra effort to clean up the damaged files. A 386 system with 6 active ports and the SAM callsign database server using DOS 6.2, QEMM 7.05 with VIDRAM ON, running under DesqView 2.4 starts out with approximately 100000 bytes free. It seems to be enough to keep up to 14 channels working with very few "busys". If that still isn't enough, you may wish to either try another multitasking system (Windows?) or run MSYS not under a multitasking system.

If all that fails, or you just have a system with no expanded or extended memory, you need to seriously consider your MSYS configuration. If you reduce the maximum number of messages (MUTIL 1) to a smaller number, you will increase the available memory for users. You may choose to limit the number of active channels. If you are experiencing low memory problems, you may also wish to disable the conference feature. While it is possible to reduce the size of the various memory stacks used by MSYS, extreme care is needed to avoid serious consequences. See the section on STACKS for further details.

Another alternative is to provide a minimum service system by running the "small" version of MSYS, which is distributed as MSYSB.EXE. The 'B' version does not provide all the fancy NODE OR TCP/IP features, but is will function quite nicely as a stand-alone BBS working behind or along side of other network switches. Probably the major loss to the packet community is the fact that this version will not show up in a NetRom node list. You will still need the main distribution disks for all of the documentation. MSYSB.EXE is the executable version that replaces the "big" MSYS.EXE

EXPANDED MEMORY USAGE

MSYS will automatically detect if either EXTENDED or EXPANDED memory is available. If sufficient EXPANDED memory is present (at least 320K must be available to be used) then all stacks will be around 16000 bytes and come from expanded memory. This should free up about 50K or so of memory as shown by the MEM command. Also each new connection will require less memory. If you have expanded memory and don't want it used, start MSYS using: MSYS NOEMM

Real expanded memory boards (that could be used with an 8086/8088 processor) will give better performance (faster) than simulating expanded memory. Expanded memory is used (if present) to contain the message headers. Now you can have 2000 msgs on your system and memory to spare. Be warned that with this many active msgs things will slow down.

NOTE: If you have problems running expanded memory try using a different EMM program.

MSYS can also run using EXTENDED memory as well. Of course, you will need to utilize a EXTENDED memory manager such as HIMEM.SYS or QEMM 7.0 or perhaps any one of several other quality memory manager programs on the market. If you have EXTENDED memory and do not want it used, start MSYS using:

MSYS NOEXMEM

AUTOMATIC MESSAGE HOLDING

If you create a file called MSYSHOLD.DAT you can specify characteristics of messages to hold automatically using parameters similar to those used in the house cleaning file. The available parameters are:

```
TO=       wildcard representation of To callsign
FROM=     wildcard representation of From callsign
AT=       wildcard representation of @BBS
SIZE=          number that is size of message
CONNECTED=  exact callsign of sending station (less SSID)
PORT=     inbound port number that message is from
BID=      This is useful for trapping known "improper" BIDs.
TITLE=    If used, this should be the last parameter on the line.
TEXT=     Names file which contains list of "bad words".
```

Normally ARRL bulletins have BIDs similar to: ARLP037. These bids begin with a letter. BIDs automatically assigned to bulletins that don't have a bid usually begin with a digit. So, to hold bulletins that probably don't have the right bid, we could use:

AT=ARRL BID=#*

Note that #* is a wildcard specification that matches anything that starts with a digit.

If the value given for TITLE= is found any where in the title of a message (and all other specifications are met) the message will be held.

There is one other parameter that can be specified, a line containing only the letters:

BBS

All lines ABOVE the line BBS will look at ALL messages coming in for the holding criteria. The lines below the BBS line will only apply to messages entered by a NON-BBS station. For this use, a BBS is defined as a station that transmitted a [...$] line. Therefore, all entries below the BBS line will apply to users of your system..

```
Example:            Comment - (NOT IN MSYSHOLD.DAT FILE!)

SIZE=2500           holds any messages bigger than 2500 bytes
```

```
FROM=WA8BXN PORT=2  holds messages from WA8BXN that come in on port 2
CONNECTED=W8XYZ     holds all messages sent by connected user station
                    W8XYZ
AT=MSYS             holds all messages with @MSYS
TEXT=BADWORDS.DAT   holds all messages that have any words found in
BADWORDS.DAT file.
BBS
AT=ALL*
```

Note: the difference between FROM= and CONNECTED=
FROM is the from call for the message (often supplied after < in the S command), while
CONNECTED is the call of the station connected to the bbs sending the message.

In the above example, a message sent to ALLUS from a local USER would be held, but would NOT be held if entered from a connected BBS.

NOTE: Held messages are visible only to the SYSOP They are killable only by SYSOP. A message will be sent to a user (not a BBS) that the message has been held and made invisible for SYSOP review. The user, at that point, has no further ability to deal with that message.

Other options on the lines in MSYSHOLD.DAT:

* at the beginning of the line makes it into a comment

TEXT=filename operand
When specified, if other tests are met, the text and title of the message will be compared with the contents of the given filename . If any of the words in the file match, the message will be held. Case is ignored in the test.

Example:

TO=ALL TEXT=WORDLIST.1

would cause all messages going to ALL to be compared with the file WORDLIST.1. Put one phrase or word per line in the file, enclosed in either single or double quote marks. You may also put a number between 1 and 255 at the beginning of the line before the quoted phrase to assign a weight to the word. If no weight is given, the word or phrase gets a weight of 255. Here is a sample file of words:

```
'war'
"900"
"send check"
50 'send'
50 "check"
"That's it"
```

If the sum of the weights of these "words" found in the title or message text exceeds HOLdlevel (default 0) then the message is held. NOTE: HOLdlevel is a new SYSOP command!

CAUTION: 'war' would match an occurrence in the word 'forWARd' in a message. Its not a real problem as it just means the message will be held not killed off.

To resolve that "problem", you may include a space either before, or after or both before and after to "isolate" the specific word you wish to hold. A space will also help catch words at the beginning or end of a sentence. For example, if our "bad" word is "dirt", the following:
" dirt " in wordlist.1 will catch the word "dirt" but not the word "dirty" in the sentence: The dirty bird's name is dirt.

If the word KILL is added to the line in MSYSHOLD.DAT, the message will be stored in the mail.bak directory and immediately killed. For example:

TO=DIRT KILL

will kill all messages to DIRT. It is suggested that this form of the command be used with care. The KILL option does NOT work with the TITLE= option.

MSYS.DO

MSYS.DO is a file which you should create to do all of the SYSOP commands needed to prepare your specific MSYS system. This file is processed as a series of commands and is far easier to do rather than manually typing them every time you start MSYS. When the system starts, it automatically tries to do a DO command. See the sample file for more information on the possible contents.

MSYSTODO.DAT

Timed execution of commands can be done in MSYS. You can create a file called MSYSTODO.DAT with commands and times they should be done. Wild cards can be specified for the times. Each line consists of a time (given as exactly 8 characters) followed by a space, and then the command to execute at the given time. The wildcard character is ? and they may only occupy positions in the left part of the time . The format of the time is MMDDhhmm where MM is the month number (1 to 12), DD is the day of the month (1 to at most 31), hh is the hour (0 to 23) and mm is the minute (0 to 59) . Here are some time examples:

MMDDhhmm (pattern)
???????? every minute

```
???????0  every 10 minutes (times that are like 00 10 20 etc.  for
          the minutes)
??????00 at the beginning of every hour
????1200 at noon every day
????0000 at midnight every day
???10000 at midnight of the 1st, 11th and 21st days of each month
??010400 at 4 am the first of every month
01010000 at the beginning of every year
```

You can also specify a day of the week in the DD field, in place of the day of the month . To do this you would put D# in the DD field (# is from 1 to 7 for Sunday to Saturday) . Examples:

```
??D11200 Every Sunday at noon
11D52300 At 11 PM on each Thursday in November
```

That is how the times are set up, now here is a complete example of things that could go in MSYSTODO.DAT:

```
????0000 nod sav
????1200 nod sav
??010000 log
??d10000 copy msys.usr msysusr.bak
???????0 dt
```

This file would save the network node list twice a day, make a new log file at the beginning of every month, make a backup of the user file every Sunday, and display the time and date every 10 minutes.

MSYSTNC.#

If you use any TNC set up files be warned that the program waits 1 sec after each line from the file is sent to the TNC to give it time to think about it.

NOTE:
AEA TNCs made after Jan. 89 can probably be treated like KPC-x TNCs to get them into kiss mode. It should no longer be necessary to have a special TNC file for them (but if it doesn't work without it, it is suggested that the TNC file be used). A good solution to the TNC to kiss problem with MFJ 1270/4 TNC-2 compatibles is to install the JKISS EPROM available from Tucson Amateur Packet Radio. The EPROM forces the TNC to ALWAYS be in the KISS mode, and no further action by MSYS (MSYSTNC.# file) is needed.

Special KISS Parameters

There are 4 parameters that may be set either before the TNC is put into the KISS mode with the MSYSTNC.x or after by commands from the MSYS.DO file.

They are:

TXD
TXT
PPersist
SLottime

It is probably best to set them in the MSYS.DO file so that you have the ability to determine what the value is in your TNC by examining the parameter from the keyboard. If MSYS has NOT set the parameter after the TNC has gone into the KISS mode, it will show -1 as the value.

Suggested values are TXD 30, TXT 5, PPersist 64, SLottime 10.
Your mileage may vary.

Packet size

Data in packets is limited to 256 bytes. Larger amounts of data in a single packet will be discarded without causing the system to crash.

PASSWORDS

There are several PASSWORD types utilized in MSYS.

REMOTE SYSOP

First, there is the remote SYSOP password. The phrase for this is stored in a file called MSYS.RMT in the C:\MSYS directory. to you.

/Sysop command: CASE SENSITIVE!

This command gets you into remote SYSOP mode which allows you to use most of the SYSOP commands through the bbs. The command must be typed exactly as shown: backslash, Capital S and lower case ysop and then return. You will then get a line like

Usage stats: Users=4 Msgs rcvd=10 Msgs sent=5 Max=8

You MUST respond with the EXACT characters needed before SYSOP privileges will be granted. To this you must reply (there will be no prompt) with the four characters corresponding to the numbers typed in order of characters found in the file MSYS.RMT (case is important). For example, if your MSYS.RMT file contained MSYS is the greatest! then you would type <space> e i t which are the corresponding characters (the first character is numbered 0). Thus you would type: eit (Don't forget to count the spaces) You should now get a prompt cmd: If you are connected via radio, you issue /Sysop command on bbs to verify SYSOP authority (enter 4 letters) You then get the next bbs prompt (if you give correct letters or not). If you have given the correct reply then you may issue any SYSOP command by starting the line with \ (while still in the bbs). So when on the bbs from the local keyboard, you can now type something like \em 1234 and you will be able to edit message 1234 without having to leave the bbs. As a remote SYSOP, once you give the right 4 letters, you retain SYSOP status until you disconnect. To get a list of available commands, type h. Most of the commands available duplicate those found normally as SYSOP commands. An added command, WHO, gives a list of who is on the system and is the same as the new U command in the bbs.

NOTE: /Sysop is not required from local keyboard.
Make sure you mark yourself a remote SYSOP, even if you won't be one.(User flag $10)

AUTOMATIC PASSWORD REPLY

If you are running a remoted MSYS system (you DO go on vacation, don't you?), you may automate that return process by having a file called PASSWORD.RMT in your C:\MSYS\FWD directory. Likewise, this is a convenient way for your neighbor SYSOP to manage your system while you are absent.

