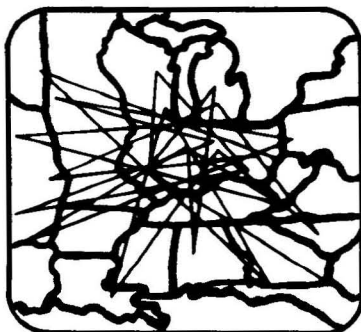


SynComm

**Modem communications
for your computer**

By Pete Goodeve



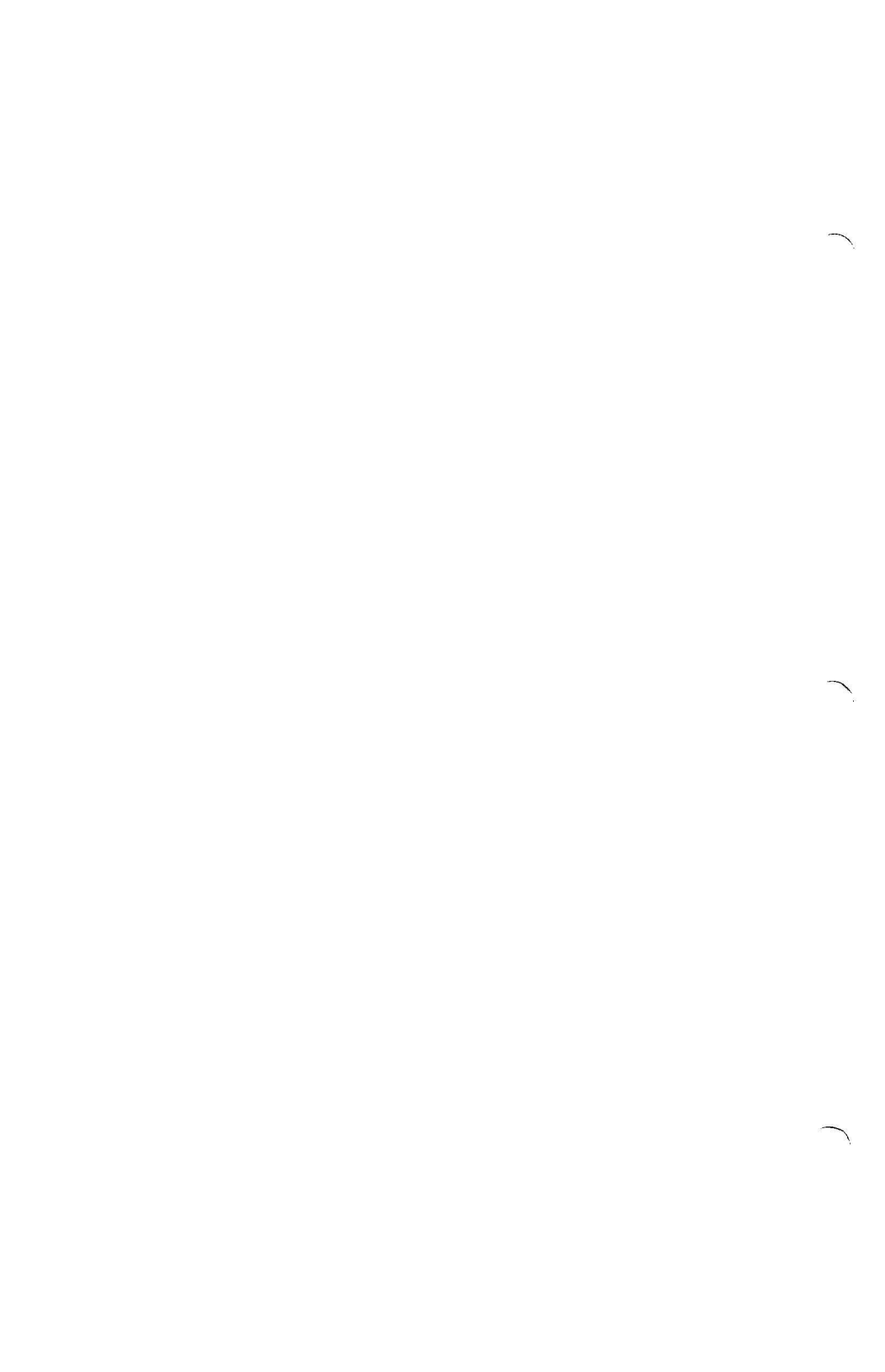
**Contains SynComm disk and user's
guide. Requires Atari computer with
minimum 48K memory, Atari-compatible
modem and disk drive. Printer optional.**

synapse

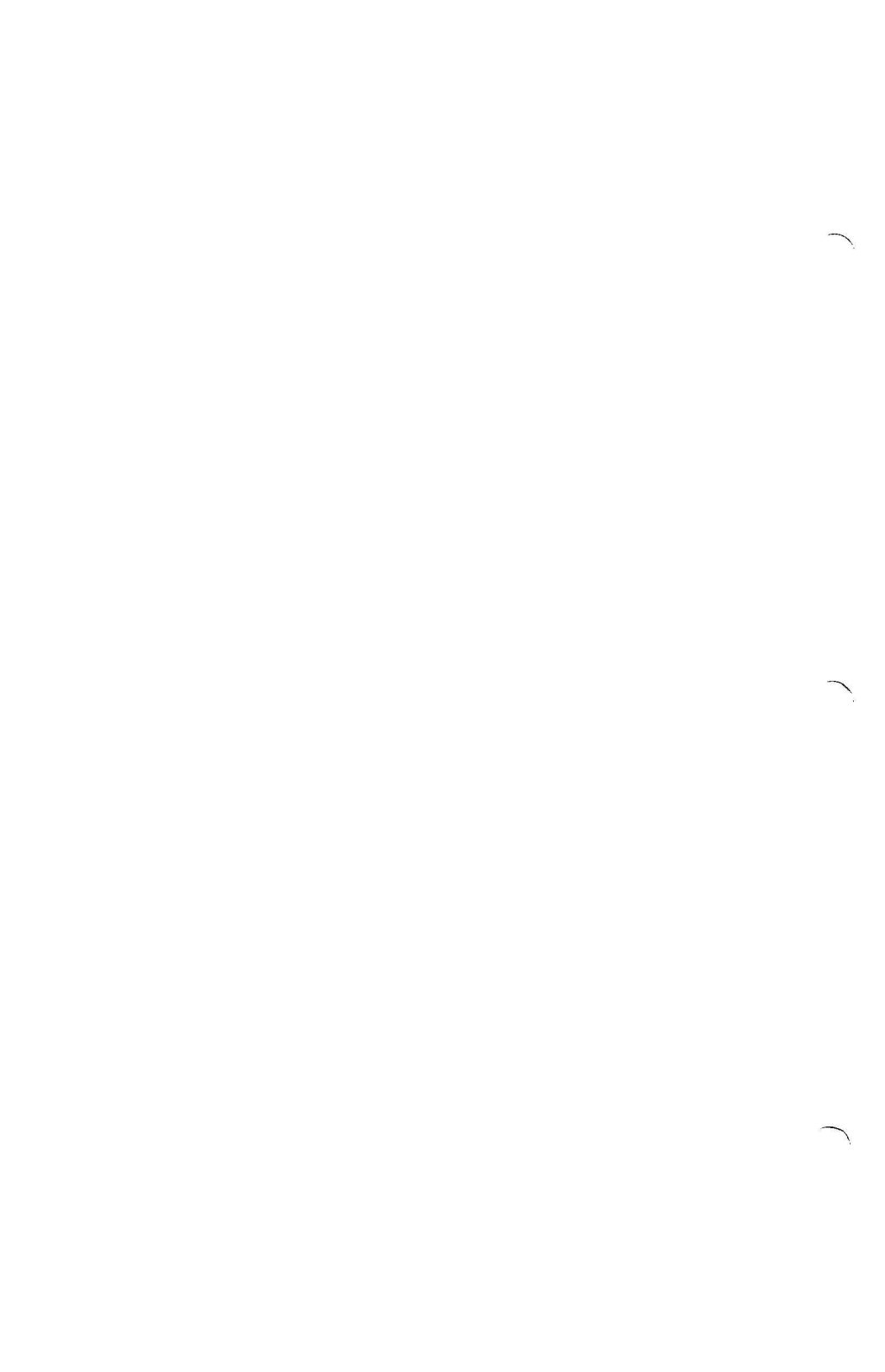
© 1983 Synapse. All Rights Reserved.
Synapse and Atari are registered trademarks of
Synapse and Atari, Inc., respectively.
Made in the U.S.A.



