<<< VERSICOM >>>

Versatile Communications Software

User's Guide   Version 2.56

Revised: July 13, 1989

Gregory F. Welch
Jet Propulsion Laboratory
4800 Oak Grove Drive
Pasadena, CA 91109

U. S. Government Sponsorship under NASA Contract NAS7-918 is acknowledged.

TABLE OF CONTENTS

1.0.0 Introduction 3
1.1.0 What Is Versicom? 3
1.2.0 About This User's Guide 4

2.0.0 Installing Versicom 5
2.1.0 First Time Installation 5
2.2.0 Modifying a Setup 6

3.0.0 Starting Versicom 9
3.1.0 Command Line Parameters 9

4.0.0 Understanding the SOE 10
4.1.0 Creating the SOE Key 11

5.0.0 The Terminal Screen 12
5.1.0 The Menu Bar 12
5.2.0 The Instruction Bar 12
5.3.0 Terminal Screen Keys 13
5.4.0 The User Command Pull-Down Menu 13
5.4.1 Adding User Commands 14
5.4.2 Editing/Removing User Commands 14
5.4.3 Special User Command Characters 16
5.4.4 Using the User Commands 17
5.5.0 The Function Key Pull-Down Menu 17
5.5.1 Adding Function Key Assignments 18
5.5.2 Editing/Removing Function Key Assignments 18
5.5.3 Special Function Key Assignment Characters 19
5.5.4 Using the Function Keys 19
5.6.0 The History Pull-Down Menu 20
5.6.1 Using the Command History 21
5.7.0 The Setup Pull-Down Menu 21
5.7.1 Save a Setup 22
5.7.2 Load a Setup 22
5.8.0 The RunLog Screen 23
5.8.1 Executing a Command From the RunLog 23
5.8.2 RunLog Keys 24
5.9.0 The Capture Pull-Down Menu 24
5.9.1 Printer Capture 26
5.9.2 Disk Capture 26
5.10.0 The DOS Shell Option 26
5.11.0 The Quit Option 27
5.11.1 Quitting and Terminating the Connection 28
5.11.2 Quitting and Leaving the Connection Active 28

6.0.0 Special Files 30
6.1.0 Setup Files 30
6.2.0 Macro Files 31
6.3.0 Installation Files 31

7.0.0 Macros . . . 33
7.1.0 What is a Macro? . . . 33
7.2.0 Macro Commands . . . 33
7.2.1 WaitFor . . . 34
7.2.2 Send . . . 34
7.2.3 Install . . . 34
7.2.4 Hide-Keys . . . 35
7.2.5 Show-Keys . . . 35
7.2.6 Delay . . . 36
7.2.7 Quit . . . 36
7.2.8 Prompt . . . 37
7.2.9 Repeat . . . 37
7.2.10 Until . . . 38
7.2.11 Timeout . . . 38
7.2.12 Beep . . . 38
7.2.13 Run . . . 39
7.2.14 Upcase . . . 39
7.3.0 Logical Commands in a Macro . . . 40
7.3.1 Receive . . . 40
7.3.2 Quiet . . . 41
7.4.0 Special Macro Characters . . . 41
7.5.0 A Sample Macro . . . 42

8.0.0 The Special @@ Commands . . . 43
8.1.0 @@Do . . . 43
8.2.0 @@Prompt . . . 44
8.3.0 @@Upcase . . . 45
8.4.0 @@HideKeys . . . 45
8.5.0 @@ShowKeys . . . 45

9.0.0 Using a Mouse With Versicom . . . 46

10.0.0 Errors and Other Messages . . . 47
10.1.0 Non-Fatal Messages . . . 47
10.2.0 Fatal Messages . . . 51

11.0.0 Glossary . . . 53

1.0.0 Introduction

1.1.0 What Is Versicom?

Versicom is a simple communications program which provides you with many of the desireable features normally available only on a dedicated or "dumb" terminal.

Most dedicated terminals will offer you the wonderful ability to correct any typing mistakes. However, this is not normally the case when using a personal computer. That is where Versicom comes in!

Although Versicom does not emulate any specific standard terminal mode (such as VT100), it does provide you with the following basic features:

- Full screen command editing capability
- Full mouse device support (Microsoft compatible)
- Custom designed command strings (User Commands)
- Command assignable function keys (Function Keys)
- A list of recently used commands (History)
- Options to save or load User Command and Function key assignments (Setup)
- A log which saves every line on the screen as it scrolls off the top (RunLog)
- Capture communications to either the printer or an ASCII disk file (Capture)
- A Disk Operating System shell to allow DOS commands to be executed (DOS)
- The option to quit while still leaving the connection intact (Quit)

Versicom is able to provide you with full screen command editing capability because it requires that a Start of Entry (SOE) character be present at the start of every command you wish to send, (see **4.0.0 Understanding the SOE**). At first, this may sound like an inconvenience. However, the SOE character can be automatically provided by Versicom whenever the host system sends a designated prompt. Or in the unlikely event that this is not convenient, it can be manually placed anywhere on the screen.

In addition, the ability to place an SOE anywhere on the screen allows you to create new commands out of any data that may have been previously received from the host computer. This can be a great time saver if you have to type long file names.

1.2.0 About This User's Guide

Throughout this document, you will see different words (text) printed in different styles. These different styles were chosen with the intention of making certain tasks easier to understand. The following list of styles have a specific meaning:

Style	Meaning	Example
Bold	The beginning of a major topic	**1.0.0 Introduction**
Underline	A sub-major topic	1.1.0 What Is Versicom?
Underline	A subtopic, or an important note	5.7.1 Save a Setup
SMALLCAP	A Versicom command	UPKEYS
Italics	Emphasis, or sample text	*@@Do MyMacro*
Redline	A certain key on the keyboard	<Enter>

This user's guide is meant not only to get the first time user started, but also to serve as a reference guide for continuing use. Certain topics (such as macro creation) are discussed in some detail here, but not enough can be said about the value of simply "playing with" Versicom, and its different capabilities.

Although an attempt has been made to give thorough examples of each feature, you will certainly become more familiar with those features with the continuing exploration and use of Versicom.

2.0.0 Installing Versicom

2.1.0 First Time Installation

1) By examining your modem switch settings and (or) modem manual, determine the following information and fill in the blanks below:

   a) Port Number________________ (e.g. COM1:)
   b) Port Address________________ (e.g. 03F8 Hex)
   c) Interrupt________________ (e.g. IRQ4)

   NOTE:
   Generally, most modems are set up with either of the following default values:

Default for COM1:		Default for COM2:	
COM1:	(a above)	COM2:	(a above)
03F8 Hex	(b above)	02F8 Hex	(b above)
IRQ4	(c above)	IRQ3	(c above)

2) Insert the Versicom installation disk into any floppy drive such as A:.

3) Make that drive the default drive by typing the drive name, followed by a colon, and then pressing <Enter>. For example, type *A:* and press <Enter>.

4) Begin the installation by typing *INSTALL C:* and pressing <Enter>. (If needed, change *C:* to the disk drive where you wish to install Versicom)

   The installation batch file takes care of several tedious tasks for you. First of all, it will create a special subdirectory called Versicom on your C: drive (or the disk drive specified above in step 4). Next, it will copy several files to your new Versicom subdirectory:

VERSICOM.EXE	- The Versicom program
VER_INST.EXE	- The Versicom Installation program
1100A.MAC	- A sample macro to connect to the JPL Univac 1100A
1100B.MAC	- A sample macro to connect to the JPL Univac 1100B
ON_ILAN.MAC	- A sample macro to direct connect to JPL's ILAN
OFF_ILAN.MAC	- A sample macro to disconnect from JPL's ILAN

OMIT.MAC	- A sample macro to exit JPL's @ED editor without changes
FINOFF.MAC	- A sample macro to log off of JPL's Univac 1100A/B
UTS.BAT	- A sample DOS batch file to start Versicom

Note that the sample DOS batch file (UTS.BAT) will be copied into the "root" directory of the C: drive (or drive specified above). You may then move the UTS.BAT file to any subdirectory, as long as that directory is included in the PATH for your computer, (see DOS manual if needed).

6) Finally, the installation batch file will ask you to press a key to continue, and then it will automatically start the Versicom installation program (VER_INST.EXE). It is this program which then allows you to select a series of default options for Versicom. (See the following section 2.2.0 Modifying a Setup for instructions on using VER_INST.EXE)

2.2.0 Modifying a Setup

The following steps will assist you with the operation of the Versicom installation program, VER_INST.EXE. This will be run automatically during first time installation, and can then be used to subsequently create additional installation files, (see 6.3.0 Installation Files).

1) Press any key (or simultaneously press the left & right mouse buttons) to cause the introduction to disappear. You can also wait for the default viewing time to pass, approximately five seconds, and the screen will disappear automatically.

2) Press the <Space> key to toggle through the choices for the communications port Versicom should use to access the modem, (use the Port Number you noted in 2.1.0 First Time Installation, step 1a).

3) Press <Enter> to accept the choice, and advance to the next option.

4) Press the <Space> key to toggle through the choices for the base address of the communications port/modem, (use the Port Address you noted in 2.1.0 First Time Installation, step 1b).

5) Press <Enter> to accept the choice, and advance to the next option.

6) Press the <Space> key to toggle through the choices for the hardware interrupt request number (IRQ) used with the modem, (use the Interrupt you noted in 2.1.0 First Time Installation, step 1c).

7) Press <Enter> to accept the choice, and advance to the next option.

8) Press the <Space> key to toggle through the choices for the baud rate (or speed) to be used.

9) Press <Enter> to accept the choice, and advance to the next option.

10) Press the <Space> key to toggle through the choices for the number of stop bits.

11) Press <Enter> to accept the choice, and advance to the next option.

12) Press the <Space> key to toggle through the choices for the parity to be used during transmission.

13) Press <Enter> to accept the choice, and advance to the next option.

14) Press the <Space> key to toggle through the choices for the number of data bits to be used.

15) Press <Enter> to accept the choice, and advance to the next option.

16) Type the default set of characters to be recognized as a prompt from the host computer. Versicom will display a Start of Entry (SOE) character following the receipt of these characters. For example, if the host computer you will most often be using sends back a greater-than character (">") every time it is ready for new input, type > to use that as the default prompt. (See also 7.2.8 Prompt and 8.2.0 @@Prompt).

    If you wish, you may leave the prompt blank, and the SOE will not be printed automatically.

    NOTE:

    Making the Versicom <Enter> character, the pipe ("|"), the first character in the prompt string, will cause the string to be recognized as a prompt ONLY when it follows a carriage return.