If your system receives the password prompt (the "Usage stats" line) from a system that you connected to during forwarding or from the Connect commands, MSYS will automatically supply the correct response. You must have a file with the <callsign>.RMT in your FWD directory that is the same as the MSYS.RMT file on the other system. If you were to automatically sysop the WB8BII system, you must have a c:\msys\fwd\wb8bii.rmt file that is identical to the C:\MSYS\MSYS.RMT file on WB8BII. You may have several c:\msys\fwd\<call>.RMT files to enable you to sysop several systems.

NOTE: If you are using MSYS to remote SYSOP a distant bbs, it will respond ONLY ONCE per initial connect! For example, if you are using MSYS to SYSOP a remote system from the campground, your local system will respond to the "Usage..." statement the first time. If you disconnect from the remote bbs and drop back to a node, then reconnect to the remote system, your local MSYS will NOT respond to the "Usage..." statement. You must disconnect completely and then re-establish the connection to the remote system for the automatic password response to be given. This is done to help prevent repeated automatic attempts to decipher the password.

USER PASSWORDS

A third form of password protection is the USER password. If the bits for USER password have been set for a user, then you must have a PASSWORD DIRECTORY. The directory will contain a file for each user with ONLY the callsign as the file name. There are NO SSID's or extensions used. The sample directory and file name would look like this:

C:\MSYS\PASSWORD\N8UDK

At a minimum, N8UDK's user bits would be set to $400.

If a user has their flag bits set with the $400 bit on, when they connect they will get the Usage stats message line and must give the proper 4 characters just like a SYSOP would. The password file for a user must be in the PASSWORD directory and has a filename of the user's call (no SSID, no file extension).

A fourth password is the one used by the FTP function of TCP/IP. It is more fully described under the heading of MSYSPASS.DAT.

A fifth password file that is should be used is the MSYSPASS.IP. This file lists the call and password and callsign for Telnet users. If this file is not present when a station makes a Telnet connect to the bbs, rather than asking for login and password, it will simply ask for callsign which will be used without further verification.

A sixth password file is utilized for modem access. Discussion of the modem passwords used in the MSYSPASS.MOD file can be found in the documentation in the MODEM section.

TheNet & NetRom RS-232 Connections

RS-232 Connections of ports to TNCs running NetRom and TheNet should use > as the first character in the name given in the port statement in MSYS.OPT.

To connect the computer to a single TNC per computer port, a straight through cable is used (with a minimum of pins 2,3 and 7). At the TNC end, pins 10 and 23 must be jumpered. DO NOT CONNECT PINS 10 or 23 at the computer end! In a NetRom/TheNet equipped TNC, the connection between pins 10 and 23 switches the TNC interface from a normal ASCII terminal interface to one that sends NetRom Serial protocol.

In a multiple TNC connection using the standard diode matrix interconnect, the computer may be connected if you reverse pins 2&3 and pins 5&20 at the computer connection.

Using MSYS into a matrix with NetRom or TheNet equipped TNCs can have great advantages on a backbone system. The TNCs handle traffic on the backbone not going to/from the MSYS system without any load being placed on the MSYS computer. MSYS then gains the advantage of a fast reliable connection to the adjacent nodes.

Forwarding may be accomplished by connecting to the RS-232 connected TNC or matrix using normal script files. When using the NetRom Serial protocol, remember the ">" character in the port statement!

SMTP

The \+\filename include capability that is currently in the BBS S command is also usable in SMTP with the PRompt command. Both of these are available only from the local keyboard.

Typing other than Y N C A or RETURN in response to More? in the BBS is taken as a new command.

STACK Requirement

NOTE: NOT USED WITH EXPANDED OR EXTENDED MEMORY!

The stack space needed by the forward process has increased. The following line can be put in MSYS.OPT (This may not be necessary):

STACK FORWARD 8000

A smaller value (maybe 7000) might work. Try it if you want. If the system randomly crashed, make it 8000.

TERM

TERM is Sysop command which allows the Sysop to directly access the hardware connected to a port. It is extremely limited in what it can do. It is designed to allow you to interrogate and test port hardware.

The syntax for this command is:

TERM #

where "#" is the port number. TERM permits direct keyboard access to the port for issuing commands to a modem or perhaps a TNC. The screen will show what is received from the hardware. The keyboard input will be sent to the hardware.

The F9 key will take most TNCs out of the KISS mode so that you may communicate with the TNC command processor. DON'T FORGET TO PUT THE TNC BACK INTO THE KISS MODE BEFORE EXITING! In a similar fashion, when accessing the PACTOR/GTOR port, the same rule applies - leave the TNC in the proper mode (<PACTOR STANDBY>/<GTOR STANDBY >) before exiting. The term mode can also be used to access the modem, however, none of the MSYS capture or include features are available. To properly exit the dumb terminal mode, depress the F10 key.

"F9" can be used to take a TNC out of the KISS mode.

When used with a KISS TNC, not much be accomplished. The TNC can be taken out of KISS mode, adjusted and reset to KISS but automated events within the system may lead to undesired and unpredictable results. There is no recording, sending of files, etc. and there is no plan to add them.

"F10" will end the TERM session.

TESTING

You can connect two MSYS systems together through their serial ports by the use of a NULL MODEM.

You can also connect a loopback by connecting pins 2 and 3 on the same serial port. This will allow you to see what the system is doing without being on the air.

TITLES TRUNCATED

Titles of messages are truncated after 37 characters.

TYPE OF MESSAGE
(the "TR" field)

The algorithm uses 3 pieces of information to determine the appropriate TR to assign: the destination "callsign", the @"bbs", and the character after the S in the Send command.

The to ."callsign" is classified as one of the following:

- 0 - Looks like a real callsign
- 1 - A zipcode (5 digits)
- 2 - something else (like ALL, for example)

The @"bbs" is classified as one of the following:

- 0 - Looks like a real callsign
- 1 - none specified
- 2 - call of this bbs
- 3 - something else

The character after the S in the s command is classified as:

- 0 - B (as in SB for send bulletin)
- 1 - P (as in SP for send private)
- 2 - T (as in ST for send traffic)
- 3 - none (as in just plain S)
- 4 - something else (SW for send weather bulletin, for example)

The following message types are generated:

- BN For bulletins going to a specific bbs - these are forwarded to a single station that handles that bbs.
- PN For private messages
- TN For NTS traffic
- xN For Sx where x is an "something else"
- N For individual non-private messages.
- B$ For bulletins going with @route (like ARRL)
- x$ For bulletins sent with Sx (x is "something else")
- P$ For private bulletins going with @route

Note: [P]N below means that the type will be N is MAKEPrivate is off, PN if MAKEPivate is ON (default).

TR Values Assigned to Messages

For messages going to Callsigns:

Character after S ->	B	P	T	None	x (Other)
@BBS type:					
callsign	BN	PN	TN	[P]N	xN
no bbs given	B$	PN	TN	[P]N	x$
this bbs	B$	PN	TN	[P]N	x$
something else	B$	PN	TN	[P]N	x$

For messages going to ZIP Codes:

Character after S ->	B	P	T	None	x (Other)
@BBS type:					
callsign	BN	PN	TN	TN	xN
no bbs given	B$	P$	TN	TN	x$
this bbs	B$	P$	TN	TN	x$
something else	B$	P$	TN	TN	x$

For messages going to SOMETHING ELSE:

Character after S ->	B	P	T	None	x (Other)
@BBS type:					
callsign	BN	PN	TN	BN	xN
no bbs given	B$	P$	TN	B$	x$
this bbs	B$	BN	TN	B$	x$
something else	B$	P$	TN	B$	x$

Messages with second character of type N are forwarded to one place (and then killed if AUTOKill is ON, the N is changed to F otherwise). Messages with second character of type $ are flooded to all stations that get the specified @BBS. When sent to all such BBSs, the second character gets changed to #.

Messages that come in with an R: line that contains the call (actually HCAll) of this bbs are automatically held. The R: line scan is terminated by the first non-R: line found in the message.

Bulletins that arrive with a bid on the S command line are rejected if their BID already exists in the BID file. Bulletins are identified as those messages that arrived via the SB command or S non-call. All messages other than bulletins are always received (never rejected). If a message identifier ($string) exists on the S command line it is used. In all other cases, a message identifier is generated internally for all non-bulletins using the bbs call and message number from the last R: line scanned. If the message identifier is found in the BID file and the message was not held because of this systems R: line, the message is not saved (but it is acknowledged as being received entirely by sending the > prompt when the ^Z is received). The processing of internal message identifiers is not indicated in the system identification line [MSYS-XXXX-H$]. If you set MIDchar to something, then the character will appear between the H and the $. For example, if you set MIDchar to M, then the SID line will be [MSYS-XXXX-HM$]. This will be sent to all stations that connect to the BBS. Setting MIDchar to a non-null character will also cause the MID to be sent during forwarding if the system to which you are forwarding has MIDchar in its SID.

Here are some combinations of the parameters to do selected processing:

For no BID (mid) processing on non-bulletins, set
```
BIDall OFF
```

To emulate AA4RE MID processing (I think), set
```
BIDallON
MIDchar M
```

To use a MID if provided, or generate one otherwise, set
```
BIDall ON
ACceptmid ON
MIDchar M
```

Currently there is still a lot of discussion going on over MIDS. Hopefully I have give enough parameters so that MSYS can be used with whatever standard wins out.

USER BIT SETTINGS:

$01 - Expert user
$02 - BBS
$04 - Bad User Bit (Disconnect immediately)
$08 - Limited BBS Permits only limited use of the bbs
$10 - Remote SYSOP status
$20 - Need to send WP Update
$40 - Multiline packets (XF)
$80 - Prevents use of the S command. Users so marked can only read messages.
$100 - Lock user
$200 - Authorizes use of the Upload command in the bbs.
$400 - Must have user password
$800 - YAPP Upload authorized
$1000 - Auto Reply turned on
$2000 - No Automatic LC in BBS
$4000 - No "You have unread mail" message
$8000 - New User flag

NOTE: These settings may be added together.

CAUTION! They ADD in HEXADECIMAL! 800 + 200 = A00

WP SERVER:

Generation of messages to the white pages master server is included in MSYS. This sends the information users give in the N, NH, NZ and NQ commands to the WP server when it is changed (or initially given) as well as periodically. Note that MSYS does not respond to messages to WP but merely passes them on without modification. WP messages are NOT generated for users with any of the following characteristics: locked, homebbs of ?, bbs system, and/or baduser. The hierarchical address is currently defaulted to N6IYA.#NOCAL.CA.USA.NOAM, but may be changed by the MASterserver command.

KANTRONICS 3.0 EPROM users:

I am told you will have to make a TNC setup file and include the following:

```
INTFACE KISS
RESET
```

You may also need to include a line with one or more control-c characters before these two lines in your file.

Later versions of the KPC series seem to work OK with no MSYSTNC.# file.

FREQUENT PROBLEMS AND NOTES FROM THE AUTHOR

IMPORTANT: You must use the new version of MUTIL with this version of MSYS (i.e., run setup, function 9). It is probably also a good idea to delete MSYS.HRD when going to a different version of MSYS or changing the port statements in MSYS.OPT! You MUST also use the new MSYSMSGS.DAT file in the \HELP directory

If you have problems getting MSYS going, here are some things to try that have helped many who have called:

1. Make sure you have FILES=40 and BUFFERS=20 in CONFIG.SYS
2. Try adding STACKS=0,0 to CONFIG.SYS
3. Get rid of anything resident (check AUTOEXEC.BAT and CONFIG.SYS)
4. Run MUTIL 4 (do this EVERY time you start MSYS)
5. Simplify MSYS.OPT as much as possible; remove all PORT statements
6. Delete your MSYS.DO file (after making a backup!)