Contents	Page
I Introduction	1
II Getting Started	3
1. What You Will Need	3
2. Booting the System	4
3. The Terminal Screen	5
III System Commands	9
1. The Command Screen	9
2. Text Blocks	11
3. Saving Incoming Text	12
Memorize Data	12
No Memorization	13
Block Delimit	13
4. Saving Text Blocks	14
Write Block	14
5. File Specifications	16
6. Getting Blocks From Disk	17
Read Block	19
7. Creating Blocks Directly	20
Screen Editor	20
Transmitting Text Blocks	21
Segmented Text Blocks	22
8. Viewing Stored Text	23
View Block	23
Deleting Blocks	23
IV System Configuration	25
1. Configure Menu	25
2. Christensen Protocol—X Modem	29
3. Atari to Atari Link	31
V Advanced Users Utility	33
1. Configure Utility Program	33
2. Command Menu	34
Configuration Menu	35



3. Configuration Files	45
4. Character Translation	46
VI Reference	49
1. A Word On Modems	49
2. Communication Basics	50
3. In Case of Trouble	51
<i>SynComm</i> Won't Load	51
Error Messages	52
4. Quick Reference Page	53
5. Glossary	57
6. Index	65
7. Warranty	67





I Introduction

SynComm™ is a flexible and easy to use program for computer communications. With *SynComm* you will be able to access any computer information service that can be reached by telephone. You will have unlimited resources at your disposal and the ability to devise and send your own computer messages, texts, or data.

SynComm has been carefully designed for easy use and convenience. As you become more familiar with *SynComm*'s various applications, you will find the exciting world of computer communications opening up to you in new and interesting ways. However, to make the most of *SynComm*, it is important to read this manual carefully, familiarizing yourself with the various features.

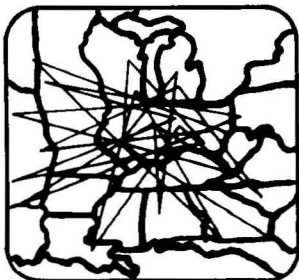
SynComm is one of the most versatile programs of its kind. Text coming in from other systems scrolls smoothly up the screen and words that would overflow one line can be shifted down to the next. This makes the screen display much easier to read during data transmission. Program commands can be entered at any time with *SynComm*. You can also store incoming text in memory blocks to be accessed later. You can record the text from the memory blocks to disk files or, if you have a printer, you can get hard copy of them. In addition you can preload saved files or already created text segments into memory and send them to other systems. In fact, you can have many text blocks in memory at once, which can save you considerable time while on-line.

SynComm also provides a special mechanism for error-free transfer of files (the Christensen Protocol, also called X-modem). It can only be used if the system at the other end has the same facility. However, it can also be used with some systems based on other computers. The protocol is rapidly becoming an unofficial standard for ATARI to ATARI file transfers.

If you frequently connect to a system that requires program settings different from the defaults built into *SynComm*, you can change the settings and save them in a file known as a configuration file. This will save you time when you set up for transmission, since all you will need to do is load the configuration file with the desired settings into the system with a few easy key strokes. There are certain features, such as the translation of characters and baud rate, that can only be altered by loading a configuration file. A utility program that lets you create your own such files, as well as some ready-to-use configuration files for common situations, are supplied on the reverse side of your *SynComm* diskette.

You may have a text string sequence that you use repeatedly. For example, some information services have a fairly complex log-on procedure, or if you have an intelligent modem you may want to send it commonly used phone numbers or other commands. You can create a file on disk containing any such sequence, then step through it whenever you need to with a few keystrokes.

If you have a friend with an ATARI and *SynComm* you can link systems directly for information transfer without going through a bulletin board or information service. See "ATARI to ATARI Link," Section VI.



II Getting Started

This manual has been designed so that the computer novice can easily step through the first part learning the basics of computer communications. The rest of the manual includes information for the more advanced and professional user. We have not included a tutorial in this manual due to the fact that much of how this program functions depends upon the type of hardware being used. The best way to approach learning to use *SynComm* is to read through the entire manual (whether you are a beginner or an advanced user), and then use the quick reference section for supplemental information while you are on-line.

What You Will Need

1. An ATARI Computer with a minimum of 48K memory.
2. An ATARI compatible disk drive.
3. An 850 interface or compatible RS232 interface. (If using the 850, make certain you are connected to port 1.)
4. A TV set or other video monitor.
5. A modem — either acoustic or direct connect.
6. A cable to link the RS 232 connector on the modem to the serial port on the interface.
7. A telephone line.

Optional:

1. Blank formatted diskettes (use the DOS formatting program. See your ATARI manual for details).
2. A printer if you wish to print your text and data transmissions.


Modems

Modems, disk drives and RS 232 interfaces vary in terms of system booting and cable requirements. Therefore, we suggest that you follow individual manufacturers operating

and hook-up requirements for your modem and RS232 interface.