    Thus, if the prompt string appears in the middle of a line, it will not be followed by an SOE character.

17) Press <Enter> to accept the choice, and advance to the next option.

18) Press the <Space> key to toggle through the choices for the RunLog foreground and background colors, pressing return after each choice. Choose the colors so that they are readable, but different from the normal terminal screen's colors of YELLOW on BLACK.

19) Press the <Space> key to toggle through the choices for the video display mode. Select the mode which is valid AND desirable for use with your monitor.

For example, a selection of the "Color 50 line VGA mode" would be invalid with a CGA graphics system. It would be valid with a VGA graphics system, but it may be undesirable because of the character size.

20) Repeat the above steps, until the desired parameters have been obtained.

21) Press the <Alt> key and the <I> key at the same time, to invoke the installation menu.

22) Press the <DownArrow> once, so that Save is highlighted.

23) Press <Enter> to choose the save option.

24) Press <Enter> to save this installation with the default file name *Versicom.INS* if this is your first time running the installation program.

NOTE:
Versicom.INS is the default installation file which must always be present when running Versicom. Therefore, if this is the first time you have run the installation program, you must use the default name *Versicom*.
(See 6.3.0 Installation Files)

If this is *not* the first time you ran the program, and you wish to enter a new name for the installation, simply type the new name and press <Enter>. This new alternate installation can now be loaded by running a macro, which contains the macro command *Install "install-filename"*, (see 7.2.3 Install).

25) Press the <Alt> key and the <Q> key at the same time, to quit the installation program.

NOTE:
If you have not saved the installation file, you will be asked whether or not you wish to save the default installation file before quitting. If you have not specified a new default name by using the <Alt><I> & SAVE

commands, then the default file will be Versicom.INS.

3.0.0 Starting Versicom

To begin running the Versicom program, at a DOS prompt follow these steps:

1) Type *CD \Versicom* and press <Enter>.
2) Type *Versicom* and press <Enter>.

Note that here, Versicom will be started in an inactive state. To begin actual communications, you must complete two tasks as follows:

1) You must *connect to a host computer*, either by a direct connection, or via the telephone. This includes dialing (if needed) and logging on to the host.

2) You must also *load a setup file* which contains a function key with the SOE character defined, (see **4.0.0 Understanding the SOE** and 4.1.0 Creating the SOE).

You can accomplish either of the above two tasks by including *command line parameters* when you start Versicom, or by manually completing the tasks once Versicom is running.

3.1.0 Command Line Parameters

To begin running the Versicom program, load a Setup file, and (or) start any number of consecutive macros, you can include command line parameters when you start Versicom. This procedure will save you additional steps usually needed such as using the Setup pull-down menu to load a file, and manually starting a macro (or macros) with the @@Do command.

With one command line, you can begin Versicom, load a Setup file (User Commands and Function Keys), and begin a macro (or several consecutive macros) such as one which will dial a telephone number for you.

To use the command line parameters, at a DOS prompt follow these steps:

1) Type *CD \Versicom*

```
2)    Type Versicom [SetupFile] [Macro1] [Macro2] ... [Macro(n)]
             ^           ^           ^
             |           |           |
             |           |           Optional macros to execute
             |           |           (*.MAC files, extension is optional)
             |           |
             |           Optional Setup file (*.VER files, extension
             |           is optional)
             |
             Versicom program
```

For example, if you had previously created a Setup file called "MYSETUP.VER", (see 6.1.0 Setup Files), and you were going to use the included sample macro file "1100A.MAC", at a DOS prompt you could type:

```
CD \Versicom
Versicom MYSETUP 1100A
```

This example would start Versicom, load the setup file "MYSETUP" (if it exists), and begin executing the macro 1100A.

As an added convenience, you may create a DOS "batch" file to run Versicom for you. A sample of such a batch file was copied to the "root" directory of your disk when you initially installed Versicom. The sample is called UTS.BAT, and can be modified with any text editor, (see your DOS manual for information on batch files).

4.0.0 Understanding the SOE

Because Versicom allows you to move freely around the terminal screen, (and the RunLog), there must be a way to tell Versicom that you are ready to send some characters (a command) to the host computer. You can do this by using the Start of Entry or SOE character, which looks like ▸.

The SOE character (▸) is probably the most significant screen character you will use while communicating with Versicom. The SOE character is shaped like a small triangle, turned on its side so that it points to the right.

Versicom uses the SOE character as a marker for the beginning of a line of characters (a command) to send to the host computer. The SOE character will always "point" to the start of an entry, and will normally be highlighted.

Therefore, it is possible to move about the screen freely, placing an SOE character at any position, to begin an entry. In fact, a past response from the host computer can be used to "create" a command, sometimes saving typing.

4.1.0 Creating the SOE Key

Now that you understand what the SOE does, and where it goes, surprise...you must create an SOE key! Why doesn't Versicom do this for you (you may ask)? By requiring you to create the SOE key, we allow you to assign it to any function key (or User Command) that you want.

Remember, Versicom does not set up an SOE key when you initially install it. To be able to use Versicom to send commands to a host computer, you must define the SOE key as described below.

To create the SOE key you must assign the five characters "<SOE>", (less-than sign, S, O, E, greater-than sign), to a function key or User command. For example, we will demonstrate assigning the SOE to function key number one, (the <F1> key).

1) From the Terminal Screen, (see **5.0.0 The Terminal Screen**), press <Alt><F> to activate the function key pull-down menu.

2) With the <F1> entry highlighted, press the <Ins> key to activate the Function Key entry/edit window.

3) Type the characters <SOE> (five characters) and press <Enter>.

4) Press <Esc> to exit the Function Key window.

5) Save this setup under any name you wish to use, (see 5.7.1 Save a Setup). (Saving the Setup file is optional.)

You now have the SOE character assigned to the function key <F1>. You can test it by pressing the <F1> key to see if the SOE character (▸) appears on the screen. You should see the single, triangular-shaped character on the screen (at the current cursor position) each time you press the <F1> key.

If you do not see the SOE character when you press the <F1> key, then you should check to see that you have all five of the characters <SOE> assigned to the function key <F1>.

5.0.0 The Terminal Screen

The terminal screen is the empty screen you see immediately following the "Logo Screen" after starting Versicom. The screen is empty except for the Menu Bar across the top of the screen, and an Instruction Bar across the bottom of the screen.

This screen is where most of the sending (typing) and receiving is done with Versicom. The whole screen can be accessed freely with keys listed below.

At any point on the screen, an SOE character can be placed to form a new command line. When the <Enter> key is pressed, all characters following the SOE character up to (and including) the character at the cursor position when <Enter> is pressed, are sent to the host computer. If there is no SOE character to the left of the cursor, on the same line, then the command is ignored, and you are notified via a message in a pop-up window.

As the screen fills from general work, every line of text that scrolls off the top of the terminal screen is placed into the RunLog, (if your computer has enough memory).

5.1.0 The Menu Bar

Across the top of both the Terminal Screen and the RunLog Screen, you will see several words which represent either pull-down menus, or other options which are available to you from that screen.

Each menu or option can be selected by pressing the <Alt> key and the first letter of the menu/option at the same time, or by clicking on the menu name with the mouse.

5.2.0 The Instruction Bar

The last line on both the Terminal Screen and the RunLog Screen is always reserved for short instructions which may help you with whatever activity you are currently doing.

If you are ever in doubt about what to do next, or what key you should press, always look at the instruction bar for a brief explanation of any options available to you at that time.

As you select new menus or options, new instructions will appear on the instruction bar, reflecting your current activity.

5.3.0 Terminal Screen Keys

The following keys can be used from the terminal screen, or the RunLog screen. See below for further explanations of menus activated by special key presses.

<Alt><U>	User Commands pull-down menu
<Alt><F>	Function Keys pull-down menu
<Alt><H>	Command history (last 20 commands sent)
<Alt><S>	Setup file pull-down menu
<Alt><R>	View/use the RunLog
<Alt><C>	Capture pull-down menu
<Alt><D>	Exit to a DOS shell
<Alt><Q>	Quit the Versicom program

<Home>	Move cursor to the start of the current line
<End>	Move the cursor to the end of the current line
<Ctrl><Home>	Move the cursor to the top of the screen (current column)
<Ctrl><End>	Move the cursor to the bottom of the screen (current column)
<Ins>	Insert a blank space at the current cursor position
<Del>	Delete the character currently under the cursor
<Backspace>	Move the cursor to the left one character
<LeftArrow>	(Same)
<RightArrow>	Move the cursor to the right one character
<UpArrow>	Move the cursor up one line
<DownArrow>	Move the cursor down one line
<Shift><Tab>	Erase all characters to the right of (and under) the cursor
<Ctrl><C>	Sends a BREAK signal to the host computer (via hardware)

5.4.0 The User Command Pull-Down Menu

Versicom allows you to maintain a list of up to twenty commonly used commands, which can be saved in a Setup file. Each command must be given an eight character name which will be used to identify that specific command, and the characters which actually make up that command. User Commands can either be direct commands to the host computer, or they may start a Versicom macro, (e.g. *@@Do MyMacro*).

Once you have created a list of User Commands, (see 5.4.1 Adding User Commands below), you can access the User Command menu to view a list of your preferred commands. The list will show the names which you previously gave each command. For instance, you could have commands with names such as QUIT, LOGON, COMPILE, MAP, etc.... In addition, you can change or edit any of the names or the commands previously entered, (see 5.4.2 Editing User Commands below).

5.4.1 Adding User Commands

To place a new User Command in one of the twenty available locations of the User Command list, follow these steps:

1) Press <Alt><U> to activate the User Command pull-down menu.

2) Use the <UpArrow>, <DownArrow>, <Home>, <End> keys (or the mouse) to select a blank User Command entry; one with no name or command.

3) Press the <Ins> key to activate the User Command entry/edit window.

4) Type the name to be used for this command (up to eight characters).

5) Press <Enter> to accept the name, (the cursor will move to the command field).

6) Type the command associated with this name (up to 78 characters).

7) Press <Enter> to accept the command, (the entry/edit window will disappear).

8) Repeat steps 1-7 for more User Commands, or press <Esc> (or the left & right mouse buttons simultaneously) to remove the User Command pull-down menu.

If you make any mistakes while entering either the name or the command, you can edit either of them by following almost the same process, (see 5.4.2 Editing User Commands below).

5.4.2 Editing/Removing User Commands

If you need to change (or remove) either the name or the command portion of a previously entered User Command, you can do so by following these steps:

1) Press <Alt><U> to activate the User Command pull-down menu.