These items should get you to the point where MSYS will work. From here try putting things back and see what causes failure.

OVERRUNS, FRAMING ERRORS

On occasion, it may be useful to do an ST S command to see the status of your ports. If you find that you have "OVERRUNS", you should probably make some adjustments in the MSYS.OPT file for speeds. An overrun is a condition that occurs when the TNC delivers a character to the serial port and the computer does not process the character in time before another character is presented. If the second character is presented before the first has been taken out, the UART will signal MSYS that there has been an overrun. One possible solution is to use a buffered UART such as the 16550, which MSYS can properly utilize. A second choice is to adjust the TNC to computer bit rate. The "normal" thinking is to run the port as fast as possible, but it may be useful to actually slow the TNC port to something over the radio speed but less than port maximum. Empirical tests have shown that a TNC/computer speed of 4800 seems to work well on a 386-25 machine using 8250 (non-buffered) UARTs. In any event, you may need to adjust the TNC speed of the various ports until you minimize any overruns. Modem speeds do need to be at least as fast as the highest speed on the modem, and preferably twice as fast. To change the speed of a port, remember it is necessary to change the value in MSYS.OPT, then run MUTIL 9 before the changes will take effect.

ABRUPT STOPS IN FORWARDING

It has been observed that ANY blank lines in the MSYSFWD.DEF, or any of the .FWD or .REV files included in the MSYS.FWD file will cause problems. Make sure that ALL if the included files do NOT contain blank lines at the ends of the file. Duplicate message numbers (caused during "crashes") will also cause a halt in the forwarding process. See "Analizing" below.

MESSAGES DON'T FORWARD

ALL forwarding is done on the @BBS. In the hierarchical routing process, it is possible that a bbs will contain 2 or more of the criteria for forwarding. MSYS will only forward based on the left-most item matched in the @BBS. For example, let us assume that we forward all messages @ .#NEOH to WA8BXN. We also forward messages @ KB8AST to NO8M. If a message @ KB8AST.#NEOH.OH comes in to be forwarded, MSYS will investigate the hierarchical routing and determine that all messages @ KB8AST go to NO8M. If, for some reason, NO8M is not able to take the message, it will NOT be forwarded to WA8BXN as a "second choice". It is therefore imperative that you not "over-specify" when setting up your forward files with the hierarchical routing. One major advantage of hierarchical routing is that you need not list EVERY bbs in a state or an area. You only need to specify the hierarchical component.

Make sure you execute the MERge command after every change in the \FWD files or in the MSYSFWD.DEF file. Without doing this the system will not forward properly.

NOTE: You must have the file MSYSBBSB.DAT to be able to forward bulletins.

The system forwards messages to improper locations.

Check to make sure that your MSYSFWD.DEF or MSYS.FWD files do not have improperly placed "*" characters or have imbedded empty lines at the end of the sections. These "empty lines" are often found at the end of the xxx.FWD or xxx.REV files.

The system appears to work but:
Nothing is on the screen!

Make sure you set the display type correctly in the MSYS.OPT file. The command is WINDOW DISPLAY COLOR for color monitors or WINDOW DISPLAY MONOCHROME for monochrome monitors.

THE SYSTEM COMES UP AND APPEARS TO HANG

This may be caused by a non-existent com port. Check the lines in your MSYS.OPT to make sure the port addresses are correct. Also, Rerun MUTIL 4 to insure the mail files are properly indexed.

SYSTEM REBOOTS AFTER FORWARDING

This may be caused by having too many entries in your forwarding file. Remember you may only have up to 400 different bbs names in your file!

SYSTEM APPEARS TO BE STUCK WHILE ANALIZING FORWARD FILE

This condition may occur after a "crash" even though MUTIL 4 has been run if there are duplicate message numbers in the list. The message number will be shown in Window 6 on the F1 screen. To resolve the problem, kill the duplicate message with the duplicate number, stop MSYS, rerun MUTIL 4, and restart MSYS. At that point, the killed message may be added to the list with the \RESTore command from the bbs.

INPUT BUFFER SPACE LOW

This message may occur while watching the F2 screen. It refers to how much of the 4K input buffer is full. It is not expanded beyond that, so the amount of memory that the MEM command returns has no significance. INPUT BUFFER SPACE LOW most often indicates a shortage of cpu speed or something unusual that may be using a lot of cpu cycles.

PERFORMANCE HINTS

Don't let too many messages pile up in MAIL.BAK <DIR> before you delete them (after archiving if you wish) so the hard disk doesn't fill up.

From time to time do a BID # command and see how many bids are active. If its over 3 or 4 thousand its probably time to run MUTIL function 21 to get rid of the older bids. You will get a warning message BID LIST NEARLY FULL when the bid list is about 2/3 full.

DELETE OLD LOG FILES.

It is probably wise to put a statement: ??D10000 log in your MSYSTODO.DAT file to keep the log file from expanding to extremes. It is not uncommon to have a log file exceed 1 megabyte in a week's time! The "log" statement will convert the MSYS.LOG file to a yymmdd.log, which then can be archived off line.

While you are creating the MSYSTODO.DAT file, it is also not a bad idea to add a: ????0001 NODe SAve just to keep your current node list in case of a reboot.

Periodically (weekly?) you should also do the following:

1. Delete old BIDs using MUTIL 21
2. Delete old backup messages. (after running Mutil 10)
3. Delete all files in the /mail.bin directory (while MSYS is NOT running!)

Here are some suggestions from users of MSYS that they claim work. Use at your own risk!

To cure the problem of certain TNCs that don't transmit properly on HF use radiospeed of 1200 in place of 300 in MSYS.OPT. I can't imagine how this could help, but someone says it did for them (they weren't using MFJ TNCs which seem to have this kind of problem). One thing you might want to try if you are having problems with a TNC on HF is to get a copy of KA9Q's net.exe program and see if you can use its AX.25 connect command with your TNC on HF. If that fails as well the problem definitely is in the TNC.

MFJ KISS MODE

To put a MFJ TNC in kiss mode, try this in your MSYSTNC.# file

```
ax off
parity 0
awlen 8
conmode trans
kiss on
restart
```

If you are just bringing up MSYS for the first time and experience crashes when mail is forwarded to you, or the size of BIDLIST.DAT is 0 bytes, do the following to solve the problem: Shut down MSYS (quit), delete BIDLIST.DAT, start MSYS, shut it down (the DOS DIR command should now show a size of 130000 for BIDLIST.DAT). You should now be ready for normal operation.

TIME

If you find that the displayed time is different from what you have set the time to be in DOS, put the following in AUTOEXEC.BAT and reboot: SET TZ=EST5 (Obviously, set it to YOUR time zone!) One more thing - the time can (and does) often get hosed up around the first of April and the end of October because of certain software language routines that attempt to account for the change in Daylight Savings time. Not to worry (too much) because it will eventually make itself right (with maybe a little help from you!)

You should also be aware that you may set your computer to local time (with the TS hhmmss command) and set the time that MSYS uses to GMT by use of the GMtime xx command where xx is your offset from GMT in hours. As an alternative, you can set your computer to local time and utilize the RLinetime command to insert the "proper" GMT time (with the "Z") in the R: line.

ADDITIONAL NOTES

Paccom ROM version 1.1.5 may work better than 1.1.662

If you are using DOS 3.3, you may need to put STACKS=0,0 in your config.sys file.

Home BBS replacement occurs only if @BBS is blank or the call of your bbs.

Note: If you run into problems with newer versions of QEMM and MSYS, try turning off the stealth option in QEMM.

OS2 REPORTED PROBLEMS

Reference a message from N8LJL @ N8ITP.OH.USA.NA

To All users of OS2 2.1 GA. A problem has been found when using Com 3 or Com 4 on IRQ 2 on ISA System Boards. When a DOS Session is opened with access to a port using IRQ 2, the program may come up initially and work fine, but when you switch back to the DeskTop and return to the DOS Session, the screen data will not update. This is most common on ProComm Version 2.01. It appears that the interrupt control portion of the program is losing it brains as to where IRQ 2 is assigned. IBM is currently working on the problem. Hopefully a resolution will not be too far down the road.

This problem has existed since the advent of OS2 2.0 GA. It will only affect those users who are trying to run more than 2 com ports and require access to IRQ 2.

There is indeed an interrupt conflict, but all is not lost. Refer to pages 376-380 in the OS2 manual, with particular note to page 378 and the "DOS settings" on 379. The book missed a point that will be important. In config.sys and an explanation of what they do:

```
DEVICE=C:\OS2\MOUSE.SYS SERIAL=COM1
DEVICE=C:\OS2\COM.SYS (3,3e8,5)
DEVICE=C:\OS2\MDOS\VCOM.SYS  (3,3e8,5)
```

The second line sets up COM3, address 3e8, to IRQ5 for OS/2 sessions. The THIRD line does the same for DOS programs running under OS/2.

I recommend using IRQ5 for COM3 unless you have an LPT2 printer attached.

Here are the DOS settings K0TS is currently using in OS/2 with MSYS.

```
COMM_DIRECT_ACCESS      ON
COMM_HOLD               ON
DOS_FILES               40
DOS_HIGH                ON
DOS_UMB                 ON
EMS_HIGH_OS_MAP         96
EMS_LOW_OS_MAP          576
EMS_MEMORY_LIMIT        As high as possible
IDLE_SENSITIVITY        93
VIDEO_MODE_RESTRICTION  CGA
XMS_MEMORY_LIMIT        As high as possible
XMS_MEMORY_HMA          63
```

> IF MSYS DOESN'T SEEM TO BE WORKING PROPERLY, REMOVE ALL (EVERY LAST ONE!) RESIDENT PROGRAMS INCLUDING YOUR ENTIRE CONFIG.SYS (keeping files=40 and buffers=20) AND AUTOEXEC.BAT FILES AND SEE IF IT WORKS THAT WAY. IF IT DOES, PUT THINGS BACK IN ONE AT A TIME TO SEE WHERE THE PROBLEM IS!

MSYS Utility Support Programs
(c) Copyright 1994 by HUB COMPUTERS, INC.
MUTIL.EXE

When you run MUTIL you will get a screen that is similar to:

```
============================================================

MSYS Utilities 1.18 by WA8BXN (C) 1994 Hub Computers, Inc.

1    Change maximum number of messages
*    Restore messages from backup directory MAIL.BAK        ***
3    Print remote sysop phrase number table
4    Check Mail index file integrity
5    Test serial ports
6    Print forwarding tree
7    Convert JNOS domain.txt to MSYSHOST format
8    Run log file analyzer
9    Run SETUP program
10   Scan message headers to create/add to BBSLIST.DAT
11   Deletes old entries from BBSLIST.DAT and BBSTONTS.DAT  files.
12   Fix mail index file (can be hazardous to use!)
13   Process BBSLIST.DAT into BBSTONTS.DAT file
14   Scan BBSTONTS.DAT to build BBSTONTS.BIN (indexed)
15   Build list of archived messages
16   Printer status
17   MERge BBSLIST.DAT files
18   Check MSYSNODE.DAT integrity
19   Convert user files 1.05 & before <===> 1.06 and later
20   Sort calls in the BBSLIST.DAT file
21   Delete Old BIDS (and MIDS)
22   Specify file name to use in place of printer
99   Quit

Enter number of selection: _
```

*** This function has been replaced by the /RESTore command in the BBS

MUTIL Function 1 - Change max number of messages

Input file(s): MSYS.MSG
Output files(s): MSYS.MSG

The maximum number of messages that can be stored in MSYS is determined by the MSYS.MSG mail index file. When MSYS starts running, it checks to see if there is an existing MSYS.MSG file. If there is one, it is used (and contains the header lines of the messages that existed when MSYS was last stopped). If there is no existing MSYS.MSG file, then an empty one is created, with a capacity of 100 messages. This utility function can be used to change the size of an existing MSYS.MSG file. You can increase or decrease the size of the file. You can't make it smaller than the number of active messages. The maximum number of active messages has been increased to 2000.