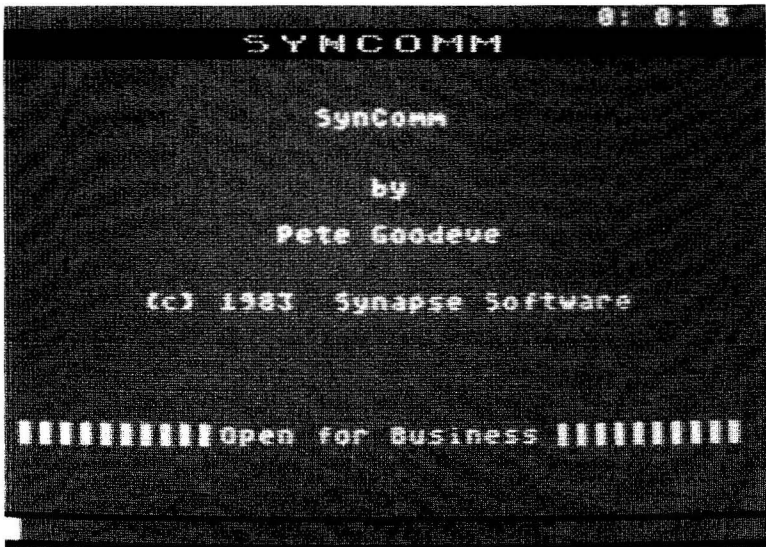
Booting The System

1. Remove any cartridges.
2. Turn on the 850 interface.
3. Turn on the disk drive.
4. Turn on your modem.
5. Insert the program diskette.
6. Turn on the computer.

 **Important Note:** If your computer is a 600XL or 800XL, you must disable BASIC by pressing **OPTION** as you turn on the computer. You can release **OPTION** as soon as the program begins to load.

If you have any trouble booting, remove the program disk, turn off all components and repeat the booting instructions.

The program takes about a minute to load. As the program loading is completed, your first screen known as the terminal screen appears and should look like this:



Once the main *SynComm* display appears, you are in a position to dial up and connect to the information service of your choice. However, before you do so we suggest you read this section with the program running on your system so you can get some familiarity with its basic features.

The Terminal Screen

Let's take a quick look over the information displayed on the screen. Starting at the top, you see:

Elapsed Time Clock

This is a real time clock used to keep track of how much time is spent on-line.

To set the clock back to zero, hold down **(START)** and press **(Z)**. This timer can be reset at any time during program operation.

Text Window

The text window displays all incoming and with some modems, outgoing transmissions as well. It can hold up to 20 lines of text and uses fine scrolling for easy reading.

Below the text window you see:

The Echo Line

The flashing cursor appears in the far left. Type a few keys on the keyboard: you will see them appear on the echo line. (They are also being sent out through the 850 module to the modem, but there is nothing beyond that to receive them yet.) Depending on your equipment, you may or may not see your text appear in the scrolling area. Some modems echo characters sent to them while they are sitting idle. The scrolling area only displays characters arriving through the modem.

Lower Status Line

The lower-right corner should read:

free:18013 characters

Information shown here is used when storing text to memory and will be covered in more detail in Section III.

Let's try using some of the basic terminal screen functions. First clear the screen.

HOLD (START)

PRESS (F)

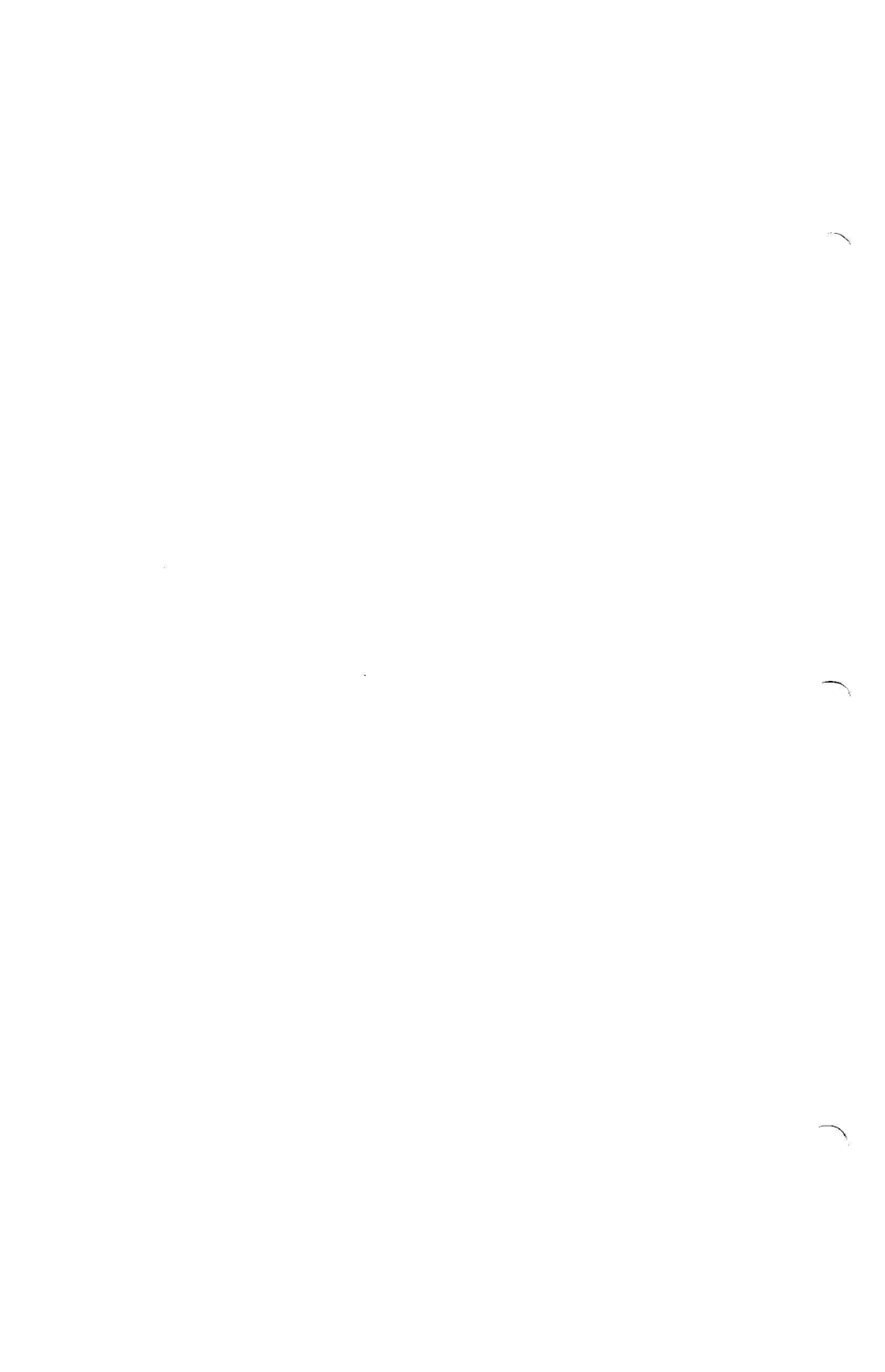
The terminal screen will quickly scroll up leaving an empty text window.