2) Use the <UpArrow>, <DownArrow>, <Home>, <End> keys, the mouse, or press the first letter of the command to select the User Command entry to be used. You should use the names that appear in the left column to help you recognize the desired command to be edited.

3) Press the <Ins> key to activate the User Command entry/edit window.

4) *To change the name...*

Use the <Home>, <End>, <LeftArrow>, <RightArrow>, <Ctrl><RightArrow> or <Ctrl><LeftArrow> keys to position the cursor where the correction is to be made. Once the cursor is in the desired position, you can type over the old name, or use the <Ins> and <Del> keys to add or remove characters. Notice that when you move the cursor for the first time, (or type new characters), the background color changes to accent the fact that a change is being made.

To remove (erase) the name...

Continue to press the <Space> key until the command name has been erased.

To leave the name alone...

Proceed to step 5.

5) Press <Enter> to accept the name, (the cursor will move to the command field).

6) *To change the command...*

Use the <Home>, <End>, <LeftArrow>, <RightArrow>, <Ctrl><RightArrow> or <Ctrl><LeftArrow> keys to position the cursor where the correction is to be made. Once the cursor is in the desired position, you can type over the old command, or use the <Ins> and <Del> keys to add or remove characters. Notice that when you move the cursor, or when you type new characters, the background color changes to accent the fact that a change is being made.

To remove (erase) the command...

Continue to press the <Space> key until the command has been erased.

To leave the command alone...

Proceed to step 7.

7) Press <Enter> to accept the command, (the entry/edit window will disappear).

8) Press <Esc> (or the left & right mouse buttons simultaneously) to remove the User Command pull-down menu.

Remember, if you want to save any changes made to User Commands, you must save the modified setup, (see 5.7.0 The Setup Pull-Down Menu below). Saving the Setup will ensure that the next time you use it, the changes you made will again be there.

5.4.3 Special User Command Characters

A few special characters (or groups of characters) have special meanings when you include them in a User Command, or a Function Key. These characters can be used to make the User Commands even more convenient for you.

Character(s) Meaning

| The "pipe" character can be used to add a carriage return to the end of a User Command, as if you were to press <Enter> each time you used it.

For example, the User Command

@XQT PROGRAM|

would result in the command "@XQT PROGRAM" being printed on the screen, at the current cursor location, followed by an automatic carriage return. Note that in this case, it is assumed that you already have an SOE character on the screen next to the cursor.

<CR> Same as | (above)

<SOE> Places an SOE character within the User Command, so that the User Command can be used anywhere on the screen, even without an SOE.

For example, using the User Command (or Function Key)

<SOE>@@X|

would result in an SOE being printed on the screen, followed by the command "@@X", followed by an automatic carriage return. Note that this User Command can be used anywhere on the screen, (provided you are not in the middle of receiving something from the host).

~ Forces a blank space into the command, usually used at the tail of the command. (Without this, blanks will not be stored with the command.)

< > Same as ~ (above). Note that only a single space is valid between the brackets, although the three character combination of < > (or the single character ~) can used any desired number of times in a row to achieve multiple spaces, (e.g. ~~~~~ or < >< >< >< >< > for five spaces).

5.4.4 Using the User Commands

Once you have entered, and edited if necessary, one or more User Commands, you are ready to use them! To select a User Command from the list and use it, follow these steps:

1) Press <Alt><U> to activate the User Command pull-down menu.

2) Use the <UpArrow>, <DownArrow>, <Home>, <End> keys, the mouse, or press the first letter of the command name to select the User Command entry to be used. You should use the names that appear in the left column to help you recognize the desired command to be used.

3) Press <Enter> (or the right mouse button) to use a command. If the command does not contain a "pipe" (|) character, then the command will be retrieved, and printed wherever the cursor was prior to activating the pull-down menu. If the command does contain the "pipe" character, then a carriage return will automatically be sent for you, as if you had pressed <Enter>!

User Commands can be retrieved and used anywhere on the screen. In fact, they can even be used while in the RunLog screen. Be sure to look closely at a User Command before selecting it, so that you don't accidentally use one you don't want!

5.5.0 The Function Key Pull-Down Menu

Versicom allows you to assign up to ten commonly used commands to the ten available function keys on your computer. These assignments, like the User Commands, can be saved in a Setup file. Function key assignments can either be direct commands to the host computer, or they may start a Versicom macro, (e.g. *@@Do MyMacro*).

Having assigned any function keys, (see 5.5.1 Adding Function Key Assignments below), you can access the Function Key menu to view a list of the commands which are assigned to the function keys. In addition, you can change or edit assignments made previously, (see 5.5.2 Editing Function Key Assignments below).

5.5.1 Adding Function Key Assignments

To assign a command to a function key, follow these steps:

1) Press <Alt><F> to activate the Function Key pull-down menu.

2) Use the <UpArrow>, <DownArrow>, <Home>, <End> keys, the mouse, or press a number 1 through 0 (for keys 1 through 10) to select the Function Key entry to be used.

3) Press the <Ins> key to activate the Function Key entry/edit window.

4) Type the command to be assigned to this function key (up to 78 characters).

5) Press <Enter> to accept the command, (the entry/edit window will disappear).

6) Repeat steps 1-5 for more Function Key assignments, or press <Esc> (or the left & right mouse buttons simultaneously) to remove the Function Key pull-down menu.

If you make any mistakes while entering either the command, you can edit it by following almost the same process, (see 5.5.2 Editing Function Key Assignments below).

5.5.2 Editing/Removing Function Key Assignments

If you need to change (or remove) the command assigned previously to a function key, you can do so by following these steps:

1) Press <Alt><F> to activate the Function Key pull-down menu.

2) Use the <UpArrow>, <DownArrow>, <Home>, <End> keys, the mouse, or press a number 1 through 0 (for keys 1 through 10) to select the Function Key entry to be edited.

3) Press the <Ins> key to activate the Function Key entry/edit window.

4) *To edit the command...*

   Use the <Home>, <End>, <LeftArrow>, <RightArrow>, <Ctrl><RightArrow> or <Ctrl><LeftArrow> keys to position the cursor where the correction is to be made. Once the cursor is in the desired position, you can type over the old command, or use the <Ins> and <Del> keys to add or remove characters. Notice that when you move the cursor, or when you type new characters, the background color changes to accent the fact that a change is being made.

To remove (erase) the command...

Continue to press the <Space> key until the command has been erased.

7) Press <Enter> to accept the command, (the entry/edit window will disappear).

8) Press <Esc> (or the left & right mouse buttons simultaneously) to remove the Function Key pull-down menu.

Remember, if you want to save any changes made to function key assignments, you must save the modified setup, (see 5.7.0 The Setup Pull-Down Menu below). Saving the Setup will ensure that the next time you use it, the changes you made will again be there.

5.5.3 Special Function Key Assignment Characters

A few specific characters, or groups of characters have special meanings when you include them in a Function Key. These characters can be used to make the Function Keys even more convenient for you.

Any special characters allowed in User Commands can also be used in the Function Key commands, (see 5.4.3 Special User Command Characters above). However, you should use the "pipe" (|) character or the <CR> characters with care, especially in Function Key assignments. It is, of course, possible to press an undesired function key, at an inappropriate time, resulting in undesirable actions. Take heed!

5.5.4 Using the Function Keys

Once you have entered, and edited if necessary, one or more function key assignments, you are of course ready to use them! You have two choices in using a function key assignment; either press the key directly, or choose it from the pull-down menu.

For quick function key usage, follow these steps:

1) Make sure the text cursor is positioned correctly for the function key you are about to use, (e.g. make sure there is an SOE if one is needed).

2) Press the desired function key, (the command will appear at the location of the text cursor).

Remember, if the function key assignment contains any of the special characters mentioned above, (see 5.5.3 Special Function Key Assignment Characters above), you will see the results of those characters immediately! (e.g. an SOE, a carriage return, etc....)

To choose a function key from the pull-down menu, follow these steps:

1) Press <Alt><F> to activate the Function Key pull-down menu.

2) Use the <UpArrow>, <DownArrow>, <Home>, <End> keys, the mouse, or press a number 1 through 0 (for keys 1 through 10) to select the Function Key entry to be used.

3) Press <Enter> (or the right mouse button) to use a function key command. If the command does not contain a "pipe" (|) character, then the command will be retrieved, and printed wherever the cursor was prior to activating the pull-down menu. If the command does contain the "pipe" character, then a carriage return will automatically be sent for you, as if you had pressed <Enter>!

As is the case with User Commands, the Function Key assignments can be retrieved and used anywhere on the screen. In fact, they can also be used while in the RunLog screen. As is also the case with User Commands, be sure to exercise caution when using the function keys, so that you don't accidentally use one you don't want!

5.6.0 The History Pull-Down Menu

Every time you type a command after an SOE, and then press <Enter>, that command is sent to the host computer. In addition, Versicom places the command in what we call a History list. This list remembers the last twenty commands (up to 78 characters in length) that were sent to the host computer.

At any time, you can use this list to repeat one of the last twenty commands you used. In addition, you can modify a command before re-sending it, saving you valuable typing time.

In the event that you continue to consecutively repeat the same command, for example *P 20*, *L 20*, etc..., Versicom will not repeat that command in the history (it will only appear once). This will prevent you from finding that your History list is filled with the same command, twenty times.

5.6.1 Using the Command History

To choose a command from the History pull-down menu, follow these steps:

1) Press <Alt><H> to activate the History pull-down menu.

2) Use the <UpArrow>, <DownArrow>, <Home>, <End> keys, the mouse, or press the key of the first character of the command to be used.

3) Press <Enter> (or the right mouse button) to "grab" a command.

4) Press <Enter> again to use the command as it appears, or edit the command (since it is on the screen in front of you) and then press <Enter>.

As is the case with User Commands, and function keys, the History commands can be retrieved and used anywhere on the screen. In fact, they can also be used while in the RunLog screen.

5.7.0 The Setup Pull-Down Menu

Versicom provides the opportunity for you to create and use an almost unlimited number of Setup files. Each Setup file contains a set of twenty User Commands, and ten Function Key assignments, (see also 6.1.0 Setup Files).

By using the Setup pull-down menu, you can save and retrieve setups which are tailored to many different applications or different jobs you may have. For example, let's suppose that you are responsible for maintaining two different programs on your mainframe; SIMULATOR and COMPARER.

For each program, you will probably want to have a unique Setup file. Each file would have User Commands and function keys which have the exact program names and commands used for it's appropriate program.