NOTES: There is a cost in memory requirements for each message possible. Each message slot costs 138 bytes of memory; 100 messages (default maximum) thus takes 13800 bytes. 500 messages would take 69,000 bytes and 2000 messages will take approximately 270,000 bytes (even if there weren't that many messages active).

VALUES I USE: I run with a 500 message maximum size. Normally I have around 400 active messages.

DIALOG WITH FUNCTION: You are first asked for the name of the MSYS.MSG file to modify:

Enter path to MSYS Mail Index file or just press return to use MSYS.MSG: _

Normally you will want to just press the return key. If the current directory from which you are running MUTIL does not contain the MSYS.MSG file that you want to modify, you can specify a different path (or even a different file name if you have renamed MSYS.MSG for some reason).

This function will next copy the specified MSYS.MSG file to a file called MSYS$$$.MSG which can be used to restore MSYS.MSG if for some reason this function fails.

Next you are asked if you have run the Check Mail function (function 4). You should run Check Mail just before this function since attempting to change the maximum number of message slots can produce very interesting (and usually bad) results if the original MSYS.MSG file is corrupt. If you respond N then this function terminates and you can then do the Check Mail function. If you have just run Check Mail, then press Y to continue. Note that you do NOT press return after either of these letters!

If you responded Y, then the function continues, telling the current maximum number of messages, the current number of active messages, and some other information. You are then asked for the new maximum number of messages you want. This number must be greater or equal to the number of active messages. It must also be less than or equal to 2000.

If the value you ask for is acceptable, the file will be changed to accommodate the new number.

Next you will be asked to press a key to continue (try the return key). This pause is put in the function so that you can see any additional messages that were produced before the screen is cleared.

MUTIL Function 2 - Restore Messages
MOVED INTO MSYS PROGRAM!

This function has been moved into the MSYS program with the SYSOP command of /REST [msg#] which MUST be done while connected to the bbs.

MUTIL Function 3 - Print Remote Sysop Phrase

INPUT FILE(S): MSYS.RMT
OUTPUT FILE(S): Hardcopy printer output (optional)

This function prints the contents of the MSYS.RMT file (used for remote sysop verification). The output of this function can be just the screen or also to the printer. The output consists of printing the content of remote sysop phrase followed by a table that gives the character at each position in the phrase.

DIALOG WITH FUNCTION:
You are asked if you want hard copy output (on the printer). Respond Y or N as desired.

SAMPLE OUTPUT:

```
Sample phrase.
   0   S
   1   a
   2   m
   3   p
   4   l
   5   e
   6
   7   p
   8   h
   9   r
   10  a
   11  s
   12  e
   13  .
```

USAGE EXAMPLE:

If you did the \Sysop command as a remote user to get sysop privileges and got the following reply:
Usage stats: Users=7 Msgs rcvd=3 Msgs sent=6 Max=12 you would type the following response:
pp e

MUTIL Function 4 - Check Mail Index File Integrity (CHKMAIL)

```
INPUT FILE(S):  MSYS.MSG
                MAIL\MSG*.DAT
OUTPUT FILE(S): MSYS.MSG
```

This function checks the integrity of the Mail index file MSYS.MSG. It checks for such things as message slots on both the free and active lists, loops in either of these lists and active message numbers that do not have text files in the MAIL directory (and vice-versa). It attempts to fix any errors if it can (and its generally pretty good at doing it).

NOTE: It will not delete any message text files in the MAIL directory if they don't have active headers. They will be reported but you will have to delete them yourself if you really don't want them. You probably won't really want them, but you decide!

NOTE: This function may be run in a batch file through MUTIL (without having to give its function number interactively) by running:
MUTIL 4

SAMPLE NORMAL OUTPUT:

```
CHKMAIL - An MSYS Utility to verify mail index integrity
MSYS Mail Index Checker

head=0 free=1 next=2
Reading Index file:
There are 100 total message headers

Pass 1:
No cycles detected
No headers on both lists
All headers accounted for
No message files without headers
All active headers have files
Press a key to continue
```

COMMENTS: I would strongly suggest you run this function EVERY time MSYS crashes for any reason (it should not crash often, if at all). To be on the safe side, many MSYS SYSOPs always run it when they start MSYS.

MUTIL Function 5 - Test serial ports

INPUT FILE(S): None
OUTPUT FILE(S): None

This function provides a very basic test of serial interfaces. It is designed mostly to see if you got the addresses/int numbers right for your machine. It is not intended to be a brutal test of their operation!

DIALOG WITH FUNCTION:
After a table is displayed that lists the normal addresses for serial cards and corresponding interrupts, you will be asked if your machine has an 8086/8088 CPU chip in it. If its a basic PC/XT compatible system, respond Y. If you have an AT (80286) or higher level machine, respond with N. If you have an 8088/8086 based system (PC class) then it will look for ints from 2,3,4, and 7. If you say not 8088/8086 (AT class and above) it will look for ints from 3,4,5, and 7. Depending on your response, certain interrupt handlers will be installed. You will then be asked for the address of the serial card you wish to check. The answer you give must be in HEXADECIMAL. A valid response might be 3F8, for example. A simple check is made to see if there might be a serial port at the address you specify. It is easily fooled if you specify the address of some other kind of I/O interface.

If there seems to be a device at the address you gave, a quick check will be made to see if it looks like an 8250 UART chip is used in your serial interface. Internal loop back tests cannot be done with the newer UART chips. If the test fails when you select internal loopback, try the external loopback option. This requires a connector that will connect pins 2 & 3 together in place of the TNC you would normally connect. External loopback can be used to test all types of UART chips.

Next you will next be asked for a baud rate to test at. Here you give a decimal number as an answer. It probably doesn't make much difference what you give . (Don't try 0!). I usually test at 1200 baud.

Now its your turn to type characters. As indicated, press Esc to terminate the test. Normal response should be something like:

Had IRQ 4 1 time(s), A received

If you press the letter 'A' key. If interrupts aren't working you would see:

Received A without interrupt.

NOTE: This is not a performance test, just an address test.

MUTIL Function 6 - Print forwarding tree

INPUT FILE(S): MSYS.FWD
OUTPUT FILE(S): Printer hard copy (optional)

This function displays your forwarding file in two ways to tell you what callsigns you forward to each bbs you forward to and also to which bbs you forward each callsign.

DIALOG WITH FUNCTION:
You are asked if you want hard copy output (on the printer). Respond Y or N as desired.

The first part of the output consists of lines like

WN7C ---> KA8Z WB8BII

which means that messages with @WN7C are sent to either KA8Z or WB8BII(which ever one you can connect to first).

The second part of the output consists of lines like

KA3NVP <--- KA3NVP NTSNJ NTSNY NTSPA

which means that you forward messages with KA3NVP NTSNJ NTSNY and NTSPA in some part of the @BBS field to KA3NVP.

NOTE: To pause the display, you must use ^NumLock (^S won't work). To resume scrolling, press any other key.

MUTIL Function 7 - Convert JNOS domain.txt file to MSYS Format

INPUT FILE: domain.txt
OUTPUT FILE: MSYSHOST.NEW

DIALOG WITH FUNCTION:
This utility takes the commonly available JNOS domain.txt file and converts it into a format that can be used by MSYS. The file will still need to be edited to provide aliases. You will need to change the name to MSYSHOST.NET (from MSYSHOST.NEW), but the bulk of the typing will be handled automatically.

MUTIL Function 8 - Run Log File Analyzer

```
INPUT FILE(S):  MSYS.LOG
         MSYS.MSG
OUTPUT FILE(S): Printer hard copy (optional)
```

This program produces a listing of BBS usage.

DIALOG WITH FUNCTION:
You are asked if you want hard copy output (on the printer). Respond Y or N as desired.

SAMPLE OUTPUT:

```
============================================================
From Wed Mar 01 15:23:48 1989 to Mon May 229 00:57:03 1989
Station  Connects     Sends   Reads     Kills     Forwards
HCLean       0          0       0        27         0
N8UDK        3          1       1         0        12
NT8V         0          1       1         0         0
WA8BXN       4          0       0         1         9
WB8ZZS     168         82      31        24         0

LAST READ       MSG# TO   @BBS  Count  Title
May 18            1 TEST         1    test
Mar 23            7 TEST         2    test
May 28            2 TEST         3    test
Mar 02         8073 TEST         7    no title

MSGINs 84  MSGKILLs 97  MSGFWDs 21  MSGREADs 33 RESTARTs 331  QUITs
311  CONNECTS 175
The following messages were received but not killed:
   1   2   3   20   21   8073
(These should be active messages on your BBS!)
============================================================
```

NOTES: This data is taken from my test system which uses WB8ZZS as its call (Thanks to my wife Ruth!) Many strange things happen on this system as I am testing new software changes.

The first line of the report gives the period of time covered by the contents of the MSYS.LOG file that was read.

The next section of the report lists stations that were active on the BBS and what they did. The headings mean the following:

- Station - Callsign of the station
- Connects- Number of times the station connected (or the SYSOP used the BB command).
- Sends - Number of messages sent by this station
- Reads - Number of messages read by this station
- Kills - Number of messages killed by this station
- Forwards- Number of messages forwarded to this station

NOTE: The callsign HCLean indicates activity by the House Cleaning routine.

The next section lists messages that were read during the period, sorted in increasing order of number of reads . The meaning of the headings is:

```
   LAST READ - Date the message was last read
   MSG#      - The number of the message on the bbs
   TO        - The "TO" station callsign of the message
   @BBS      - The "@" field of the message.
   Count     - The number of times the message was read
   Title     - The title of the message
```

NOTE: Private messages are not shown, only bulletins. This section is useful to see what interest there is in the various types of messages.

The next section provides overall counts of various activities:

MSGINs	The number of messages that were sent to the BBS
MSGKILLs	The number of messages that were killed
MSGFWDS	The number of messages forwarded by the BBS to other systems. Note: If a bulletin is forwarded to 4 different systems, it counts as 4 forwards
MSGREADs	Total number of Read commands used on BBS
RESTARTs	Number of times MSYS was started
QUITs	Number of times MSYS was gracefully shut down using ^F4 or Quit command. Restarts - quits is the number of crashes for whatever reason.
CONNECTS	Number of user connects to BBS + SYSOP BB cmds

The final section of the report gives a list of message numbers that were assigned to messages sent to the system but not killed. Normally these message numbers should be active messages on the BBS. If they aren't, they may be messages to restore since they got lost somehow.

MUTIL Function 9 - Run Setup Program

INPUT FILE(S): MSYS.OPT
OUTPUT FILE(S): MSYS.DEF

This function sets default initialization parameters. IT MUST BE RUN BEFORE YOU CAN RUN MSYS THE FIRST TIME and any time you change MSYS.OPT. It is documented elsewhere in this document.

DIALOG WITH FUNCTION:
You are asked if you want hard copy output (on the printer). Respond Y or N as desired.