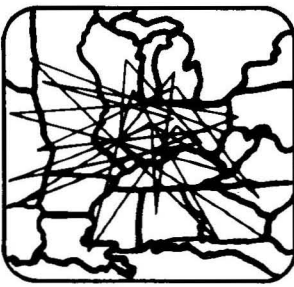
Next, we'll set the clock to zero.

HOLD (START)

PRESS (Z)

This has been a brief look at the terminal screen. At this point you can dial up a friend or BBS, however we suggest that you take the time to read through the remaining sections. These sections will demonstrate how to make use of the full power of *SynComm*.





III System Commands

The system commands are used in the process of saving data and text into resident memory, X-Modem transfer of files, system configuration and access to disk operations. To access these features you are given the option of using three different function keys. *SynComm* commands are selected by holding **OPTION**, **SELECT**, or **START** and typing the initial letter of the desired function.

OPTION

Displays the command screen. It also clears the command display area.

SELECT

Displays the command screen and the last set of commands used.

START

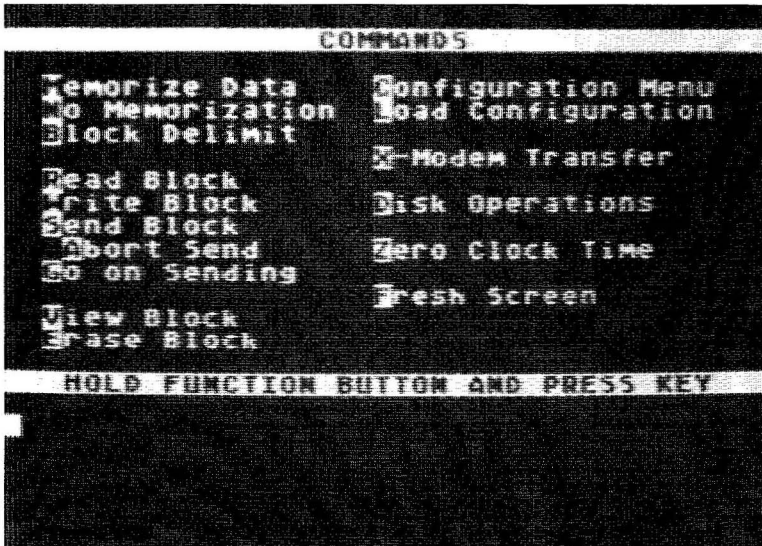
This is the expert users key. It immediately executes any valid command without displaying the command menu.



Important Note: You may use any of these function keys to execute a system command, however, to simplify our system explanations we will only use one key. Feel free to interchange any of the function keys using the one most convenient for you.

1. The Command Screen

To view the command screen, press **OPTION** and hold it down.



All the available command functions are shown on the Command screen with their identifying letters as inverse characters. You can use capitals or lower case with all command transactions. File names and other entries will be automatically converted to capitals where necessary.

When a command requests a user response, a **RETURN** alone is always permissible. This will either terminate the command (usually if it is the first question of a command sequence) or skip only that part of the string of commands.

Let's try a disk command:

PRESS **OPTION**-**D** for DISK OPERATIONS

! **Important Note:** Make sure you are holding down the **OPTION** key while Pressing **D**.

You can now release **OPTION**. The Command Screen will remain until you are done with it.

At the bottom of your screen appears the prompt:

Directory
Erase
Rename
Set Default Drive

These are four possible disk operations which will be discussed later. To leave Disk Operations, or any other selection from the Command Menu,

PRESS .

You should be back at the Terminal Screen.

PRESS and hold it.

You are back at the Command Screen again. Notice the command display area has not been cleared.

RELEASE

PRESS and hold it.

This time the information on the Command Screen is erased and a fresh Command Screen is displayed. You can always use to review the previous command transaction or two.

2. Text Blocks

One of the advanced features of *SynComm* is its provision for multiple, independent blocks of text stored in memory. These can originate from incoming text or from disk files. Short ones may even be created using the ATARI Screen Editor. You can transmit any text block over the telephone line or write it to a disk file. You can also view it on the Command Screen or print it on your printer.

The advantage of independent text blocks is the freedom you have to organize your time while on-line. Before making the connection, you can load any files and messages that you expect to transmit. During the session, you can record messages from the other end into separate blocks (up to the capacity of your memory) without having to worry about how you are going to file them until you hang up the phone.

Up to 20 blocks are allowed. They can be of any size up to the limit of the free space remaining. Watch the display on the lower status line of the Terminal Screen. This will inform you how many free characters remain available for storing text blocks. Blocks are identified by number. The lowest available number is allocated when the block is created and it retains that number until it is erased. You may erase blocks in any order, and at any time, to recover free space.

Commands that manipulate blocks (Write, Read, Send, Erase, and View) all begin by listing the currently existing blocks. An asterisk next to a block number indicates that it was recorded from incoming text, as opposed to being created by you with the screen editor or loaded into memory from a disk file. The operations that may be performed are detailed in the sections that follow.

Blocks do not have to contain only text, although they are referred to as text blocks. The program treats them simply as strings of 8-bit bytes. Depending on the Configuration options, you can transmit and receive blocks either as raw data or with translation appropriate to an ASCII character stream. (See section V Advanced Users Utility.)

You can have blocks segmented into pieces for transmission one by one for things like log-on sequences, by placing a special character at the break points. (See Segmented Text Blocks in Section III, Item 7.)

3. Saving Incoming Text

Memorize Data

Initially characters received by *SynComm* will only be written to the screen, and will not be recorded. To record the data in a text block, you would use the MEMORIZE DATA option on the Command Menu. Do this by, **PRESSING** START - M for Memorize Data

The message **storing** will appear on the lower command line at the far left of the Terminal screen. Make sure that you have this function activated BEFORE downloading text you may want to save from another system, otherwise the text will not be saved to block form in the computer's memory. You will either have to access it again or write it down by hand if you want to save it.

Saving text to block, however, is only permanent storage as long as your computer remains on. When you shut your computer off the blocks are lost. For permanent storage, text blocks must be saved either to a disk file or to a printer.

No Memorization

To stop recording, follow the same process again:

PRESS - for No Memorization

The message **open blk** will appear at the bottom right of the lower command line indicating that the recording has stopped but that the block is still open for receiving more data when you select: M again. (If no text is actually in the block, the **open blk** message will not appear.)

This procedure of alternating M and N can be used to record separate pieces of the incoming text into the SAME text block. All text that comes onto the screen after N has been pressed and before M is pressed, is not saved to a block. Everything that comes onto the screen after M has been pressed and before N has been pressed will be saved to a block.