In this example, you could create some User Commands, (see 5.4.1 Adding User Commands), and assign some special commands to function keys, (see 5.5.1 Adding Function Key Assignments), which you be useful when working on the SIMULATOR program.

Then, by using the SAVE option from the Setup pull-down menu, you could save these User Commands and function keys in a setup file called SIMULATR, (a maximum of eight characters for the name). You would then have a file called SIMULATR.VER saved in you Versicom subdirectory, (see also 6.1.0 Setup Files).

5.7.1 Save a Setup

After creating or editing User Commands or Function Keys, you can save these assignments under a unique Setup file. To save a Setup, follow these steps:

1) From the Terminal screen, press <Alt><S> to activate the Setup pull-down menu.

2) Press the <DownArrow> key once so that SAVE is highlighted, (or you could also press the <S> key or use the mouse to select the SAVE option).

3) Press <Enter> to open the SAVE filename entry/edit window, (the window will open with the default name for the setup, if one exists).

4) If a default name is shown, and you wish to use it, simply press <Enter> to accept that name.

   If there is no default name shown, or if you wish to use a new name, simply edit the default (if it shows) or type a new name, and press <Enter> to accept that name.

   In either case, if a Setup file already exists with that name, then Versicom will ask you if it is alright to overwrite (replace) the existing file.

5.7.2 Load a Setup

Once you have added some User Commands, [or] assigned some function keys, and saved those commands in a setup file, (see 5.7.1 Save a Setup), you can retrieve (load) that setup file, retrieving both User Commands and Function Key assignments. To load a Setup, follow these steps:

1) From the Terminal screen, press <Alt><S> to activate the Setup pull-down menu.

2) Press <Enter> to open the LOAD filename entry/edit window, (the window will open with the default name for the setup, if one exists).

4) If a default name is shown, and you wish to use it, simply press <Enter> to accept that name.

   If there is no default name shown, or if you wish to use a new name, simply edit the default (if it shows) or type a new name, and press <Enter> to accept that name.

In either case, if User Commands and (or) function keys are already assigned when you load a setup, they will be replaced by the new setup.

5.8.0 The RunLog Screen

The RunLog screen allows you to view and "reuse" text that has previously "scrolled" off the top of the terminal screen.

The RunLog screen is activated (and deactivated) by pressing the <Alt> key and the <R> key at the same time from the Terminal Screen. To de-activate the RunLog Screen, again press the <Alt> key and the <R> key at the same time.

When the RunLog is activated, the lower-right hand corner of the RunLog screen shows a percent level representing how far "into" the RunLog you currently are.

For example, when you first view the RunLog screen, the percent should be very small, (e.g. 0% or 1%), because you are viewing the most recent data to scroll off of the top of the terminal screen. In other words, you are not very far "into" the RunLog.

Also, if while in the RunLog you continue to press either the <UpArrow> or the <PgUp> key, the percent value will eventually read 100%. This shows that you have reached the oldest data to scroll off of the top of the terminal screen. In other words, you are all the way "into" the RunLog.

When the RunLog is active, you have full use of the same editing keys available at the terminal screen. In addition, a few special keys have been added to move you through the RunLog, from the newest, to the oldest screen data.

The size of the RunLog depends on the amount memory available on your computer. Versicom will allocate as much memory as is possible for the RunLog, while still leaving approximately 30,000 bytes free. If this is not possible, no RunLog will be created.
Finally, note that the RunLog is "circular", so that when it is full, it will "wrap around" to the beginning again, and the oldest screen data will "written over".

The Menu Bar and the Instruction Bar are the same as those in the Terminal Screen (described above).

5.8.1 Executing a Command From the RunLog

In addition to being able to review the RunLog for previous communications interactions, you may also "grab" text anywhere in the RunLog, edit it if you wish, and re-send it to the host.

For example, you might review the RunLog, finding a command (which is not in the History) that is similar to the one you would now like to send. In this case, you would follow these steps:

1) Place an SOE character on the screen at the start of the desired text, using the <Insert> key if needed.

2) If needed, edit the text using the standard Versicom cursor movement keys.

3) Finally, place the cursor either under the last character of the command or in a blank space following the command, and press <Enter>.

When you press <Enter> Versicom will "grab" the text you have specified, return you to the terminal screen (leave the RunLog), and send the text to the host. It is in this fashion that you can *construct* a command to be sent from the RunLog.

5.8.2 RunLog Keys

In addition to the keys available at the terminal screen, the following keys are available at the RunLog screen:

While at the terminal screen...
<Alt><R> Displays the RunLog screen (activate the RunLog)

While in the RunLog...
<Alt><R> Returns to the terminal screen (deactivate the RunLog)
<PgUp> Scroll back one screen into the RunLog (older text)
<PgDn> Scroll forward one screen into the RunLog (newer text)

In addition, you can use the mouse to maneuver through the Run Log. Simply press the left mouse button down, and while still holding it down drag the mouse cursor either up into the top of the screen (to go backwards) or into the bottom (to move forwards).

5.9.0 The Capture Pull-Down Menu

Versicom allows you to "capture" text going to and from the host. This feature allows you to keep a record of transactions on the printer (LPT1:), or in an ASCII log file (filename.LOG). The file capture option can also be used for simple (ASCII) file transfer.

The Capture menu is activated by pressing the <Alt> and the <C> keys at the same time. The by using the <Arrow> keys, select either Printer or Disk and press <Enter>.

5.9.1 Printer Capture

To turn on the printer capture feature, follow these steps:

1) From either the terminal or the RunLog screen, press <Alt><C>
2) Press <Enter>

Notice that when the menu appears, the first choice now appears as PRINTER ON, emphasizing the fact that the printer capture option is currently *off*.

Also note that the word "PRINTER" now appears at the lower-right hand of the screen, to remind you that the printer capture is on.

To turn off the printer capture, follow the same steps. Notice however that this time the menu choice which was previously PRINTER ON is now PRINTER OFF, emphasizing the fact that the printer capture option is currently *on*. The word PRINTER should now be absent from the lower-right hand of the screen.

5.9.2 Disk Capture

To turn on the file capture feature, follow these steps:

1) From either the terminal or the RunLog screen, press <Alt><C>
2) Press the <DownArrow> key once, (or you could also press the <D> key or use the mouse to select the Disk option).
3) Press <Enter>
4) Type the name of the file to use in the disk file capture

Note that the DOS file extension ".LOG" will automatically be appended to the name you supply. Any other supplied extension will be ignored.

Notice that the word "DISK" now appears in the lower-right hand corner of the screen, to remind you that the disk capture is on. The disk file will be placed in the current directory (usually \Versicom) but can be moved, renamed, and (or) edited later.

To turn off the disk capture, follow the same steps. Note that this time the DISK ON option was instead DISK OFF, and was already highlighted when the menu appeared. The word "DISK" should now be absent from the lower-right hand of the screen.

5.10.0 The DOS Shell Option

Another versatile feature of Versicom is that it allows you to leave the program (go to

DOS) momentarily, and then return without losing the contents of your current screen, the RunLog, or your History.

You must remember however that Versicom allots as much memory as is "feasible" for the RunLog when you start the program. Therefore, the type of work you are able to do when you run a DOS shell may be limited by the amount of memory being "hogged" by Versicom.

Generally, if there is not enough memory available to do what you want in the DOS shell, then you should consider the option to quit Versicom without terminating the connection, (see 5.11.0 The Quit Option). However, because the *quit option* actually terminates the program, both the command History list and the RunLog data will be lost.

To use the DOS shell option, follow these steps:

1) Press <Alt><D> from the Terminal Screen to activate the DOS shell option window.

2) Press the <Y> key (or use the mouse to click on the "Y" in the "(Y/N)?" portion of the prompt) to escape to a DOS shell.

3) Run any desired programs (execute any desired commands) as if you were at a normal DOS prompt.

   Remember, if you experience an error message such as "Not enough memory", consider quitting Versicom without terminating the connection, (see 5.11.0 The Quit Option).

4) When you are done using the DOS shell, and you wish to return to Versicom, type the command "EXIT" at the DOS prompt, and you will be returned to Versicom, as if you had never left.

When you use the DOS shell option, it is very important to remember that Versicom is still running in the background, and it is possible to terminate the connection to the host computer, or otherwise alter the communication setup by using another communications program from the DOS shell. In other words, use care.

5.11.0 The Quit Option

As much as you may enjoy using Versicom, there will eventually come the time (for most of us) when you must quit, and hopefully go home for the night! The Quit option on the Terminal Screen allows you to terminate the program, and hang up the telephone line (disconnect) if you haven't already done so.

Normally, you should tell the host computer to terminate the connection, as opposed to you forcing that action on the host. The reason for this is that some host computers may require a "clean" termination, so that any special port settings will not be "inherited" by future users of the same port. Check your host computer's manual for the correct command (e.g. @@*TERM*) to terminate the connection.

5.11.1 Quitting and Terminating the Connection

Normally, when you quit Versicom, you are truly done, and will not wish to return within the same computer session. In this case, you will want to "terminate" the connection to the host computer, leaving it's communications port free for someone else to use.

To quit Versicom and terminate the connection, follow these steps:

1) Press <Alt><Q> from the Terminal Screen, to activate the Quit option window.

2) Press the <Y> key (or use the mouse to click on the "Y" in the "(Y/N)?" portion of the prompt) to indicate that you do wish to quit Versicom.

3) If you have already terminated the connection as described above, you will immediately leave Versicom, with no further intervention. This would be considered the normal method of quitting Versicom. Proceed to step 4 below.

   If you have not already terminated the connection, Versicom will ask you to indicate whether or not you do wish to "force" the termination of the connection.

   Press the <Y> key (or use the mouse to click on the "Y" in the "(Y/N)?" portion of the prompt) to indicate that you do wish to terminate the connection.

4) You are done using Versicom!

5.11.2 Quitting and Leaving the Connection Active

There are some cases where you may want to leave Versicom, without "terminating" the connection to the host computer. Such a case might be where there is not enough free memory to run another program using the DOS shell option. In this case, you would want to quit without terminating the connection, and then return shortly to pick up where you left off.

To quit Versicom and leave the connection active, follow these steps:

1) Press <Alt><Q> from the Terminal Screen, to activate the Quit option window.

2) Press the <Y> key (or use the mouse to click on the "Y" in the "(Y/N)?" portion of the prompt) to indicate that you do wish to quit Versicom.

3) Because you have not already terminated the connection, Versicom will verify that you do not wish to "force" the termination of the connection by prompting you for an indication of whether or not you want to terminate it when you quit.