MUTIL Function 10 - Scan Message Headers
to create/add to BBSLIST.DAT

INPUT FILES: MAIL.BAK\MSG*.DAT (or specified directory) BBSLIST.DAT (if it exists)
OUTPUT FILE: BBSLIST.DAT

This program scans backup mail message files that are (or were originally found) in the MAIL.BAK directory to determine what BBS stations have been "seen" (using the R: lines added by each BBS as it forwards a message).

DIALOG WITH FUNCTION:

You will be asked to give the name of the directory that contains the messages to be scanned or to press return to use the MAIL.BAK directory directly.

NOTE: This function may be run from a batch file by using either:

```
MUTIL 10
```

or:

```
MUTIL 10 dirname
```

In the first form (MUTIL 10) it will be assumed that the input files are in the MAIL.BAK directory. In the second form (MUTIL 10 dirname) the files are assumed to be in the given directory name.

OUTPUT FILE FORMAT:
The first line of BBSLIST.DAT gives the date this function was run to create or add to the file.

The remaining lines in the file consist of the following fields:
callsign of bbs
location of bbs
postal code of bbs
number of times this call has been seen

Note: If you choose to edit this file, be sure to keep things in the same columns. Also note that the callsigns are lined up using the digit of the callsign. The list is sorted by digit in callsign, letters after the digit, and then letters before the digit (pretty much callbook order). Calls that do not begin with A, K, N, or W are grouped together at the end of the list, with an * in the first position in the record (non-US calls).

MUTIL Function 11 - Deletes Old Entries From BBSLIST.DAT and BBSTONTS.DAT

This utility function will prompt the user to determine what the oldest record date you wish to keep in a yymmdd sequence. It will then edit the files to delete BBS records older than the specified date. It might be a good idea to then run MUTIL Function 10 to add any new BBS's to the list before running MUTIL Function 13.

MUTIL Function 12 - Fix Mail Index File

***** WARNING ***** This function can DESTROY the mail index file

```
INPUT FILE(S): MSYS.MSG
               MAIL\MSG*.DAT
OUTPUT FILE:   MSYS.MSG
```

This function allows manual inspection, checking and modification of the mail index file. It is a utility I wrote to see what destroyed a mail file and perhaps fix it. There is one function that may be useful to most SYSOPS, that allows you to specify the next message number to be assigned to incoming messages. You may want to do this if you were running some other bbs and want to continue using message numbers from where you left off. Personally I would suggest that you don't do this but rather start with the default value 1. But its up to you!

NOTE: Make a copy of your existing MSYS.MSG file before using this function!

NOTE: Most of the checking and repair functions have been automated in the CHKMAIL function (MUTIL Function 4).

DIALOG WITH FUNCTION: You are first asked for the name of the MSYS.MSG file to modify:

```
Enter MSYS message header file name or simply press return for
MSYS.MSG: _
```

Normally you will want to just press the return key. If the current directory from which you are running MUTIL does not contain the MSYS.MSG file that you want to modify, you can specify a different path (or even a different file name if you have renamed MSYS.MSG for some reason).

To change the number that will be assigned to the next message, use the following procedure:
Select F8 (Manual changes)
Select F5 (Change Next)
Enter new value for next message number to be used.
Select F10 (Return to previous menu)
Select F9 (Save changes and quit)

RENUMBERING MESSAGES

All active messages may be renumbered using MUTIL function 12. You should renumber your messages when you reach about message number 64000. As message numbers increase above this they will wrap around back to 0 if you don't use this function. Some of the L command options will not work properly if newer messages have numbers smaller than older ones. To renumber the messages, go through the following steps:

1 - Start the MUTIL program
2 - Type 12 and return
3 - Press return to accept use of MSYS.MSG
4 - Press F8 for manual changes
5 - Press F6 for Renumber messages
6 - Type new starting message number (1 is good) then press return
7 - Press F10 to return to previous menu
8 - Press F9 to save changes
9 - Press return to go back to main menu
10 - Type 99 and return to exit

It should be noted that any files in the mail.bin directory should also be deleted when renumbering messages.

MUTIL Function 13 - BBSLIST.DAT to BBSTONTS.DAT

```
INPUT FILES:  BBSLIST.DAT
              ZIPCODES.DAT
              STATES.DAT
              ROUTES.DAT
OUTPUT FILE:  BBSTONTS.DAT
```

This function builds a list of BBS calls (and optionally zip codes) and the corresponding xx designator to which they should be forwarded.

If you select processing US BBS calls by state the data found in BBSLIST.DAT is scanned for state and zip code. The state is identified as being something after a comma in the location field that matches the list of state two letter codes found in the STATES.DAT file. If only one state is found and it corresponds to the zip code found, then the BBS call is automatically assigned to xx where xx is the state abbreviation. If either of these pieces of information is missing, or they do not match the test fails and you are asked to decide what state the bbs belongs in.

DIALOG WITH FUNCTION:

You will be asked if you want to include state routings for ZIPCODES. You should reply Y the first time you are creating the BBSTONTS.DAT file. Otherwise reply N. If you delete BBSTONTS.DAT for some reason, you should reply Y the first time you create a new one.

If the state of the BBS cannot be automatically determined, you will get a screen that looks like this:

```
============================================================
      WB8ZZS   KIRTLAND OH        44094   (4)

State/ZIP check failed - Zipstate OH
Enter two letter state code then <Return key> or press <Escape key>
then <Return key> to ignore or just press <Return key> to use OH

============================================================
```

The first line of this screen is the line from BBSLIST.DAT that is being considered. The next line gives the state that corresponds to the zip code (if available). The third line gives the guess that can be accepted if you just press return. If the guess is good, hit return. If you want to use a different state, type the two letter code for that state and then press return. If you want to omit this call from the output file, press some non-letter key before pressing return. Space bar, return works well.

In this example, the state could not be identified because there is no comma between the city and the state. (It would be nice if everyone used standardized R: lines!) The guess of OH was made from the zip code alone.

When the US calls have been processed, you will be asked if you want to process non-US calls. If you have earlier chosen not to process US calls by states/zipcodes, the US calls will be processed with all the other calls if you choose to process non-US calls. This feature is useful for non-US hams that handle all US messages as just going to the US.

If you choose to process the calls at this point, you will be presented with screens that list an entire entry from the BBSLIST.DAT file. Following this line will be the callsign of the BBS identified from the line. Using the call prefixes found in the ROUTES.DAT file, a guess will be made as to what routing should be used for the given call. You may then choose to ignore the call, to use the routing obtained from the call prefix (if any) or to type the complete route you desire.

A sample screen might look like this:

```
============================================================
   *  WB8ZZS   KIRTLAND OH        44094   (4)
Callsign:    WB8ZZS
Possible route: USA.NA

Press <Escape key> <Return key> to ignore
Type desired routing then press <Return key> or
   just press <Return key> to use USA.NA

============================================================
```

BBSTONTS.DAT FILE FORMAT:
This is an ASCII file containing in bytes 1 to 9 a BBS call or zipcode (left justified, possibly with wildcard * indicator), and the corresponding designator to which messages for that BBS call or zip code should be forwarded beginning in position 10 . Example lines:

```
89*     NV
WB8ZZS  OH
VK1XX   AUS.AS
```

NOTE: Bulletin routing designator (like ARRL) should NOT appear in this file.

NOTE: The file is sorted in ascending order using the entire record as key.

NOTE: There should be no duplicates of calls/zip codes found in the first 9 bytes of the records.

MUTIL Function 14 - BBSTONTS.DAT to BBSTONTS.BIN

INPUT FILE(S): BBSTONTS.DAT
OUTPUT FILE(S): BBSTONTS.BIN

This function builds an index of the calls in the BBSTONTS.DAT file, placing the index at the beginning of the BBSTONTS.BIN file. After the index, the contents of BBSTONTS.DAT is copied to BBSTONTS.BIN.

DIALOG WITH FUNCTION:

This function does not require any keyboard input.

NOTE: The format of the resulting file is such that no attempt should be made to edit it. With care, BBSTONTS.DAT may be edited and the result then processed by this function.

NOTE: The file BBSTONTS.BIN is used by MSYS. The files BBSTONTS.DAT and BBSLIST.DAT are NOT used directly used by MSYS and may be stored on a diskette, if desired, when MSYS is running. I just keep them in the normal MSYS directory.

MUTIL Function 15 - Build List of Archived Messages

This function will look at messages found in a given directory (in MAIL.BAK format) and construct a file of the message headers. The result is a file of all your messages like doing a L command on the BBS for every message that has passed through your system. You can use the grep function on it to find messages with some particular string of characters in their header line. I use it to find a message when I think to myself "I remember there was a message about ... one time". Then knowing the message number you can go back to your archive diskettes and extract the complete text.

MUTIL Function 16 - Printer Status

This function provides a very simple printer test to see if its connected and can be initialized.

MUTIL Function 17 - MERge BBSLIST.DAT Files

This function takes as its input your current BBSLIST.DAT file (which it first renames to BBSLIST.BAK) and another file you specify in the same format as BBSLIST.DAT) and merges the two files into one called BBSLIST.DAT. I use this function to merge the BBSLIST.DAT file that some of the users of MSYS have sent me. This allows me to include in my file some bbs calls that normally would not have been seen in messages passing through my system.

MUTIL Function 18 Check MSYSNODE.DAT Integrity

This function performs the following functions:

MSYSNODE.DAT viewing and repair program

```
 0     Printer echo toggle (now OFF)
 1     List active neighbors
 2     List all neighbor slots
 3     List active known destinations
 4     List all known destination slots
 5     Data consistency check
 6     View given neighbor or destination
99     Exit
```

Any output from the program will be directed to the screen unless the printer has been toggled on.

MUTIL Function 19 - Convert User Files 1.05 & Before To 1.06 and later

Version 1.05 and earlier used a different format for storing user files. This utility probably will be of little use to most current users.

MUTIL Function 20 - Sort BBSLIST.DAT File

This utility will sort the callsigns in the BBSLIST.DAT file. It will sort the bbs callsigns based on number, suffix after the number, prefix to the number and generate a new file called BBSLIST.NEW At the conclusion of this operation, it will suggest that you rename the files in the proper manner. This utility has been revised in MUTIL 1.16 to operate faster and to permit more BBS calls.

MUTIL Function 21 - Delete Old BIDs (and MIDs)

This function deletes old BIDs/MIDs from the file BIDLIST.DAT. It asks for the number of days worth of bids that should be kept. Run it periodically (maybe once a week when you do your hard disk backups). The bid file can hold up to 6500 entries The more it has, the slower it works. The maximum number of BIDs in the BID file is settable.

MUTIL Function 22 - Specify Printer File Name

The file you specify will be opened and will receive any output that would normally go to the printer. Any existing data in the given file will be lost. If you want to abort this function, simply press return when asked for the file name.