Block Delimit

When you have completed storing text into a block with the M-Memorize DATA and/or N-NO MEMORIZE, it is then necessary to use the B-BLOCK DELIMIT command to close the block. A block of recorded text cannot be accessed in any way (to view it, read it, erase it, or write it to a disk or a printer) until it has been terminated by the B-

BLOCK DELIMIT command. When you give this command, the block is closed to any further input and is allocated an identifying number. The **open blk** status message will then be cleared.

If you press: B before giving the N command (while you are still storing), the block will be closed but recording will continue in a new one. In this case the **storing** message on the screen will remain displayed.


In either of these instances (where the **BLOCK DELIMIT** command is given while in memorize data mode or where the command is given after turning off memorization), the message **new blk n** (where n is the actual number of the block) will appear on the left side of the top status line, and if you hold down **(SELECT)**, you will see the message **stored to blk #n** written to the Terminal Screen.

4. Saving Text Blocks

Write Block

To save blocks to disk or to the printer use the **W-WRITE BLOCK** command. Remember you may use **(OPTION)**, **(SELECT)**, or **(START)** while selecting: **(W)**.

Giving this command will switch you to the Command Screen where a list of currently available blocks is displayed. The number of the most recent block recorded will be on the first line in the command message area.

 **Important Note:** The information “**current blk is #n**” is handy because the creation of blocks is not always in order. It can be out of order when you have erased blocks. The next block created is the lowest available number. So if you have 15 blocks and erase Nos. 3-7, the next created block number will be 3 and not 16.

The prompt at the bottom of the screen will then ask you:

which to file? blk:

Respond with:

The number of the block and

PRESS .

Pressing without making an entry will end the operation.

Writing blocks can be done either to a disk or a printer. The procedure is executed one block at a time. So, if you want to write 3 blocks to a disk, you must perform the procedure described here three different times. The prompts at the bottom of the Command Screen will aid you in this operation.

At this point, you should be sure that you have either an appropriate disk in your drive on which you want to store the text, or a printer connected to your system if you want to write the text to your printer. The disk will need to be formatted using the DOS program. Follow the instructions in your ATARI manual to perform this operation.

The next prompt is for the name of the file you wish to write it to. A new file is needed for each block you are saving. You cannot append a block to a file that already exists.

If you don't include an extension in the file name you type, *SynComm* will give it the default one ".TRX". Read the following section on File Specifications for more information about file names and their extensions.

If you want to save a file to disk,

ENTER: FILE NAME

PRESS

However, if you want to write the block to a printer instead of to a disk file, you should:

ENTER: P:

PRESS

It is important to remember the colon after the P so that *SynComm* will not interpret the command as a file name.

Next you will be asked if you want to erase the block from memory.

PRESS if you do not want it erased, or

ENTER: Y,

PRESS , if you want it erased.

Any response not beginning with Y means No.

You will again be presented with information on the blocks in storage and you can then execute this operation again if you want to store or print more blocks.

A block may be deleted at any time. See Section III, Item 7 on Deleting Blocks.

5. File Specifications

A DOS file name can be up to eight characters long with an optional extension of up to three characters. If an extension is used, it is separated from the name by a period. You may indicate on which drive the file will be written. To do this, precede it with "Dn:" where "n" is the number of the drive (n will be a value between 1 and 4 depending on how many drives you have).


When *SynComm* asks you to supply a file name, simply type it in, followed by . You may use upper or lower case. If you include an extension this will always be used. However, if you omit both extension and separating period, many *SynComm* operations will assume a default extension which indicates the file's purpose. If you really want the file to have no extension, use a period with no characters following.

Default extensions used are:

- .TRX — file created from a text block by “Write Block”
- .XMR — file received by “XMODEM” protocol
- .TCF — Configuration file

The “Read Block” command does not assume any extension. If you request a file to be read that is not on the disk or if you omit the extender, you will get the message **couldn't find it!** and the prompt will be repeated. If you try to write a file of the same name as one that already exists, you will get **file already exists — do you want to replace it?** An answer of Y will proceed with the operation. Any other response will cause a new prompt. To abort in any situation, press .

As always on the ATARI, you can specify a peripheral device or program instead of a disk file. The ones applicable to this program are the printer (P:) or the screen editor (E:).

 **Important Note:** You cannot use the ATARI 410 or 1010 cassette recorder with this program.

Make sure you do not forget to enter the colon for the peripheral device. Otherwise, the program will assume you want to open a file of that name.

6. Getting Blocks From Disk

For convenience in managing your files, *SynComm* provides a few of the usual DOS functions: DIRECTORY, ERASE FILES and RENAME FILES, and SET DEFAULT DRIVE. To use these functions:

HOLD , or , and
PRESS -DISK OPERATIONS

You will be presented with the selection:

Directory
Erase
Rename
Set Default Drive

ENTER: First character of the desired selection

Either D,E,R, or S. Do not type until you have responded to the immediate prompt that the selected function makes. The program remains in DISK OPERATIONS (signalled by an *) after each function until is typed.

Directory

If you Enter: -DIRECTORY, the program will display:

*** directory filespec:**


If you just type you will get a list of all the files on the disk. Alternatively, your response can be any standard DOS file name specifier, either a specific name or including the usual wild cards "?" and "*", followed by . If the disk is the selected one (see below) there is no need to prefix it with a device specifier (Dn:) but you can list any disk by specifying it explicitly. (See the DOS manual if you are not sure about file name specifications.)

Erase Files

If you type: -ERASE, the program will ask for:

*** erase Filespec:**

Your response should be either a specific name or a name using wildcards to designate a group of files. Typing alone in this case exits the function. Be very cautious with this function.

 **Important Note:** Unlike DOS, *SynComm* does not request confirmation before deleting files. We suggest using the DIRECTORY function with the same filespec first to prevent erasing the wrong file.


Rename Files

The R-RENAME function will ask for:



Rename Old,New Filespec:

In this case your response should be two file specifiers, separated by a comma. The original name first, followed by the name it is to be changed to. Wild cards can again be used, where appropriate. A cautionary reminder is worthwhile here, too: The DOS file manager (which *SynComm* uses) has no protection against renaming one file to be the same as another that already exists, and from that point on they are Siamese twins — inseparable without disk surgery.

Set Default Drive

The S-SET DEFAULT DRIVE function permits you to designate which drive is to be the default. All file references not preceded by a disk specifier will be directed to this drive. Initially it is set to Drive 1 (D1: — the master drive). When you type: S, the currently selected drive is displayed and you are asked for a new setting, which should be a single digit between 1 and 4. Typing  alone, or making an invalid entry, leaves the old selection unchanged.

Read Block

To bring a file from disk into memory, PRESS - for Read Block

The Command Screen will appear and you will be prompted with: **file to read: .**

When the file has been read, the program will write the number of the created block to the Command Screen and immediately switch back to the Communications Screen. Use **SELECT** to check the information on the Command Screen if you need to.