   Press the <N> key (or use the mouse to click on the "N" in the "(Y/N)?" portion of the prompt) to indicate that you do not wish to terminate the connection.

4) Run any desired programs (execute any desired commands) as if you were at a normal DOS prompt, because you are!

5) When you are ready to re-start Versicom, simply start the program as you would normally, (see **3.0.0 Starting Versicom**), but this time, do not include (as command line parameters) any macros which would dial the telephone number, (such as the included sample 1100A.MAC). Remember, you are still connected to the host, you don't need to redial!

6.0.0 Special Files

When using Versicom, there are several files you may wish to become familiar with. The following files are used by Versicom to maintain function key assignments & User Commands, macros, and system installation information.

6.1.0 Setup Files

The Versicom Setup files contain information for two of the Versicom pull-down menus: User Commands, and Function Keys. The Setup files can not be "looked at" from DOS, because they are not ASCII files. The names of Setup files are constrained by the normal DOS file naming conventions; eight characters, etc....

The setup files can be loaded by one of two methods:

1) The name of the setup file can be supplied as the first parameter when executing Versicom from DOS. (e.g. *Versicom MYSETUP*)

2) Once at the main terminal screen of Versicom, you can press <Alt><S> to activate the Setup menu, and choose to load a setup.

To create or update a Setup file, you must change the User Commands or Function Keys from the Terminal Screen, and then save the setup via the Setup pulldown menu, (see 5.4.0 The User Command Pull-Down Menu, 5.5.0 The Function Key Pull-Down Menu and 5.7.1 Save a Setup).

Setup files are not required to use Versicom, but you must define an SOE key, (most likely a Function Key), to use Versicom, (see **4.0.0 Understanding the SOE**).

6.2.0 Macro Files

The Versicom macro files contain a sequence of macro commands to be executed, one at a time. These macro files (macros) can be used to automate a logon, or other such repetitive tasks.

The Macro files are "pure" ASCII files, usually created with a simple text editor, or with the "non-document" mode of a word processor. The names of Macro files are constrained by the normal DOS file naming conventions; eight characters, etc....

Commands must appear only one per line, and each line may contain an optional semicolon (";"), followed by a comment describing that command.

The total length of a command and its comment should be less than 80 characters to prevent confusion.

Macro files are not required to use Versicom.

6.3.0 Installation Files

When Versicom is installed for the first time, it will create a special file called Versicom.INS, which is the default Installation file. This file must always be present to run Versicom. However, once Versicom is running, alternate Installation files may be loaded by running a macro which contains the INSTALL command, with the new installation filename.

The Installation files can not be "looked at" from DOS, because they are not ASCII files. The names of Installation files are constrained by the normal DOS file naming conventions; eight characters, etc....

The Installation files contain the following:

-Communications port number to be used (COM1, COM2, etc...)
-Base hardware address for the port/modem (02F8 Hex, 03F8 Hex, etc...)
-Hardware interrupt request number for the modem (IRQ3, IRQ4, etc...)
-Baud rate (speed) for transmission (1200 baud, 9600 baud, etc...)
-Stop bits to be used during transmission (1 bit or 2 bit)
-Parity to be used during transmission (Even, Odd, etc...)
-Data bits per character (7 bits or 8 bits)
-Prompt string to be used as a default (">", ":", etc...)
-RunLog colors (Blue on Gray, Red on Gray, etc...)
-Video display mode (Color 50 line VGA, Color 25 line CGA, etc...)

More than one installation file can be created, allowing you to access different communication ports, or use different video display modes. For example, you might create the following installation files which are identical except for the video display mode:

Versicom.INS   The default installation, with the color 25 line CGA video mode.
MY-VGA.INS   An alternate installation, with the color 50 line VGA video mode.

Therefore, normally when you begin Versicom, you will have the color 25 line CGA video mode, because that was the mode selected in the default Installation file Versicom.INS.

However, you could also create a macro, possibly called MY-VGA.MAC, with the single following command:

Install "MY-VGA"   ; Load an installation with color VGA video mode

If you decide now that you would like the special 50 line video mode, you could simply run the MY-VGA macro either from the DOS command line, or by using the command *@@Do MY-VGA*.

This would run your MY-VGA macro, which loads your MY-VGA installation, which has the color 50 line VGA video mode. (See also 7.2.3 Install under 7.2.0 Macro Commands).

7.0.0 Macros

7.1.0 What is a Macro?

A Versicom macro is an ASCII text file which contains a set of instructions to be carried out automatically by Versicom, without necessary intervention from you.

Macros can be initiated by either of two methods:

1) The name of the macro (or macros) can be supplied as the second (third, fourth, through ninth) parameter when executing Versicom from DOS.

   For example, typing *Versicom MYSETUP MYMACRO* from DOS will begin Versicom, load the setup MYSETUP.VER, and execute the macro MYMACRO.MAC

2) Once at the main terminal screen of Versicom, you can type *@@Do* followed by a space and a macro name, after any SOE character.

   For example, typing *@@Do MYMACRO* and then pressing the <Return> key, from the terminal screen, will execute the macro MYMACRO.MAC

If you are initiating macros from DOS, when starting Versicom, you may include up to nine macros names, separated by spaces. You need not supply the ".MAC" extension for the macros.

NOTE:

> If you wish to start Versicom and supply a macro name as a parameter, then you must also include a Setup file name. For instance, if you attempt to run Versicom by typing *Versicom MYMACRO*, then MYMACRO will be used as the Setup file name, because it is the second parameter (in this case). You may wish to fool Versicom by supplying a Setup file name which doesn't exist, followed by the macro name.

7.2.0 Macro Commands

All of the following macro commands can be included in macro files to create the desired list of commands to be executed. These commands can be typed in either upper or lower case letters, or a combination of the two. However, many command parameters which are included between quote signs, must match a specific upper or lower case, depending on the host computer you are connecting to.

7.2.1 WaitFor

Syntax: WAITFOR "TEXT"

The WAITFOR command can be placed in a macro to suspend execution, until a specific string of characters ("Text" above) is received from the host computer.

Example:

```
...
WaitFor "User ID:"
...
```

In this example, placing the command *WaitFor "User ID:"* in the macro will cause that macro to pause during execution, waiting for the string "User ID:" to be received. Once the string is received, the macro will continue. Remember that what is in between the quote signs is literal, and case sensitive.

As a startup default, Versicom will wait indefinitely for the "Text" to be received. If, however, you wish to limit the amount of time spent waiting for the "Text", you can change it by using the TIMEOUT macro command, (see 7.2.11 Timeout).

7.2.2 Send

Syntax: SEND "TEXT"

The SEND command can be placed in a macro to automatically send (transmit) a string of characters to the host computer.

Example:

```
...
Send "My Login String|"
...
```

In this example, placing the command *Send "My Login String|"* in the macro will cause that macro to send the string "My Login String" to the host computer, followed immediately by a carriage return, (notice the pipe character at the end of the string). Remember that what is in between the quote signs is literal, and case sensitive. (See also 7.4.0 Special Macro Characters).

7.2.3 Install

Syntax: INSTALL "FILENAME"

The INSTALL command can be placed in a macro to automatically load a new installation file.

Example:

```
...
Install "VGA"
...
```

In this example, you could use the included program Ver_Inst.EXE to create an installation file called "VGA.INS", which is the same as your default installation file, except that it has the VGA 50 line color mode selected. Then, by placing the command *Install "VGA"* in the macro you will cause that macro to load the new installation "VGA" which would then switch you to the VGA 50 line color video mode.

7.2.4 Hide-Keys

Syntax: HIDE-KEYS

The HIDE-KEYS command can be placed in a macro to instruct Versicom to hide any typing on the keyboard, until the SHOW-KEYS command is used.

Example:

```
...
WaitFor "Password: "
HideKeys
WaitFor "Password accepted."
Show-Keys
...
```

In this example, we show the instance where you may wish to hide your typing when the host computer asks you for your password. By placing this combination of commands in the macro, during execution the sample macro will begin to hide your typing as soon as the host sends the string "Password: ", and will show your typing once again when the correct password is entered.

7.2.5 Show-Keys

Syntax: SHOW-KEYS

The SHOW-KEYS command is used in a macro to counteract the HIDE-KEYS effect. This command must always be used after the HIDE-KEYS command is used, otherwise you will no longer be able to see what you are typing!

Example: (See 7.2.4 Hide-Keys for an example of the SHOW-KEYS command.)

7.2.6 Delay

Syntax: DELAY "MILLISECONDS"

The DELAY command can be placed in a macro to pause the macro execution for a predetermined amount of time. This time can be as short as one millisecond (1/1000th of a second), or as long as 60 seconds.

Example:

```
...
Delay "5000"
...
```

In this example, by placing the command *Delay "5000"* in the macro, you will cause the macro to pause for exactly five seconds, and then resume operation. During this pause, Versicom will continue to "watch" for incoming characters from the host computer. The parameter "milliseconds" can be determined by multiplying the number of seconds to delay by 1000. In other words, milliseconds = seconds x 1000.

7.2.7 Quit

Syntax: QUIT

The QUIT command can be placed in a macro to terminate the connection and end a Versicom session, without any warnings or messages.

Example:

```
...
Send "logoff|"
WaitFor "Session complete."
Quit
```

In this example, placing the command *Quit* in the macro would end the Versicom session after the string "logout" (and a carriage return) was sent to the host, and the host had responded with "Session complete.".

Remember, the QUIT command will terminate the connection if it is still active, and you will not see any messages telling you that this is happening. The idea is to provide a "quick and dirty" method of ending a session, so use this command with care!

7.2.8 Prompt

Syntax: PROMPT

The PROMPT command can be placed in a macro to change the expected host computer prompt, while the macro is executing. This has the same effect as typing the special @@PROMPT command from the terminal screen.

Example:

```
...
Prompt "NewPrompt:"
Send "Connect newhost.host.myname|"
WaitFor "<PROMPT>"
...
```

In this example, placing the command *Prompt "NewPrompt"* in the macro, causes Versicom to replace its default host prompt with the new one you have specified, ("NewPrompt" in this case). Because this "NewPrompt" is now the default in Versicom, any time the string "NewPrompt" is received from the host computer, Versicom will automatically print an SOE character on the screen. (See also 8.2.0 @@Prompt.)

7.2.9 Repeat

Syntax: REPEAT

The REPEAT command can be placed in a macro to create a loop which will repeat any following commands, up to the point where an UNTIL command is encountered. When the UNTIL command is encountered, Versicom will attempt to determine whether or not the condition specified by the UNTIL command has been met (logically true).