SAMPLES OF MSYS FILES

MSYS.OPT:

```
port 0 at $3E0 int 7 speed 4800 name 145.01 RADIOSPEED 1200 TNCTYPE 1
port 1 at $2E8 int 7 speed 9600 name 223.68 RADIOSPEED 1200 TNCTYPE 1
port 2 at $3E8 int 7 speed 4800 name 145.05 RADIOSPEED 1200 TNCTYPE 1
port 3 at $2F0 int 7 speed 9600 name >AKRNODE RADIOSPEED 9600
port 4 at $2F8 int 3 speed 19200 name MODEM RADIOSPEED 1200 TNCTYPE 2
port 5 at $3F8 int 4 speed 9600 name PACTOR_80M
numchans 15
bbs alias akrbbs
bbs call wb8bii
id call k8eiw
id every 30
forward call wb8bii-1
answer call k8eiw-15
kanode call wb8bii-3
sysop name Don
sysop qth Cuyahoga Falls
sysop zip 44223
process id window 2
monitor outgoing on
window display color
window 0 size 1 1 80 13
Window 6 size 1 23 48 25
make
```

MSYS.DO:

```
*----------------CALLSIGN STUFF-------------------------------
HCA WB8BII.#NEOH.OH.USA.NOAM
HOST WB8BII
NRI AKRON
NRC WB8BII-1
MYT WB8BII-1
myip 44 70 242 6
mya akrbbs
*-----------------TNC PARAMETERS-----------------------------
monsf on
check 0500
*------------Turn on CTS for Modem and PACTOR ports
cts 4 1
cts 5 1
*------------Set FRACK for ALL ports
frack 12
*------------Reset FRACK for high-speed matrix port
frack 3 5
*------------Set Response time, slot time, txtail and txdelay
resp 1
slot 10
tac 2
txt 5
txd 0 35
txd 1 30
txd 2 35
txd 3 5
*------------Set maximum packet length, and maxframes per port
paclen 236
maxfr 0 4
maxfr 1 7
maxfr 2 4
maxfr 3 7
*------------Set p-persist
PP 64
*------------Turn Digipeat function off
DPORTS $00
DIGI OFF
*-----------------NET/ROM PARAMETERS -------------------------
*------------Adds New Neighbors Heard if On
AUTOn off
*------------Broadcast Interval 30 minutes
BIN 1800
*------------KaNode Turned off
KPORTS $0
*------------Net Default Window Size
ND 7
*------------Enable BBS in NetNode
NETB 1
*------------Enable Connect Comand in NetNode
NETC ON
```

```
*------------Net Full Function Hours
NETFBBS 0-23
*------------NetNode On
NETN ON
*------------Network Connect for BBSs Only (Off permits ALL)
NETO 0
*------------Mail-for beacons turned off if 1 (port MASK)
NOBE $0
*------------NetRom PORTS 0,1,2,3
NPORTS $F
*------------Network Time to Live
NTL 15
*------------Obsolescence Count Initial
OCI 6
*------------Obsolescence Count Min for Broadcast
OCM 5
*------------Path Quality of heard nodes port 0
PQ 0 120
*------------Path Quality of heard nodes port 1
PQ 1 120
*------------Path Quality of heard nodes port 2
PQ 2 120
*------------Path Quality of heard nodes port 3
PQ 3 200
*------------TCP ON PORTS 0,1,2,3
TPORTS $F
*------------Min. DESt Quality to record
WQ 120
*--------------------BBS PARMS--------------------------------
*------------Kill on Forward OFF
AUtokill OFF
*------------Attended Mode ON (resolves GTOR problem)
AT 1
*------------Turns on AX25 Level 2
AX25L2 on
*------------Turns On AX25 Level 2 backoff (RNR)
AX25b $F
*------------BADUSER
*------------BUDLISTC
*------------Turns on BBS Node broadcast to network
BBSNODE 1
*------------Turns on BID function
BID 1
*------------Display Time for F2 Key (0=infinite)
f2timer 0
*------------FBB Compatibility/Compression On (SEE NEXT STATEMENT!)
FBBC 2
*------------FBB Compat./Compression OFF for MODEM PORT (PORT 4)
FBBC 4 0
*------------FBB Forward Group Number of Messages
FBBG 5
*------------Forward Time interval (- = every x minutes)
fti -10
```

```
*------------Network FBB Compatability/Compression On
NETFBBC 2
*------------Network FBB Forward Group Number of Messages
NETFBBG 5
*------------Network RLI Batch Size
NETXB 5000
*------------Network RLI Compression Option On
NETXO ON
*------------House Clean at x Minutes after 00:00
HCL 10
*------------Heard List turn off point if Dispatches < #
HE 10
*------------Initializes Printer before each File Printed (0=NO)
IN 0
*------------Makes Messages Private by default
MAKEP ON
*------------Maximum Bulletin Age
MAXB 10
*------------Allow Connects to SYSOP Keyboard
MCON ON
*------------Min Memory for Memory Low Busy Msg
MINM 32000
*------------Turns on Message Trailer Function
MS on
*------------Establishes New User default bits
NEW $4040
*------------Valid Non-Call in FROM field
OK WP REQWP REQQTH
*------------Turns on CallSign Server
PC ON
REQQTH ON
REQDIR ON
REQFIL ON
*------------Adds Received BID to R: line
RBid ON
*------------Screen Save Setting
SC 10
*------------Sets TimeZone
TIMezone EST
*------------Adjusts GMT offset from local time(CHANGE WITH DST!)
*------------(Mutually exclusive with RLine Time)
*GMtime 5
*------------Set R: line time offset from Local time (Mutually
*------------exclusive with GMtime.
RLinetime 5
*------------Turns on SYSOP Verification
VERIFY on
*------------Adjust color of Sysop BBS screen
WB 3 1
WF 3 14
*------------RLI Batch Size (non-compressed bytes)
XB 5000
```

```
*------------RLI Compression Option On
XO ON
*----------------NET/ROM NEIGHBORS----------------------------
NODE ADD AKR:K8EIW-1 3 192
NODE ADD #AKRU:K8EIW-2 3 200
NODE ADD #AKRU2:WB8BII-9 3 200
NODE ADD AKR220:K8EIW-11 3 200
AUTODEST On
NETNODE ON
NODE RESTORE
DESt REP AKR:K8EIW-1 K8EIW-1 3 0 230
*-------------ARP ROUTING TABLE-----------------------------
arp add w8upd -2 44 70 241 1 /24
arp add ka8z-4 -2 44 70 4 4
arp add k8eiw-4 -2 44 70 4 32
arp add wa8bxn-2 -2 44 70 4 6
arp add wb8lvp-2 -2 44 70 247 3
*------------Default TCP/IP ROUTE--------------
DRoute w8upd 5
```

MSYS.OTD:

```
(Update into MSYS with GETM command.)
     <<< RADIO CLUB MEETING TONIGHT @ 8:00 PM - CITY BUILDING >>>
***
```

MSYSK.OTD

```
      < Please try  ### MEDINA  Gateway 145.05 <-> 445.075 >
```

MSYS.REP:

```
NTS* ##### @2 @1    COMMENTS (NOT IN FILE!!!)
440* * @1 NO8M      This sends ALL 440xx Zip Code to NO8M
44001 * @1 N8ITP    This redirects 44001 Zip Code to N8ITP
44003 * @1 KB8AST   This redirects 44003 Zip Code to KB8AST
4403* * @1 N8ITP    This redirects 4403x Zip Code to N8ITP
44030 * @1 KB8AST   This redirects 44030 Zip Code to KB8AST
                   The result of the above is that all ZIP coded
                   mail not specifically addressed to the 44001,03
                   or 3x Zip Codes will go to NO8M.
441* * @1 W8GRG
44107 * @1 NO8M
46* * @1 W8LBZ      This redirects all Zip Code 46xxx to W8LBZ
* OHNET @1 ALLOH
* ARL @1 ARRL
* ALLUSA @1 ALLUS
* USA @1 ALLUS
* K8EIW @1 WB8BII
* WB8CQR @1 W8GRG
* WB8BXN @1 WA8BXN

NOTE:  THERE SHOULD BE NO BLANK LINES IN THE FILE!
       MAXIMUM of 150 entries in file!
```

MSYS.RMT:

PASSWORD TEST PHRASE 0123456789

(This file can be more than 2K in length if you desire! If you do much remote sysop activity, we STRONGLY urge a long password and utilize MSYS or MSYST as the remote terminal server.)

MSYSBBSB.DAT:

```
0
1  KA8Z
2  NO8M
3  KB8DRN
7  W8GRG
8  WA8BXN
9  NG8P
10
16 AD8I
17 W8CQK
18 KA0FPO
19
22 WA8DCE
23 K1LT
31 W8UPD
```

NOTE: 32 is the MAXIMUM number of BBS's you can forward bulletins to!

MSYSFWD.DEF:

```
FILE CONTENTS  COMMENTS  (DO *NOT* INCLUDE COMMENTS IN FILE!)

F1 w8upd       This is a straight forward out port 1 to a
w8upd          local PBBS.  Only Messages @ W8UPD will be forwarded.
-----          Necessary dashes! (5 minimum - 79 maximum)
F3 kq8m
@k8eiw-2       Forward through an adjacent TheNet node.
.c #cledx      Connect command given to TheNet node.
+to            Necessary triggers.
?from
?with
.c kq8m
#120           This extends the wait for connect to 120 seconds
+to            seconds after the .C KQ8M command was issued.
?with
?Busy
kq8m
-----
R3 ka0fpo      This is a forward with a request for Reverse Forward
@k8eiw-2
.c #cledx
+to
?from                          This example shows how to use the
?with                          include file KA0FPO.REV.  The .rev
.c kq8m-1                      extension indicates ka0fpo may
#120                           reverse forward from this system.
+to
?with
?Busy
.c ka0fpo
#120
+to
?with
?Busy
$c:\msys\fwd\ka0fpo.rev
------
f1 wa8dce
!25                            This indicates that WA8DCE will never
$c:\msys\fwd\wa8dce.rev        be forwarded to, but will reverse
--------                       forward all messages.
```

MSYSHOST.NET:

```
44.70.4.4 [ka8z] ka8z-4 ka8z.ampr.org        #Tom Barnes,canton,ka8z
44.70.4.6 [wa8bxn] wa8bxn wa8bxn.ampr.org    #Mike Pechura,lake
                                                   county,@wa8bxn
44.70.245.2 [kb7yw] kb7yw kb7yw.ampr.org     #Fred Peachman,
                                                brookfield @wb8lvp
```

NOTE: ALL ENTRIES FOR EACH HOST MUST BE ON ONE LINE!

MSYSPASS.DAT:

```
guest * 3 c:/msys/public
anonymous * 3 c:/msys/public
ka8z tom 7 c:/msys/public
mike wa8bxn 7 c:/msys/public
fred kb7yw 7 c:/msys/public
```

The format is: call pass privilege base directory
And is used for FTP access to the system.

MSYSPASS.IP

```
wa8bxn apple wa8bxn
terry n8hsp n8hsp
fred kb7yw kb7yw
swiss normal no8m
```

The format is: call pass callsign
and is used for Telnet access to the bbs.

Typical MSYSTNC.1
(AEA PK-232)

```
VHF OFF
HB 300
AWLEN 8
PARITY 0
CONMODE TRAN
TRACE OFF
HID OFF
BE 0
passall off
RAWHDLC ON
HP OFF
PP ON
KISS ON
RESTART  OR  HOST ON         !!!CONSULT YOUR TNC MANUAL!!!
```

NOTE: The MSYSTNC.x files are associated with the corresponding tnctype.x specified in the MSYS.OPT file. The above file COULD be named MSYSTNC.2 depending on your configuration. Only the Kantronics TNCs are fixed at MSYSTNC.0

NOTE: Consult your TNC manual to properly specify the proper parameters for your tnc to put it into the KISS mode.

The TNC-2 clones, such as the MFJ 1270B or 1274, may benefit greatly by using the JKISS EPROM which forces the tnc to be in the KISS mode all the time. The JKISS file is included on the distribution disk or is available as an EPROM from the Tucson Amateur Packet Radio organization.