SynComm gives you 18013 characters of free space in a 48K machine. If you get the message **partial load**, your file is too big for the program's memory and the remaining text cannot be accessed.

7. Creating Blocks Directly

Screen Editor


You may want to generate short blocks of text to send out by using the ATARI screen editor. To do this:

PRESS **START**-**R** for Read Block

Now, instead of opening a file,

ENTER: E:

You will be presented with a blank screen on which you can type in anything you like.

 **Important Note:** You may use the standard screen editor facilities for creating your text, but remember that a line of text is read when you press **RETURN**. This means that you can't change it after that. If you go back and change it, it will be put into the block twice.

When you have entered the last line (including the last **RETURN**),

HOLD **CTRL** down

PRESS **3**

This will generate the Atascii end-of-file command. Now you are out of READ BLOCK and looking at the Terminal Screen ready for normal operation. It is probably a good idea to

view the block at this point (see Viewing Stored Text) before storing it to disk or transmitting it.

Transmitting Text Blocks

Options:

Send Block
Abort Send
Go On Sending

You can transmit a block of text from memory by:

PRESSING (START)-(S) for Send Block

This switches you to the Command Screen where you are requested for the number of the block you want to send.

ENTER: (The number of the block.)

PRESS (RETURN)

Transmission will begin immediately and you will flip back to the Terminal Screen. The message **sending** will appear on the lower status line. Unless for some reason you are connected to a system that does not echo the characters it receives, the echoed text will scroll up the main screen so that you can view your transmission. (You can use HALF DUPLEX if there is no echo, see System Configuration, Section IV).

If you need to, you can hold up transmission momentarily by:

PRESSING (START)

Pressing (OPTION) or (SELECT) will also work, but will cause you to switch to the Command Screen. You should normally press (START) to halt the transmission.

To abort the transmission completely,

HOLD (START) until the scrolling stops,

PRESS (A) for Abort Send.

While *SynComm* is sending, it will not accept keyboard

interaction or commands except: A.

To restart from the point at which you stopped transmission, **PRESS** **START** - **G** for Go On Sending

This command can be used at any time after interrupting the transmission unless you have sent or deleted something else in the meantime. Once the complete block has been sent, the command will have no further effect.

Segmented Text Blocks

Blocks intended for transmission can be divided into segments by a special character. The current SEND BLOCK operation will stop when this character is encountered (and OUTPUT TRANSLATION is enabled), just as if an ABORT command had been given. It can be resumed later from that point by giving the GO ON SENDING command. This lets you set up such things as log-on procedures, where you have to send a sequence of several (fixed) strings in response to prompts by the receiving end.


You can put these stop characters wherever you like in the text block. If one is in the middle of a line, the rest of the line will be sent when you resume with the G command. If the string to be sent needs a **RETURN** at the end, put the stop at the beginning of the next line.



You can create short blocks for log-on and other purposes conveniently by using the screen editor with the READ BLOCK command and save them on disk with WRITE BLOCK.

To generate the segment separator code, type the following sequence:



1. Push the ATARI key once to put characters in inverse video.
2. Hold **CTRL** down
3. Press: comma (,) an inverse heart will appear.
4. Press the ATARI Key again to return to normal text.

 **Important Note:** The special character is hex code 80 (decimal 128), which does not occur in normal ASCII text. This code is also used for “don’t pass” in Character Translation. All special properties of this character are turned off when translation is disabled and it becomes just like any other.

8. Viewing Stored Text

View Block




You can display any text block in the Command Screen area:

PRESS - for View Block

You will be asked: **Which Block Do You Want to View?**

Respond with the appropriate number,

PRESS 



Pressing  will cause the display to pause. If you also press -ABORT SEND while holding , the display will abort just as it does in the SEND BLOCK command mode above.

The command repeats the block number request after each display is completed.

PRESS  to exit

Deleting Blocks

Erase Block

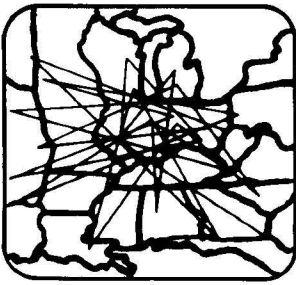
You can remove unneeded blocks by,
PRESSING - for Erase Block

You will be asked for the number of each block you wish to delete. When there are no more blocks to be deleted, you will be returned to the Terminal Screen.

To exit this operation at any point prior to deleting the last block,

PRESS

Blocks may be deleted at any time, even when another block is open for recording. The space they occupied is freed immediately.



IV System Configuration

1. Configure Menu

To call up the Configuration Menu,

HOLD

PRESS

Your screen should display the following:

```
CONFIGURE SYSTEM
Trap words..... YES
MON/XOFF..... NO
Duplex (half)..... NO
Incoming translation..... YES
Outgoing translation..... YES
8-bit (atascii)..... NO
Echo ctrl chars..... NO
Flow transmission..... 0
Configure to file

ENTER OPTION -- OR RETURN TO EXIT
```

Word Wrap

When WORD WRAP is on "YES", words written to the main scrolling area that would overflow the line are moved down to the beginning of the next line. When WORD WRAP is off ("NO") characters will always be written up to the right margin, and words will be split as necessary. This option affects nothing except the display. The default is "YES".

XON/XOFF

This option is only important if the computer at the other end is likely to send you data when you are using a command like Print or View. In most cases, you will know ahead of time whether you will need to set it. The other system must be set up to recognize this code.

Most information services and bulletin boards recognize the ASCII characters XON (equivalent to ASCII Control-Q on the keyboard) and XOFF (equivalent to ASCII Control-S on the keyboard) as transmission control signals. When they receive XOFF, they will cease sending characters until XON arrives. So if you have to attend to something else, this feature can be used to prevent the screen being overwhelmed with characters.

Half Duplex

The terms "Half Duplex" and "Full Duplex" refer to the capabilities of the communications link, not the actions of the program. Modem links are always Full Duplex. Sometimes, however, you may encounter a system that doesn't echo the characters to the screen that you are sending it, and this has come to be called Half Duplex.

The Half Duplex option controls whether or not transmitted text is also sent to the screen. If the remote end is echoing characters itself, you want it set to NO (Full Duplex), the default value. You need to set it to YES if the transmitted characters are not being copied directly to the screen, and you want them to be.

Incoming Translation

As described in Section IV on Character Translation, some incoming ASCII characters have to be translated to their ATASCII equivalents to be displayed properly on the screen. Normally text being stored in memory is translated in the same way. To prevent stored text from being translated (if


the incoming data is already ATASCII for example, or is not text), set the Incoming Translation option to NO. This option does not affect the screen display. Default is YES.

Outgoing Translation


Outgoing characters are normally assumed to be ATASCII and are translated where needed to their ASCII equivalents (see Character Translation). To prevent translation of text blocks that are being transmitted from memory, set the Outgoing Translation option to NO. This option does not affect characters typed on the keyboard, which are always translated. Default is YES. If Translation is turned OFF, the segmented text blocks feature will also be inactive.