Example:

```
...
Repeat
        Send "Is anyone listening?|"
        WaitFor "I'm listening"
Until Receive "I'm listening"
...
```

In this example, you see how a REPEAT UNTIL loop could be used to continually send a message to the host until someone responds by sending you the string "I'm listening".

Remember that the REPEAT must eventually be followed by an UNTIL command with a Logical Command, or you will receive an error message from Versicom. (See also 7.2.10 Until and 7.3.0 Logical Commands).

7.2.10 Until

Syntax: UNTIL LOGICAL-COMMAND "PARAMETER"

The UNTIL command is used to terminate (complete) a REPEAT UNTIL loop. You should remember that Versicom's loop structure is designed to repeat some commands until some condition occurs.

Example: (See 7.2.9 Repeat for an example of the UNTIL command.)

In this example, the command *Until Receive "I'm listening"* was placed in the macro, following the REPEAT command, in order to terminate (complete) the loop. In this case, the Logical Command is the RECEIVE command, and the parameter to the RECEIVE command is the string "I'm listening".

7.2.11 Timeout

Syntax: TIMEOUT "MILLISECONDS"

The TIMEOUT command can be used to change the maximum amount of time Versicom will wait for a string with the WAITFOR command. In other words, you can specify how much time will be allowed to "run out" while waiting.

Example:

```
...
Timeout "15500"
WaitFor "I'm listening"
...
```

In this example, by placing the command *Timeout "15500"* in the macro, we will tell Versicom that from now on, we will wait no more than fifteen and one half seconds in the WAITFOR command. Versicom's default TIMEOUT value is zero (0) which means "wait forever". You can change the value back and forth, as much as you like. Use the TIMEOUT command *Timeout "0"* to restore the Versicom default time of "forever".

7.2.12 Beep

Syntax: BEEP

The BEEP command can be placed in a macro to generate a short tone with the computer's sound generator.

Example:

```
...
WaitFor "Password: "
Beep
...
```

In this example, by placing the BEEP command in the macro, we will tell Versicom to generate a short tone after the host computer sends the string "Password: ". In other words, this example will notify you when the host computer is ready to receive your password.

7.2.13 Run

Syntax: RUN "PROGRAM"

The RUN command can be used to run another program or DOS command from within a Versicom macro.

Example:

```
...
Run "dir c:\versicom\*.mac"
...
```

In this example, placing the RUN command in the macro will make Versicom temporarily leave to DOS, and effectively type the command "dir c:\versicom\*.mac". Here, you would execute this macro every time you wish to get a listing of all of the available macros, (remember that macro files have the extension .MAC).

7.2.14 Upcase

Syntax: UPCASE "ON" or UPCASE "OFF"

The UPCASE command can be used to allow Versicom to use lower case characters, even though the default state is all upper case characters.

Example:

```
...
UpCase "OFF"
...
```

In this example, placing the command *UpCase "OFF"* in the macro will change the Versicom default to allow lower case characters. This command only takes effect when it is encountered in the macro, as commands are sequentially executed.

Remember, many computers decide whether or not to use lower case characters, as soon as they receive the first few characters from you. Therefore, if you want to communicate with a host computer in lower case mode, it is a good idea to include this command at the start of the "dial-up" macro.

7.3.0 Logical Commands in a Macro

The logical commands are designed to be used with the Versicom macro command UNTIL which requires an indicator of true or false, determining whether the "something" in "until something is true" actually occurs. Every UNTIL command must be used with one of the logical commands listed below.

7.3.1 Receive

Syntax: UNTIL RECEIVE "TEXT"

The RECEIVE command can be used in conjunction with the UNTIL command to end a REPEAT loop. With the required WAITFOR command placed immediately prior to the UNTIL command, the loop will repeat until the "text" specified by the RECEIVE command is received and acknowledged by the preceding WAITFOR command with the same "text" as its parameter.

Example:

```
...
Timeout "10000"
Repeat
        Send "<CR>"
        WaitFor "Password: "
Until Receive "Password: "
...
```

In this example, by including the *Until Receive "Password: "* command immediately after the *WaitFor "Password: "* command, the macro will send a carriage return, (the *Send "<CR>"* command), until the string "Password: " is received from the host computer.

Make note of the use of the TIMEOUT command prior to the start of the loop. This actually is a necessity to the correct execution of the loop. It should be clear that with the default timeout value of zero (wait "forever"), the macro would not loop. Instead it would pause at the WAITFOR command, waiting until the string "Password: " was received, at which time the UNTIL command would be satisfied, and the loop would be terminated. Here, the contents of the loop would only be executed once.

Remember, the UNTIL RECEIVE command combination must always be preceded by a WAITFOR command in order to "trap" the desired string of characters.

7.3.2 Quiet

Syntax: UNTIL QUIET "MILLISECONDS"

The QUIET command can be used in conjunction with the UNTIL command to end a REPEAT loop. It will allow the loop to repeat until the line has been "quiet" for the specified amount of time.

Example:

```
...
Repeat
    ...
Until Quiet "10000"
...
```

In this example, by placing the *Until Quiet "10000"* command in the macro, the execution of the loop will continue until the line has been quiet (no characters are received) for 10000 milliseconds or 10 seconds.

7.4.0 Special Macro Characters

A few special characters, or groups of characters have special meanings when you include them in the text string of a macro command. These characters can be used to make the macro commands even more effective for you.

Character(s) Meaning

| The "pipe" character can be used to add a carriage return to the end of a SEND "TEXT" command, as if you were to press <Enter> each time it was used.

For example, using the command

Send "MyUserID|"

would result in the text "MyUserID" being sent to the host computer, followed by an automatic carriage return. Note that in this case, it is not necessary to have an SOE character on the screen next to the cursor.

<CR> Same as | (above)

<SOE> Places an SOE character within the text of a SEND command, so that the command.

For example, using the send command

Send "<SOE>"

would result in an SOE being printed on the screen.

~ Forces a blank space into the command, usually used at the tail of the command. (Without this, blanks may not be sent with the command.)

< > Same as ~ (above). Note that only a single space is valid between the brackets, although the three character combination of < > (or the single character ~) can used any desired number of times in a row to achieve multiple spaces, (e.g. ~~~~~ or < >< >< >< >< > for five spaces).

7.5.0 A Sample Macro

The following few lines will demonstrate a macro used to dial a telephone number, wait for a connection, and send a login string of characters. This file can be created with any ASCII text editor. Note that commands can be in either upper or lower case.

```
                         ; Begin the sample macro
                         ;
Timeout "15000"          ; Change maximum time of WaitFor, to 15 seconds
                         ; (Timeout "0" sets WaitFor time to forever)
                         ;
Repeat
                         ; Repeat the following commands
  Send "ATZ|"            ; Send the characters "ATZ", plus <Enter>, to the modem
  Delay "500"            ; Pause for 500 milliseconds (1/2 second)
  WaitFor "OK"           ; Wait for the modem to send back the characters "OK"
  Send "ATDTxxx|"        ; Instruct the modem to dial the number xxx
  WaitFor "CONNECT"      ; Wait for the modem to tell you that it has a connection
                         ;
                         ; Continue repeating the last five instructions until
                         ; the last WaitFor command was satisfied
                         ;
                         ; If the string "CONNECT" is not received before the
                         ; Timeout time (15 seconds here), then we didn't
                         ; Receive "CONNECT", so the next statement will
                         ; jump back to the Repeat
Until Receive "CONNECT"
```

```
                        ;
WaitFor "LOGIN:"        ; Wait until host computer asks for the login string
                        ;
Send "MyLogin"          ; Send the characters "MyLogin" to the host
                        ; Note that anything surrounded by quotes, retains its case
                        ; (e.g. upper and lower case are recognized)
```

8.0.0 The Special @@ Commands

The @@ commands are special commands used either to affect a Versicom system parameter, or to execute Versicom command macros.

Each @@ command can be typed (used) from anywhere on the screen, as long as you assure that there is an SOE character preceding the @@, as is the case with any command to be sent to the host.

The main difference between normal commands and the @@ commands is that normal commands are used to inform the host computer of any desired actions, where the special @@ commands are used to inform Versicom of a desired action.

Normally, commands are typed at an SOE, and when the <Enter> key is pressed, that command is sent to the host computer. However, if Versicom detects that you have typed any of the special @@ commands, it reads your instruction, but does not pass it on to the host computer. Therefore, the special @@ commands will not normally affect the host computer, unless of course if you activate a macro which sends commands to the host computer.

8.1.0 @@Do

The @@Do command is used to start the execution of a Versicom macro, (see **7.0.0 Macros**). The command syntax (format) is as follows:

```
@@Do filename
```

The "filename" must be a valid Versicom macro file (ASCII text file) which resides in the current directory (usually Versicom). The extension ".MAC" will be used as the file extension, any other extension will be ignored.

Note that macros can also be started automatically when you begin Versicom by including the names at the DOS command line when you start Versicom. Remember that names at the command line will only be recognized as macros if they appear as the second (or more) parameter, (see 3.1.0 Starting Versicom With Command Line Parameters).

8.2.0 @@Prompt

The @@PROMPT command is used to change the default string which is recognized by Versicom as the host computer's command prompt. Such a change can conveniently allow Versicom to automatically print an SOE following the receipt of such a string. The @@PROMPT command syntax is as follows:

@@PROMPT "new-prompt-string"

For example, let us suppose that you normally connect to a host which sends the prompt string "system.username> " every time it is ready for a command. In this case, you would probably install this as the default prompt when you installed Versicom, (see 2.2.0 Modifying a Setup, step 15).

What if, however, you needed to temporarily connect to another host computer which uses a different prompt such as "newsystem.username: "? In this case you would connect to the new host by typing any necessary host commands, supplying any needed SOEs during the connection process. Once connected, you would type

@@PROMPT "newsystem.username: "

and press <Enter>. From that point on, Versicom would automatically provide an SOE each time it received the string of characters "newsystem.username: " from the host computer. Upon returning to the original host computer, you would use the same procedure to restore the Versicom prompt to the original string.

Remember that you can also automate the process of connecting to a second host computer by creating a Versicom macro such as the following example.

Create the ASCII text file NEWSYS.MAC with the following events/commands:

1) Send the needed commands (e.g. *SEND "connect newsystem"*)
2) Change the prompt to the password prompt (e.g. *PROMPT "Password: "*)
3) Wait for the new system response (e.g. *WAITFOR "Password: "*)
4) Wait while you type your password (e.g. *WAITFOR "Password accepted."*)
5) Change the prompt again (e.g. *PROMPT "newsystem.username: "*)

With this example, you could always connect to the new system simply by typing *@@Do NEWSYS*, which would connect you to the new host and change the prompt for you.