INFORMATION FOR ADVANCED USERS

REPLACEABLE DIRECTORY NAMES

Beginning in MSYS 1.04 you can specify alternate directory names to be used by MSYS in place of the standard ones by using appropriate DOS set commands. The directory names that can be replaced and their default values are:

Default Dir Name	Set DOS variable to replace
FILES	MSYS:FILES
FWD	MSYS:FWD
GREP	MSYS:GREPDIR
HELP	MSYS:HELP
MAIL	MSYS:MAIL
YAPP	MSYS:YAPPDIR

As an example, assume you have a RAM disk at drive E: and have set up a directory called HELP on E: and copied the normal help files to this directory. You can then issue the following DOS command before starting MSYS:
set MSYS:HELP=E:HELP
and now MSYS will get the help files from the E:HELP directory.

NOTE: You must be very careful in using a RAMDISK since its contents are easily lost. Also, you should probably not us a RAMDISK that takes space out of the normal 640K address space.

NOTE: You are not limited to using RAMDISKS in specifying the disk and/or directory to be used for the above directories. You could specify floppy and/or hard disks as well.

To establish a separate directory that is "GReppable", use the DOS SET command similar to this: SET MSYS:GREPDIR=C:MSYS\GREP Don't forget to create the C:\MSYS\GREP directory!

REPLACEABLE FILE NAMES

You can also replace the following file names of files normally found in the "MSYS" (default) directory. Judicious choices can speed up operation and/or let you make use of multiple disk drives. Poor choices can be a disaster!

Default File Name	Set This DOS Parameter to replace:	Notes
MAIL.BAK	MSYS:MAIL.BAK	(6)
MAIL.BIN	MSYS:MAIL.BIN	
BBSTONTS.BIN	MSYS:BBSTONTS.BIN	(1)
MSYS.FWD	MSYS:MSYS.FWD	(2)
MSYS.HRD	MSYS:MSYS.HRD	(3)
MSYS.LOG	MSYS:MSYS.LOG	(3)
MSYS.MSG	MSYS:MSYS.MSG	(4)
MSYS.USR	MSYS:MSYS.USR	(4)
MSYSHOST.NET	MSYS:MSYSHOST.NET	(5)
MSYSBBSB.DAT	MSYS:MSYSBBSB.DAT	(2)
MSYSPASS.DAT	MSYS:MSYSPASS.DAT	(5)

Notes:

1. Very good candidate for ram disk (read only)
2. Good to put on ram disk if you are careful
3. These can be made NUL if you don't want them
4. If you like to live very dangerously, consider putting them on ram disk. Otherwise keep them on hard disk . Updated frequently during normal operation, they are very important files.
5. Good for ram disk if you run tcp/ip much. Read mostly.
6. A very strong suggestion would be to put this file on a different disk partition. Backup message files tend to get forgotten and expand at a very high rate. If they fill up the primary MSYS disk, MSYS will no longer be able to accept messages. If the MAIL.BAK directory is on a different drive, the most that can happen when that drive is full is that MSYS just will not make backup copies of the incoming messages.

Example 1. Lets move the mail.bak directory to the D: drive. It is appropriate to put the SET command in the AUTOEXEC.BAT so that it will happen on a reboot. You would put the statement:
SET MSYS:MAIL.BAK=D:\MAIL.BAK
in the AUTOEXEC.BAT file. MSYS will then look for the mail backup directory on the D: drive.

CAUTION MUTIL Functions always assume the C:\MSYS directory path! When doing MUTIL functions, you MUST specify the changed path!

Example 2. Assume you still have a ram disk at E: and have copied the normal contents of BBSTONTS.BIN to that disk. You could do the following DOS command:

SET MSYS:BBSTONTS.BIN=E:BBSTONTS.BIN
to cause MSYS to read the file from your RAMDISK.

NOTE: With all these changes you can make, you can configure a lot in terms of files now. Be careful of file sizes when thinking about where to put things. MSYS.LOG grows endlessly as does the MAIL.BAK directory. These two are VERY good candidates for another drive so as to NOT accidentally run out of disk space and bring the bbs to a humbled state.

NOTE: The programs in MUTIL assume the files are where they should be and totally ignore any replacements you may have made of file names and/or directory names.

NOTE: If you are not absolutely sure of what you are doing, don't (do anything with these facilities).

NOTE: Before you try any of this, MAKE BACKUPS OF EVERYTHING!

MSYS MSG FILE FORMAT:(For the adventuresome only!)

(If you know what you are doing, this will make some sense to you)

```
int head;   /* slot number of head of active list */
int free;   /* slot number of head of free list */
int next;   /* next message number to be assigned */

/* Each slot has the following definition: */
struct msghdrdef {
    unsigned  mmsgnr;
    unsigned char mmsgtype[3];
    int  mmsgsize;
    unsigned char mmsgto[7];
    unsigned char mmsgfrom[7];
    unsigned char mmsgatbbs[39];
    unsigned char mmsgdate[7];
    unsigned char mmsgtitle[40];
    long mmsgbits;
    unsigned char mmhbbs[8];
    unsigned char mmsgtries;
    unsigned char mmsgbid[15];
    int  mmsglink;
    };

/* Note: word alignment is forced for all ints & longs */
```

ROUTES.DAT FILE FORMAT:

This is an ASCII file that contains callsign prefixes and routings. The callsign prefix begins in the first byte. Comparisons are made with the BBS call for the length of the prefix from this file. The routing field begins in byte 9 (a tab character can be used to get here). Here are some sample entries:

```
4X    ISR.AS
9M2   MYS.AS
A     USA.NA
```

The idea here is that incoming messages for a given bbs will get the routing appended to the @BBS field. Thus if someone sent a message to 4X1ABC @ 4X1XYZ the @ field would become @4X1XYZ.ISR.AS

It is expected in your forward file you would either have provisions to route the country part (ISR) to some bbs, or the continent part (AS) to some bbs. Thus to route foreign messages, you might just have all the continent designators (like EU, AS, NA etc.) in your forwarding file or you can be more selective. Remember that you can have some countries along with the continent routings in your forward file. The part of the @BBS from left to right that is found in the forwarding file is the one that is used. If you do a lot of direct forwarding to various countries that can be accommodated; if you just dump all foreign stuff on a single station that is easy too. Feel free to modify the contents of ROUTES.DAT as you see fit; I have just included country prefixes for countries for which I seen message headers pass through my system.

The following program may be used to create a BBSLIST.DAT file from the BBSTONTS.DAT file supplied on distribution disk:

```
10 OPEN "BBSTONTS.DAT" FOR INPUT AS 1
20 OPEN "BBSLIST.NEW" FOR OUTPUT AS 2
30 IF EOF(1) THEN CLOSE:END
40 LINE INPUT #1,A$
50 I=1:C$="":  WHILE  MID$(A$,I,1)<>" "  :C$=C$+MID$(A$,I,1):
     I=I+1: WEND
60 D$=MID$(A$,35,6)
70 H$=MID$(A$,10,25)
90 IF LEFT$(D$,2)="99" THEN 30
100 X$=LEFT$(C$,1): IF X$>="0" AND X$<="9" THEN 120
110 X$=MID$(C$,2,1): IF X$>="0" AND X$<="9" THEN C$=" "+C$
120 PRINT #2,C$;TAB(8);H$;TAB(74);D$
130 GOTO 30
```

You may use a text editor to extract the program from this file. The result of this program is put in a file called BBSLIST.NEW. It may be renamed to BBSLIST if you don't have one already. Or you can merge your file with this one by using the following commands:
copy BBSLIST.DAT+BBSLIST.NEW TEMP
del BBSLIST.DAT
ren TEMP BBSLIST.DAT
Then run MUTIL function 20 to sort the file and discard the duplicates from the BBSLIST.NEW file.

MSYS under WINDOWS 3.1

To run MSYS in the background mode with Windows 3.1 you must make a PIF file called MSYS.PIF. The secret is when you get ready to save the MSYS.PIF file, show the path as:

```
     c:\windows\msys.pif
NOT
     c:\msys\msys.pif.
```

Try using the variable 130 for background and 110 for foreground to give the best results for timesharing.

To run MSYS under DesqView 386 Version 2.4 on a 386 or a 486 machine, try these values in the "Add Program" (AP) setup:

```
     O (for Other) F1 (to specify Program Information)
     PROGRAM NAME = MSYS
     Keys To Use = MS (or your choice!)
     Memory Size = (as big as you can make it - try 620K)
     Program  = C:\MSYS\MSYS.EXE (be sure to specify proper drive and
                                  sub-directory)
     Directory = C:\MSYS
     Writes text directly to screen     [Y]
     Displays graphics information [N]
     Virtualize text/graphics (Y,N,T)   [Y]
     Uses serial ports (Y,N,1,2)        [Y]
     Requires floppy diskette           [N]
     F1 for advanced options
     Starting Height:                   25
     Starting Row.                       0
     Starting Width.                    80
     Starting Column:                    0
     Close on exit (Y,N,blank)          [ ]
     Uses its own colors                [Y]
     Allow Close Window  command        [Y]
     Runs in background (Y,N,blank)     [ ]
     Uses math  coprocessor             [Y]
     Keyboard conflict (0-F):           [0]
     Share   CPU   when   foreground    [Y]
     Share EGA when foreground/zoomed   [Y]
     Can be swapped out (Y,N,blank)     [ ]
     Protection level (0-3)             [0]
```

To run MSYS under DesqView on a 386 machine, try these values in the "performance" section:

```
     foreground 3
     background 2
     common memory 15
     dos buffer for ems 5
     optimize communications Y
     allow swapping of programs N
     manage printer contention N
```

If you choose to use QEMM as your memory manager, the following line in your CONFIG.SYS file should give satisfactory results:

```
DEVICE=C:\QEMM\QEMM386.SYS RAM ROM
```

NOTE: A WORD OF CAUTION:
When using QEMM-386, the /ST:M option in the DEVICE= line will cause MSYS to not load properly. There may be other options which will also cause grief. These suggestions are not necessarily the optimum settings for your system, but they have been known to work.

If you are running QEMM version 7, and are NOT running under DesqView, running the command: LOADHI /LINK before loading MSYS will make additional memory available to MSYS.

If you are running QEMM and are running under DesqView, if you have no other need for graphics, you may use the: VIDRAM ON command to make additional memory available to MSYS.

MYIpaddr

Takes four integers in the range 0 to 255. It sets the IP address that the system will respond to. Example: MYI 44 70 4 6 If MYIpaddress is not specified, or set to all zeros, the TCP/IP support will be inactive.

MYTcpcall

Takes a callsign (with SSID) that will be used when sending arp and TCP/IP frames. It can be the same as the ID call in most cases.

TElnet

Takes an IP address and optional server number as parameters. A symbolic IP address may given to be looked up in the file MSYSHOST.NET. Default server is 23. This command is used to establish keyboard to keyboard connection with a TCP/IP station. Examples: TE [44.70.4.10]
te hsp (hsp would have to be defined in MSYSHOST.NET)

TElnet operates much like C# normally used to connect to AX.25 stations. ^Cd is used to disconnect.

TPorts

Takes a port mask value as an argument. This number is used the same way as in the other similar commands: each port corresponds to a port (port 0 is the rightmost bit) and when the bit is set to 1 for a given port that port is available for TCP/IP use (particularly ARP broadcasts).

ARP

Used without an argument lists the known callsign/IP address correspondence determined by TElnet connects. An optional argument, CLEAR, may be used to erase all of the ARP entries.