8-Bit (ATASCII)

ASCII characters are normally expected to have seven meaningful bits, with the eighth (highest) bit either being a parity check or set to zero. The 850 module, therefore, is normally configured to strip this bit from both incoming and outgoing characters. This is not acceptable when transferring either ATASCII or non-text data, both of which may have information in that eighth bit. To make the system pass all bits of the character unchanged (although still subject to the Translation Tables and options) set the 8-bit (Atascii) option to YES. The default is NO.

 **Important Note:** In many cases, setting to YES will not harm your normal communication activities.

Echo Cntrl Chars

Depending on your own preferences and the system you are accessing, you may or may not want to see control characters (those typed with the  plus the key such as the arrows keys) on the echo line. To make the control characters visible, set the ECHO CNTRL CHARS option to YES. The default position is NO.

Slow Transmission

This feature will probably be needed in only a very few cases. Quite often when you try to transmit a message from *SynComm* to a Bulletin Board System running on someone else's microcomputer, it will end up totally garbled. This is because, for one reason or another, the program at the other end is unable to keep up with the stream of characters coming in at full speed. You probably won't find this a problem with the bigger commercial systems, at least at 300 baud. A suitable delay may be set between each character using the Slow Transmission option.

Typing: S-SLOW TRANSMISSION will evoke a request for a numeric parameter setting. The units of this parameter are approximately 5 millisecond intervals. The specified number of these intervals is inserted between each character sent. Thus typing: 10 will put about 50 milliseconds between each character. A suitable value for your situation will have to be found by trial and error. The default is of course 0.

Configure To File

If you have set up a configuration that you are likely to use on frequent future occasions, you may want to save it to a disk file. Use the C-CONFIGURE TO FILE option for this. A file name will be requested of you. We recommend that you do not supply an extension. Let the system supply its standard default .TCF. The complete configuration will be saved, including any previously loaded parameters and translation tables. There is no provision in this mode for you to give the file a descriptive header like those generated by the configuration utility. You can always add one later by using that utility.

Read In A Configuration File

To read in a configuration file from disk, use the "R" selection from the Main Menu. You will be asked for the

name of the file to read. There is no need to give the extension, as long as it is the standard. TCF.

2. Christensen Protocol - X Modem

A number of Bulletin Board Systems, based on different machines, have adopted a standard method of transferring files over the phone which is more reliable than just sending character by character. Any type of file, not only text, can be handled.

The protocol, developed by Ward Christensen (actually for CP/M based systems) is in the public domain. It is a little complex and there is no need to describe it fully here. Briefly it sends successive blocks of data, each of which must be properly received before the next is sent.

You can use this mode only if the computer you are hooked into is also equipped with the protocol. AMIS type ATARI-based systems are among those that do. You will also have to find out from the Board how to invoke the protocol at that end. The procedure is usually very simple.

An important use for the protocol naturally is to pass files between two ATARI computers, both running *SynComm*. This is also very easy. It just requires you and your friend to agree on the files to be transferred. Then you both enter the X-modem mode, one sending and one receiving and the transfer will start automatically. The protocol is self-synchronizing.

How to Enter Christensen Protocol

X-Modem Transfer

To enter the X-modem mode, either Transmit or Receive, you use X-MODEM TRANSFER on the Command Menu. Before you do this, you will have to ensure that the remote end is ready to go. If the remote end is a Bulletin Board, this

means giving it whatever commands it might need. When you are communicating with another user, it does not matter which of you enters the mode first. The system passes synchronizing messages to ensure that they are both ready. Current configuration settings have no effect on operation in this mode.

Transmit Mode

To send a file from your computer, first take whatever steps are necessary to get the other end ready to receive it, then give the X-MODEM TRANSFER command. (**START** X)

A prompt will appear on the screen asking you to Enter: T to transmit or R to receive. You should respond with T for Transmit. The next prompt asks you for the name of the file you want to send. Type in the name of the file and press **RETURN**. Transmission will now begin.

Unfortunately, some Bulletin Boards don't adhere strictly to the protocol (which states that the receiving end must keep sending prompts until the transmitter is ready). In such a case nothing will happen when you start a transmission because *SynComm* is waiting to be prompted before beginning. You will see the display "block 1" remain static on the screen. If the display doesn't change to "block 2" after fifteen seconds or so, you will have to start the transmission yourself, by holding down **START** and typing **G** (for "Go").

The program keeps you informed on the Command Screen of the progress of the transfer. At the end you will be returned to the Terminal Screen. You can abort if you need to with the **START**-**A** command. Keep the **START** button down after typing: "A" until the program responds, as there may be some time delay. You will then have to abort the remote end. To do this, type **CTRL**-**D**.

Receive Mode

To receive a file onto your disk, first get the other end into Transmit mode by whatever method is suitable for that system. Give the X-MODEM TRANSFER command. Enter R for Receive to the prompt. When asked for the file name with the second prompt, designate a name for the file into which you want the incoming information to be placed.

If you need to abort the operation,

PRESS (START)-(A),

keeping the (START) button down until the program responds. You will also have to terminate the other end. Use the appropriate method, which may be typing (CTRL)-(X).

3. ATARI To ATARI Link

You can link up with another ATARI running *SynComm*, rather than a central information service, without difficulty. Messages typed on one keyboard will appear on the other's screen, and by coordinating your actions you can send and receive text blocks just as when talking to a central service. There is one major difference: You will not see the text you type echoed in your own scrolling area, unless you set the Configuration to half-duplex. Of course, it will appear on the key echo line in any case. You have an ideal method of transferring files in this situation using the Christensen protocol (X-Modem).

When you set up the phone connection, the modem at one end must be set to Answer, and at the other to Originate, so that each recognizes the other's frequencies. (Some old modems don't have an Answer mode. If neither has, you won't be able to use this feature. Some newer modems don't have a selection switch, but make their own decision depending on what they hear. In this case, just make the connection and things should work). As soon as the

connection is made, you are free to start sending messages.



1. Configure Utility Program

This program, on the reverse side of your *SynComm* program disk, has one purpose only: To generate Configuration Files for *SynComm*. Configuration Files contain settings for nearly all the parameters used to control the communication process. Some of these— the ones on the Configuration Menu — can be reset from within *SynComm*. Others can only be changed by loading a file created with the Utility. For most uses of *SynComm*, the default settings of the latter will be the right ones or, if not, those in one of the supplied Configuration files will be. This program is intended mainly for the advanced user with special requirements. It will be assumed in the following instructions that you are familiar with the commands and options in *SynComm*.

Use the utility to set baud rate, character translation, and RS232 parameters such as parity. Screen color and left margin can also be changed to suit your preferences. When you create a file, you are given a blank screen area in which you can write descriptive text. This screen will be displayed anytime the file is loaded. (You can also save Configurations from *SynComm* itself, but they will not contain a description).