8.3.0 @@Upcase

The @@UPCASE command allows you to turn off or on Versicom's ability to force all characters sent or received to upper case. The @@UPCASE syntax is as follows:

@@UPCASE "OFF" (Versicom will no longer force letters to upper case)

@@UPCASE "ON" (Versicom will once again force letters to upper case)

The default UPCASE status (setting when Versicom is started) is ON. In other words, unless you instruct Versicom to the contrary, it will force all characters sent and received to their upper case equivalents. Therefore, action is only required when you wish to access a host computer which distinguishes between upper and lower case letters.

A good item to note is that most host computers will attempt to determine during the login process what type of remote "terminal" is connecting to them. Therefore if you are connecting from a host which only recognizes upper case characters to a host which only recognizes lower case characters, you should use the @@UPCASE command after you tell the original host to begin the connection, but before you type any of the login information for the new host (such as UserIDs or Passwords).

Issuing *UPCASE "OFF"* too early may result in the original host not recognizing your connection command. Issuing *UPCASE "OFF"* too late may result in the new host deciding that you are using a upper-case only terminal, and therefore will no longer use lower case.

Remember that the UPCASE command can also be included in a Versicom command macro which connects you to the host, so that you do not have to type the @@UPCASE command each time you connect to a "case-sensitive" host computer.

8.4.0 @@HideKeys

The @@HIDEKEYS command is used to instruct Versicom to replace any characters typed at the keyboard with a small box character. The idea is to prevent your typing from being viewed by others, innocently or not. HIDEKEYS can be deactivated by using the @@SHOWKEYS command seen below.

8.5.0 @@ShowKeys

The @@SHOWKEYS command is used to instruct Versicom to no longer replace characters typed at the keyboard with the small box characters. Although no harm can be done by using this command at any time, normally you would only use it after having used the

@@HIDEKEYS command. Remember, in this normal case, the keys will be hidden, so you will not be able to see yourself type @@*ShowKeys*. However, you should immediately be able to see whether or not you typed the command correctly!

9.0.0 Using a Mouse With Versicom

Most of the popular mouse pointing devices can be used with Versicom, provided they are installed, and the mouse driver is active, prior to running Versicom. For information on installing your mouse, you will have to read your mouse manual!

Once installed, the mouse can be used to click on any of the Versicom pull-down menus (or other options) on the terminal or RunLog screens. Once the menu appears, the mouse can be used to select or execute an item from the menu. In addition, you can "click" on the letter of your response to a yes/no question. For example, click on the Y to answer yes to the question, "Do you wish to quit, (Y/N)?".

The mouse cursor will appear as a small rectangular box, approximately the size of a single character, which moves about on the screen as the mouse is moved. On a color screen, the mouse cursor will be colored red (against the black background).

The mouse buttons function as follows:

PRESS (CLICK)	TO...
Left Button	Move cursor to current location of mouse cursor. Activate a pull-down menu. Highlight a pull-down menu option. Select a response to a Versicom window question. (e.g. click on the "Y" for a "Yes" response to the window which asks "Do you wish to quit, (Y/N)?")
Left Button twice quickly (double click)	Press the <Enter> key at the current position of the mouse cursor.
Right Button	Press the <Enter> key, regardless of the current position of the mouse cursor.
Simultaneous Left & Right Buttons	Press the <Esc> key, to escape from a pull-down menu or a Versicom window.

10.0.0 Errors and Other Messages

The following is a list of messages you may receive from Versicom. The messages have been divided into non-fatal and fatal categories.

Non-fatal messages, however serious they may be, will not cause Versicom to terminate (quit).

Fatal messages on the other hand, only occur when a vital rule has been violated, for one reason or another. Fatal messages will be printed just prior to system termination. Fatal errors must be corrected before Versicom can be run, so you would be well advised to contact the manufacturer.

10.1.0 Non-Fatal Messages

01: HideKeys is active, the <BackSpace> key is the only cursor movement allowed.

HideKeys is a command which is issued from an executing Versicom macro. Its purpose is to prevent any characters you type from being printed on the screen so that bystanders (innocent or otherwise) won't be tempted to look at sensitive information such as passwords. Because it has a limited purpose, there is limited cursor movement allowed while it is active. If you make a typing mistake while HIDEKEYS is active, simply use the <BackSpace> key to backup to the mistake. Keys such as <Home>, <End>, and <LeftArrow> (etc...) are not allowed until HIDEKEYS is again inactive, (see 8.5.0 @@Show-Keys).

02: Missing SOE, (input ignored).

Every command you type while using Versicom must be preceded by the SOE character. If the SOE character is not currently being provided automatically, you must manually provide it, (see **4.0.0 Understanding the SOE**).

03: No hidden text remaining to <BackSpace> over.

You have apparently tried to press the <BackSpace> key while HideKeys was active, even though there were no more characters to erase. If you wish to see the keys you are typing, use the SHOWKEYSmmand, (see 8.5.0 @@Show-Keys).

04: Connection is already active.

When you start Versicom, it will tell you if a connection through the current serial port is already active. In many cases, this may be completely understandable, such as if you previously chose to quit without terminating the connection. However, if this message was not expected, and incorrect to the best of your knowledge, you may have incorrectly specified the serial port settings when installing Versicom, (see 2.2.0 Modifying a Setup).

05: Versicom terminated with connection intact.

If you chose to quit Versicom while leaving the connection intact, Versicom will inform you that it is aware of this. This also serves as a quick reminder just in case your actions were unintentional.

06: Syntax error with RECEIVE command in macro.

This message occurs when you are executing a command macro, and Versicom finds the logical command RECEIVE,which doesn't follow the normal convention (syntax) expected by Versicom, (e.g. *Repeat ... Until Receive "Password: "*). It's possible that the command has a typographical error, or that it is being used in an incorrect manner, (see 7.3.1 Receive).

07: Syntax error with QUIET command in macro.

This message occurs when you are executing a command macro, and Versicom finds the logical command QUIET,which doesn't follow the normal convention (syntax) expected by Versicom, (e.g. *Repeat ... Until Quiet "1000"*). It's possible that the command has a typographical error, or that it is being used in an incorrect manner, (see 7.3.2 Quiet).

08: Prompt string too long, command ignored.

You have instructed Versicom to use a new prompt, (via the special @@PROMPTcommand or the PROMPTmacro command), which is longer than the maximum number of characters allowed for a prompt, (15 characters). Usually this can be fixed by using only the last 15 characters of the new prompt, which should provide you with an SOE at the appropriate time.

09: Missing quote sign (").

Versicom has encountered an instruction (from you) which is missing a second quote sign ("). Many macro commands and special @@ commands require that part of the command (an argument or parameter) be placed in between a pair quote signs. If a macro is running, check the macro to see that all commands which require quote signs, have a complete pair for each command.

10: Syntax error in macro file 'macroname' ('command').

This message is to notify you that when Versicom was reading the macro file 'macroname', it encountered a command ('command') which it didn't understand. Usually, this simply means that there is a typographical error with the command. To fix the problem, edit the macro file and correct the typo'. If the command in question looks correct to you, it is possible that you are not using the command correctly, or that the command just isn't known to Versicom, (see 7.2.0 Macro Commands).

11: Macro 'macroname' has no commands to execute.

Versicom attempted to begin executing a macro, but found no commands in the specified macro. Edit the file 'macroname.MAC' to correct the problem. Either add the correct commands, or correct any syntax which may have indirectly caused the error.

12: Error opening macro file 'filename'.

Versicom attempted to begin executing a macro, but was unable to open the specified file containing the macro commands. The reason for this error could be as simple as the problem of the file being nonexistent in the current directory, or as complicated as a disk problem with your computer.

13: Macro halted.

This message, which follows other error messages related to macros, is sent by Versicom because the preceding error (message) was such that the execution of the macro can't continue. For this reason, you must correct the preceding error(s) before the macro can be executed.

14: Invalid delay time Delay encountered in active macro.

Versicom allows you to specify a delay of up to 60 seconds during macro execution, by using the DELAYcommand. The specified delay time must be a number which is less than or equal to 60,000 milliseconds. For example, specify *DELAY "10000"* in the macro to halt execution for 10 seconds.

15: Install request ignored.

While reading the current macro, Versicom encountered an instruction to load a new installation, and the required installation filename was omitted. Edit the macro file and add the desired installation file name as the INSTALLcommand parameter, (see 6.3.0 Installation Files, and 7.2.3 Install).

16: Could not find the installation file 'filename'.

Versicom attempted to process the loading of a new installation file ('filename'), but was not able to find that file in the current directory. This could happen when Versicom is initially started if the default installation file 'Versicom.INS' doesn't exist in the current directory. It could also happen if the macro command INSTALL was encountered in a macro, and the specified installation file didn't exist. (See also 6.3.0 Installation Files, 2.2.0 Modifying a Setup - step 24, and 7.2.3 Install).

17: Could not find the setup file 'filename'.

Per your instruction via a given command line parameter when starting Versicom, it attempted to load the specified setup file 'filename' which contains certain User Command and Function Key assignments. However, Versicom couldn't find the specified setup in the current directory. It's possible that you have incorrectly placed the startup parameters, possibly including them in the incorrect order. It is after all also possible that you intentionally specified a setup file which doesn't exist, in which case you can disregard this message. (See also 6.1.0 Setup Files, 3.1.0 Command Line Parameters, and 5.7.2 Load a Setup).

18: Must supply a valid Setup file name.

An attempt was made to save a setup, (User Command and Function Key assignments), but a name for the setup was not given. You must supply a valid eight character (or less) name to be used for the setup. Versicom will append the extension '.VER' to the name, and use that as the DOS file name. (See also 6.1.0 Setup Files, 3.1.0 Command Line Parameters, and 5.7.1 Save a Setup).

19: Error writing to the printer.

You have instructed Versicom to capture the current session to the printer, but Versicom is having trouble accessing the printer. Check the connections to your printer, and verify that it is 'On-Line'. If there is still a problem, make absolutely sure that your printer is connected to the parallel port LPT1:, because this is where Versicom expects to find it. If it is not, you can use the DOS MODE command's redirection capability to fool Versicom into sending the capture text to the port where your printer is connected, such as COM1:, (e.g. *MODE LPT1:=COM1:*).

20: Error writing to the capture file CapFile.

You have instructed Versicom to capture the current session to an ASCII disk file, but Versicom is having trouble accessing that file. If this occurs, it is most likely that there is a problem with the disk you are attempting to save the capture data on. Check to see that there is sufficient space available on the disk. If needed, consult your DOS manual for assistance.