FTP

Takes a host-id as an argument (it can be either an IP address in the square brackets or a symbolic IP address). It establishes a connection to the FTP server of the specified system. FTP is the File Transfer Protocol used in TCP/IP.

SMTP

Allows local access to the SMTP server (mainly for test purposes).

Note: The following servers are currently implemented in MSYS:

7	Echo
9	Discard
21	BBS
23	Telnet
25	Simple Mail Transfer Protocol
87	Local keyboard

Files related to TCP/IP

MSYSHOST.NET

This is an ASCII file you create with an editor (such as the MSYS EF command). Each line in the file begins with an IP address that includes the periods but not the square brackets around it. Following this there is at least one space and/or tab character, then the list of symbolic names (or aliases, or nicknames, what ever you want to call them). You may place comments on the line by beginning the comments with a pound sign (#). Example entry line for this file:

```
44.70.4.6 bxn mike wa8bxn # kirtland, oh
```

This line would allow "bxn" "mike" or "wa8bxn" to be used as arguments in the TElnet and FTP commands. Note that WA8BXN here just happens to look like a callsign. The actual callsign (and SSID) associated with a particular IP address is determined by ARP (address resolution protocol) broadcasts. These symbolic names you define in this file simply relieve you of having to remember (and type) complete IP addresses. Nothing more, nothing less! See sample for MSYSHOST.NET file.

MSYSPASS.DAT

This file contains the user names and passwords for FTP users. Each line contains 3 or 4 items: username password access directory Username is the name the user will use to log in. It can be anything for a given user. It could be their callsign or something else. The password can also be anything. If you put * as the password in the file for a particular, any password given by the user will be accepted. The third entry on each line, access, is a number that says how much access the user will have to your system: 1 means can only read files, 3 means read and write (but not replace or delete) and 7 means read, write, replace and delete. You shouldn't set 7 for any user since the passwords given by a user can be monitored by others. The last item which should always be given is the root directory that the user will have. They may never go closer to the real root than what you give here.

Here are some examples:

anonymous * 1 files	This allows user anonymous to read any files in MSYS/FILES and any of its subdirectories. Any password can begiven for this user. Anonymous with * for the password is an expected "standard" entry for TCP/IP systems to give access to totally public files. Include it in your system.
apple runtime 3	This allows user apple (who must give password runtime) read/write access to your entire disk (DEFAULT DIRECTORY IS THE REAL ROOT DIRECTORY!)

USER ACCESS level chart

```
Level 1: CD DIR GET PWD QUIT
Level 3: PUT MKDIR
Level 7: PUT (to replace existing file) DELE RMDIR
```

If you are going to allow level 7 access (i.e., can do everything) I would suggest you also assign a directory that doesn't allow access to any needed files. For example, you might set up the following entry in the MSYSPASS.DAT file:

WA8BXN MIKE 7 /MSYS/FILES/WA8BXN

You will have to create a directory WA8BXN in the files directory. Now stations can log in with WA8BXN MIKE and do anything in /MSYS/FILES/WA8BXN and any directories found in this directory.

If you set up ANONYMOUS * 1 /MSYS/FILES then the user ANONYMOUS can read anything put in the WA8BXN subdirectory of FILES (as well as in FILES itself) but not write or otherwise change anything. The rest of your files should be pretty well protected.

FTP COMMANDS:

When you use the FTP command to connect to another system you can type: HELP to get a list of the supported commands. They currently include:

```
CD dirname     To move around in the directory structure
CD             with no arguments gives the current directory
DELE filename  To delete given file
DIR filespec   gives a directory listing
GET filename   downloads a the given file
MKDIR          makes a directory
PUT filename   uploads the given file  Syntax is:
     put [localfilename] remotefilename
     If the optional localfile name is omitted it is assumed to
          be the same as the remote file name.
PWD            gives the current directory
QUIT           disconnects
RMDIR dirname  removes the given directory

FTP function   will give an estimate on the number of bytes
               transferred during GET and PUT commands.
```

IP Routing:

What does this mean? Its sort of like digipeaters for the TCP/IP world. If a TCP/IP user sets you up as a route for IP frames going to a particular IP address MSYS will now repeat them if it has (or can get through an ARP request broadcast) a known route to the DEStination IP address. Consider the following fictitious example: W9AAA [44.60.1.1] is in Indiana and wishes to connect to W3AAA [44.80.1.1] in Pennsylvania but can't do it directly. W9AAA can reach W8AAA [44.70.1.1] in Ohio who can in turn reach W3AAA. W9AAA will set up the route to [44.80.1.1] to be W8AAA (an MSYS system). When attempting to connect to W3AAA (using Telnet, for example) will send out a SYN IP frame (connect request) to callsign W8AAA but with an IP address of [44.80.1.1], the IP address of W3AAA. W8AAA upon hearing this will check its ARP table to see if it knows how to get to [44.80.1.1]. If the entry is not there, W8AAA will do an ARP broadcast on all the ports enabled for TCP/IP. W3AAA will respond on one of them and this will put an entry in the ARP table. Now when a SYN is heard from W9AAA it will be repeated by W8AAA and thus heard by W3AAA.

To use IP ROUTING yourself in MSYS you use the arp add command. When adding a given IP address you would put the call of the TCP/IP station you want to use as an IP repeater in place of the call that belongs with the IP address. You can still use normal AX.25 digipeaters (up to 8) to reach the TCP/IP station. If the path from W8AAA to W3AAA in the above example was not direct, then the following arp entry could be done at W8AAA, assuming that K8AAA is a normal AX.25 digipeater: arp add W3AAA port# 44 80 1 1 K8AAA

NOTE: For more on TCP/IP refer to The KA9Q Internet Software Package.

In the TCP/IP related ARP command, use: -2 as the port number for routing through the network interface.

TCP/IP connects are rejected if not enough memory or MCON is exceeded.

EDITVIEW (ED Command)

Commands The ED command used from the MSYS command mode can be used to edit small text files. The file is limited to records each 80 bytes or less in length. By default up to 200 records may be in the file. This can be modified by giving the maximum number of lines after the file name. The intent of this editor is to do little things without having to take down the system to run a more powerful editor. These might include editing message texts, the forward file, etc.

ED is a full window editor (it adjusts its operation to fill the window it is run in). It must be run in a window that is 80 characters wide. Thus using the default window definitions it can be run in the normal command window (0) or you can use the CMdwindow command to first go to window 4 (the one you get when you press F4) and then use the ED command. You will then be using the entire screen for your editing.

ED is pretty much a what you see is what you get editor. There are two modes of operation. When you start the editor you are in full screen or edit mode in which changes you make to the screen are also made to the file. In this mode you get a line at the bottom of the window that gives such information as the current line number and the name of the file being edited. The other mode is command mode in which you give commands to be done by the editor. To enter command mode press the Escape key. The bottom line will change to Edit Cmd: and await your command. The command you type will appear on this bottom line. To get a list of available commands type Help. You should see something that looks like the following:

```
   Find   locates given string .  Example:  F wa8bxn
          Notes: Not case sensitive .
   F10    from edit mode is find same string again.
   Insert puts blank line at beginning of file
   QUIT   Abandons edit
   SAVE   Saves file and exits
   DELete Delete specified # of lines from  file  beginning with
          current line
   Return key in edit mode with insert on inserts blank lines

Press a key to return to edit
```

Use of the keypad to move around in the file is what is expected of the labeled keys, with the following enhancements. If you are somewhere in a line and you press the Home key, you go to the beginning of the current line. If you are the beginning of the current line and press Home, you go to the beginning of the screen. If you are at the beginning of the screen, pressing Home takes you to the beginning of the file.

EDITFILE (EF Command)

The EF command used from the MSYS command mode can be used to edit small text files. The file is limited to records each 80 bytes or less in length. By default up to 200 records may be in the file. This can be modified by giving the maximum number of lines after the file name. The intent of this editor is to do little things without having to take down the system to run a more powerful editor . These might include editing message texts, the forward file, etc. This isn't a full screen editor but rather a line editor that was influenced in its design by the UNIX editor. Plain and simple nothing fancy is found in it. To begin editing, use the command EF filename when you have the cmd: prompt. The filename can contain drive and path information. If the file does not exist you will be given the option to create one by that name. To create a new file this way, use the Append command. Type the lines of the new file and then a line with a period at the beginning to exit append mode. Then use the commands Write and Quit (each followed by the return key) and you have created a new file. All of the commands can be shortened to the first letter of the command, except for the DElete command which requires at least the first two letters.

Here is information about the edit commands:
(i.e., a line number) makes that line the current line and also displays it

+ # Makes the current line # more than it is now . If # is omitted, then the current line is incremented by one.

- # Same as + # except current line number is decremented.

<ESCAPE> Using the Escape key followed by the Return key abandons edit with no changes made to the file (unless you used the Write command yourself).

<RETURN> Pressing the Return key alone displays the next line and makes it the current line.

Append enters input mode, placing the following lines of input AFTER the current line. A line beginning with a period terminates append mode.

Bottom makes the last line the current line and displays it.

Change can be used to replace part of the current line. The first non-blank character after the command is the quote character. The operands are quote-character string-to-replace quote-character replacement-string quote-character . For example,

C /YXZ/abc/

would replace XYZ in the current line with abc. The line is printed after it is displayed.

DElete can be used to delete the current line. If a number is specified as an operand, it is the number of lines to delete beginning with the current line.

Down moves the current line down the number of lines specified (or 1 if none is specified).

Find uses the first non-blank character after the command word as the beginning of the search string. The search is from the current line on in the file. Case is NOT important. If a match is found the line is displayed and that line becomes the current line. If no match is found the current line remains unchanged.

Help displays the list of commands.

Insert is like Append except that the new lines are placed BEFORE the current line.

Near displays the contents of the file 5 lines before and 4 lines after the current line which remains unchanged.
Print with no operands displays the current line. Give 1 operand it displays the line with that number. Given 2 operands (separated with a blank or comma) it displays from the first to the second line number. The current line number remains unchanged.

Quit is used to exit the editor after the file has been written.

Top makes the first line (numbered 0) the current line and displays it.

Up decrements the current line number by the number given or 1.

View displays the current line and the 9 lines that follow it. The current line remains unchanged.

Write causes the file to be written to disk with any changes that have been made.

^F is used to search again for the same search string give with last use of Find.

To start the editor, type EFile filepathname [# of lines max].
Examples:
ef msys.fwd 250
ef mail/msg25.dat
efile a:f1.c

Epilog

Notes on sending me messages with questions/suggestions/etc.: I do try to respond to your personal messages. Keep in mind I do get quite a few at times. If you want an answer from the author, you need to send the question to the author personally! Messages that contain demands (add this feature or correct this bug immediately and put out a new release or I won't use your program) generally are saved in the bit bucket. For a free program, please be considerate! Try to put just one thought/question in each message. If you bundle up a whole bunch, I don't respond until I can address all your issues. If you send follow-up messages, please give a sentence or two summary of the problem so that I may understand what you are talking about. Just saying "I tried what you suggested and it still doesn't work, anything else to try?" doesn't give me much of a clue about the situation. Although you may have only one problem you are dealing with and can easily remember its entire history, I don't have that luxury here! If you find what you think is a bug, please try to tell me how to reproduce it, what you are seeing, and what you think should be happening. The harder the problem, the longer it takes me to work on it and respond. When you run into any sort of problem, simplify everything as much as possible. Remove all statements from config.sys, autoexec.bat, msys.opt and msys.do that are not essential to reproduce the bug. Believe me, this will probably help you find at least half of the problems and their solutions yourself.

Use and Enjoy!