21: Could not find Setup file SetupFile

Per your instruction via the Setup pull down menu, Versicom attempted to load the specified setup file 'filename' which contains certain User Command and Function Key assignments. Apparently, the setup file does not exist in the current directory. Either correct any typo's, or exit to DOS (with <Alt><D>) and change to the correct directory. (See also 6.1.0 Setup Files and 5.7.2 Load a Setup).

22: Cannot use the RunLog while a macro is executing.

You have chosen to activate the RunLog screen, but you are apparently already in the process of executing a macro. If you wish, press the <Esc> key to halt the execution of the macro, and then access the RunLog screen. Otherwise, you must wait for the macro to complete execution before you can access the RunLog screen.

23: Run log not available (not enough memory).

When you start Versicom, it attempts to determine how much memory you have available at that time. If enough memory is available, it will set up the RunLog for you, using as much memory as is 'reasonably' possible. If not enough memory is available for even the smallest version of the RunLog, it will be disabled for the current session. If you wish to still use the RunLog, you must free up some memory by removing other programs prior to starting Versicom. Or, if necessary and possible, install more memory in your computer!

24: Unsuccessful attempt to open the capture file.

You have instructed Versicom to capture the current session to an ASCII disk file, but Versicom is having trouble accessing that file. It is possible that the file you specified is 'Read Only' and Versicom is not allowed (by the operating system) to use it. If you have checked everything else, it is most likely that there is a problem with the disk you are attempting to save the capture data on. Check to see that there is sufficient space available on the disk. If needed, consult your DOS manual for assistance.

25: 'filename' not found [in the root directory].

This message will appear on two occasions. First, it is possible that you attempted to use the DOS Shell option (from the terminal screen), and Versicom couldn't find the Command.COM file in the root directory, ('filename' will actually be Command.COM). In this case, make sure that the file Command.COM is in the root directory. Secondly, it is possible that you used the RUN command in a macro, and Versicom couldn't find the program/command you specified to execute, see your DOS manual if needed. (See also 5.10.0 The DOS Shell Option, and 7.2.13 Run).

26: Path to 'command' not found.

This message will appear if you attempt to use the macro command RUN, and accidentally specify an invalid path to the program/command you wish to execute, where 'command' is the program/command you try to execute.

27: Access denied to 'filename'

For whatever reason, DOS will not allow Versicom to access the desired file. For further information, consult your DOS manual.

28: Not enough memory to run a DOS shell.

In order for Versicom to use the DOS Shell option, there must be enough system memory (RAM) available to load an additional version of the DOS system file Command.COM. Try to 'unload' a few programs, or quit Versicom while leaving the connection active. If you quit Versicom, you will of course lose your command history and any RunLog information. (See also 5.11.2 Quitting and Leaving the Connection Active).

29: Empty prompt command ignored.

When using the macro command PROMPTor the special command @@PROMPTyou must include a parameter which is the new prompt to be recognized by the system. This error will occur if Versicom is given an empty prompt parameter, (PROMPT ""). (See also 7.2.8 Prompt and 8.2.0 @@Prompt).

30: Invalid TIMEOUT value in current macro.

The timeout value determines the amount of time Versicom will continue to wait for a string, when using the WAITFORcommand. Valid timeout values are either zero, or anywhere from 1 millisecond to 60 seconds. In other words, this would correspond to a command which is either TIMEOUT "0", or anywhere from TIMEOUT "1" through TIMEOUT "60000". The value between the delimiting quotes used for the parameter, must be in the range of the above values, or this error will appear. (See also 7.2.11 Timeout).

31: SEND command ignored.

While reading the current executing macro, Versicom found a SENDcommand which had an empty parameter string. In other words, there was no value between the quote signs which delimit the send parameter. A command such as SEND "" will cause this error to appear, (see also 7.2.2 Send).

32: Invalid UPCASE parameter, (should be ON or OFF).

Versicom has encountered either an @@UPCASEcommand, or the macro command UPCASEwhich has an incorrect parameter. The only valid parameters for either command are "ON" and "OFF". Any other parameter will cause this error message to appear. (See 7.2.14 Upcase and 8.3.0 @@Upcase).

10.2.0 Fatal Messages

Messages contained in the following list should never be seen during normal program execution. If you do however see one, you may be able to "recover" by simply starting Versicom again. Usually, if a fatal error occurs, Versicom will not disconnect the telephone connection. Therefore, you can normally restart Versicom and pick-up where you left off.

In the event that a particular error continues to repeat itself, you should contact those people responsible for maintaining Versicom, and work with them in attempting to resolve the problem.

200: Message length greater than 79 in USERINFO
201: Too many windows in WINDOWS.
202: No window to pop in WINDOWS.
203: Background colors limited to 7 in WINDOWS.
204: Invalid display adapter mode detected in WINDOWS.
205: Baud rate specification is invalid in AUX_INTR.
206: Parity specification is invalid in AUX_INTR.
207: Word length specified is invalid in AUX_INTR.
208: Number of stop bits specified is invalid in AUX_INTR.

209: Must define serial port parameters in AUX_INTR.
210: Unknown pull down in VER_KEYS.
211: Maximum macro events exceeded in VER_MCRO.
212: Cannot open file in SAVESETUP.

11.0.0 Glossary

ASCII

The *American Standard Code for Information Interchange (ASCII)* is a code which defines the translation of alphabetic & numeric characters, symbols, and control instructions into seven bit binary code.

Baud Rate

The term *baud rate* refers to the speed which bits of information are transferred between your computer and the host computer. The *baud rate* is a good indicator of how quickly commands are received by the host computer (sent from you), and conversely how quickly text is received by you (sent from the host computer).

Generally, the character transmission speed is approximately one tenth of the *baud rate*. Therefore, at 1200 baud, characters will be sent and received at a rate of approximately 120 characters per second.

Case Sensitive

Case sensitive is a phrase used to describe a system or object which is recognizes the differences between both upper or lower case characters. When dealing with objects which are *case sensitive*, you must be careful to preserve the upper and lower case status of each character. A computer which is not *case sensitive* will recognize both upper and lower case characters as the same corresponding opposite-case character.

Default

The term *default* refers to the nominal status of object, when you encounter it for the first time. In other words, unless you change the status, the object will remain in its nominal or *default* status. For example, the *default* status of your automobile is "Turned Off". Normally when you first get into your car in the morning, you change the *default* state of "Turned Off" to "On" by starting the car.

Dumb Terminal

A computer terminal is usually considered "dumb" when it does no actual processing of commands locally. When you are operating a personal computer, you are actually using a very "smart terminal" because it contains a microprocessor, and processes each command or instruction locally.

However, if you use a modem to connect your personal computer to another computer (a remote computer) for the sole purpose of using the remote computer, your computer in effect becomes a *dumb terminal* because all of the real processing is done by the remote computer. In this case, your personal computer appears to be very "dumb" in relation to the host.

Host Computer

A *host computer* is one which you are temporarily "visiting" via a modem (or some other connection). The term is meant to imply that you are a "guest" of the "host", for a temporary session. In this sense, the host computer does most of (or all) of the computing, and as a visitor you are usually seen as a *dumb terminal.*

Interrupt

An *interrupt* is the occurrence of a process, while an initial process is already being serviced or processed. *Interrupts* can be initiated by either hardware or software, and usually result in the computer jumping to another routine (or program) momentarily, to take care of a specific task.

For instance, on most computers when you press a key on the keyboard, an *interrupt* occurs and causes the computer to momentarily jump to a second routine which checks to see which key you pressed. In the case of Versicom, an *interrupt* occurs each time a character is received from the host computer. This *interrupt* causes Versicom to momentarily stop what it is doing in order to get the new characters, and store them until you are ready to see them.

Macro

In the case of Versicom, a *macro* is a set of commands (or instructions) which are combined into a single object (an ASCII file) so that they can be issued together as a group, an unlimited number of times. An example of a macro would be one which contains the combination of commands required to dial the telephone number of your host computer, send your user i.d., wait for you to type your password, and start a program on the host computer.

Serial Port

A *serial port* is a hardware device which when connected properly to your computer, will allow you to communicate with another similarly configured *serial* device. When using a modem, the *serial port* is the device which handles the communication between your computer and the modem. Your modem software (such as Versicom) then handles the communication between your modem and the host computer. The device is termed *serial* because any information which passes "through" the port is handled as a long series of single bits, as opposed to a large group of bits being transferred simultaneously.

Stop Bits

When your modem communicates with another computer's modem, they must share a common protocol (or set of rules) so that they are not attempting to "mix apples with oranges". The number of *stop bits* is a characteristic which helps each modem separate and organize all of the bits which make up individual characters. If the number of stop bits specified for each modem is different, one modem will misunderstand where one character ends and another begins. Specifically, the most common protocols for computer transmissions specify either one or two *stop bits* which instructs the modem to look for that specific pattern to end a character.

Syntax

The *syntax* of a command refers to the way the command is structured, in the same sense that a sentence must also be structured. As a sentence must have certain structures for certain situations, (such as a noun, followed by a verb), Versicom commands have certain specific structures or *syntax* also.

Parameter

Additional information which is included with instructions to Versicom, in order to assist in the selection of certain options or operations. For example, let's assume we have a command called *Light* which allows us to turn a light on or off. If we simply specify the command *Light*, there would be no indication of whether we wanted to turn the light on or off. So, in this case we would typically provide a *parameter* which determined whether the light would be *on* or *off*.

To turn the light *on* in our example, we would say *Light On*. To turn the light off, we would say *Light Off*. In either case, the word *Light* is the command, and the first *parameter* is either *On* or *Off*. In addition, we might wish to include a second *parameter* which specified which light to control. In this case we would say *Light Off Upstairs*. This final example demonstrates the use of two *parameters*.

Parity

In using your modem to communicate with a host computer, Versicom can utilize many of the features provided by your modem. One standard feature of the modem is their ability to correct errors which are detected during the transmission of a character. If there are problems with the telephone connection, then one character such as an 'A' may be sent by the host, and without error checking you might receive a 'T'! One method of error detection/correction is to use *parity* checking.

Parity refers to the equality of a certain quantity of objects. Specifically in the case of the modem, the *parity* of the expected bits of data is checked to assure that no artificial bits were lost or added during transmission. If parity is set to *even*, then the modem expects to see an even number of bits for every character. If it is *odd* then the modem expects to see an odd number of bits for every character. If it is set to *none*, then no *parity* checking is done.