Loading The Configuration Utility

Insert the *SynComm* diskette in the drive with the reverse side up. Turn on the computer and the utility will load. There is no need to have the 850 module or any other peripheral connected when you are running this program, but they will not do any harm if they are.

If you will want to load any of the Configuration Files supplied by Synapse on the diskette, leave it in for now. Otherwise remove it and insert a disk formatted by Dos 2.0S, and you may also want DOS.SYS and DUP.SYS on it for your own convenience. You can use a disk that already contains files, but make sure you have enough space to store your configuration files. Each Configuration File takes eleven sectors. (If you have more than one drive, of course, you can write or read a diskette in any of them).

2. Command Menu

The Utility program starts up with a display of the Command Menu choices. These are:

```
COMMANDS
C onfiguration menu
T erminal configuration

I ncoming translation
O utgoing translation
V iew translation

W rite config to file
R ead config from file

D isk functions
Q uit to DOS

PRESS DESIRED INITIAL CHARACTER
```

To make a selection, just type the initial character (don't follow it with a **RETURN**). The function buttons **OPTION**, **SELECT** and **START** are not used in the Utility program.

The Configuration Menu is similar to that in *SynComm* with some added parameters, described below. Terminal

Configuration sets up RS232 parameters, baud rates, parity, DTR, and RTS settings, and auto-linefeed after

RETURN.

Incoming Translation allows you to set up custom translations of characters received by *SynComm*. Outgoing Translation does the same for transmitted characters. View Translation gives you a quick look at all the translations performed by the current configuration.

Write Config. to File lets you create the descriptive text screen, and write this and the configuration itself to a file name of your choice. Read Config. from File brings in an already existing file, setting up the configuration for examination and/or modification.

Disk functions are identical to those in *SynComm*. Quit to DOS does just that, provided that you have a disk containing DUP.SYS in drive No. 1 at the time.

Configuration Menu

The Configuration Menu has the following selections:

```
CONFIGURE SYSTEM
Trap words..... YES
DTR/XOFF..... NO
Duplex (half)..... NO
Incoming translation..... YES
Outgoing translation..... YES
8-bit (Atascii)..... NO
Echo cntrl chars..... NO
Slow transmission..... 0
Left margin..... 0
Background luminance..... 4
True (background)..... 9
Character luminance..... 10
ENTER OPTION -- OR RETURN TO EXIT
```

Make your choice by typing the corresponding initial character. The first eight items (Wrap Words through Slow Transmission) are identical to the ones in *SynComm's* Configuration Menu.

The last four choices appear here only. They are all numeric values: To set them, type the desired initial letter and respond to the NEW VALUE: prompt. No checks are made for sensible entries, so make sure the display shows the intended values before leaving the menu.

The only reasonable values for Left Margin are probably 0, 1 or 2, but you can make a wider margin if you like. *SynComm* assumes that your TV or monitor does not cut off any of the display, and defaults the margin to 0. The right-hand margin is always at column 39.

The hue and luminance settings allow you to set up screen color and brightness in a similar way to using the BASIC SETCOLOR statement (Chap. 9 of your BASIC manual). If we denote the background luminance value by X, the hue value by Y and the character luminance by Z the settings are equivalent to the pair of BASIC statements:

Setcolor 2,Y,X
Setcolor 1,Y,Z

Values should only be in the range 0-15. Hues generated by particular values are given in Table 9.3 of your BASIC manual (p. 50). Brightness is proportional to the luminance value (which should actually be an even number). Make sure that the background and character luminances differ by at least 4.

Terminal Configuration

This menu is used to set up parameters in the 850 module and its handler. You are not likely to find a need to change any of them, except perhaps baud rate, for normal use. No attempt will be made to explain them in detail here. For more information see the 850 manual.

The menu looks like this:

Baud Rate
Parity (In and Out)
DTR Setting
RTS Setting
Linefeed After CR

An option is selected as usual by typing the initial character of each option. Baud rate requests numeric input of the desired rate (see below), but the others are cycled through a set of possible states, moving on one state for each key press. The set of states is different for each option.

Baud Rate

When you Enter: B, the program will request: DESIRED BAUD RATE: To which you should reply with one of the following (or to cancel your request):

110, 134, 150, 300, 600, 1200
1800, 2400, 4800, or 9600

If you give a value that is not one of these, you will get a reminder of the possible values and be reprompted. Of all the above, the only ones you are likely to need are 300 (the default) and 1200.

Note that *SynComm* is not designed for continuous reception at the higher speeds. The smooth scrolling feature adjusts its speed to incoming data, and will work properly up to 2400 baud. Above this rate, characters may be lost if a continuous stream of data overflows the input buffer. The Christensen protocol mode, on the other hand, can handle the maximum rate, but it is limited by disk transfer speed.

Parity

By Typing: P, the parity setting can be sequenced through the states: None, Odd, Even, Set. Note that though the menu label says: In and Out, and in fact both directions in the 850 handler are set appropriately, the current release of *SynComm* does not report errors in incoming data. The only use of the parity setting is compatibility with systems that insist on a particular parity for characters they receive. If set to NONE, the default, the parity bit on transmitted characters is always zero. SET on the other hand means it is always one. ODD and EVEN adjust the bit to the appropriate parity mode.

DTR Setting

“D” sequences this parameter through UNCH, OFF, ON. UNCH (Unchanged) means that the DTR signal at the interface connector will not be altered when the configuration is loaded. Unless any other choice has been made at some point, it will in fact be off. OFF and ON force the signal to the specified level, where it will remain until the next forced change.

RTS Setting

This is identical to DTR setting, except that it controls the RTS line of the interface.

Linefeed after CR

Many terminals and other output devices require both a (to return the carriage or cursor) and a linefeed (To feed the paper or move the cursor down) at the end of each line. In almost all other situations a alone suffices to terminate the line. The 850 will add a linefeed to each (EOL) sent out, if you so desire. This option has two states, YES and NO, the second being the default.

Incoming and Outgoing Translation

We will look at the two translation table modification functions together at first, as the procedures concerned are identical. Only the table displays differ, and these will be dealt with in the two sections following this one.

When you enter one of these two modes, by either the "I" or "O" selection from the Main Menu, you will be presented with a screen showing an empty display frame at the top and some text, including a prompt, in the lower half. Initially the prompt requests: "Enter Char Numeric Value:" You type in the number of the slot in the table representing the character that you wish to examine or change. You can give the decimal value or, if you prefix it with \$, the hexadecimal equivalent. Once you have selected hex, all future entries will be in the same base until you prefix one with a space. This is indicated by a change in the legend at the bottom of the screen.

When you have entered a slot number, its current contents will appear in the frame at the top. The meaning of the items in the display will be discussed in the following sections. The prompt will now say: "Translate to Value:". If you want to change an entry, type in the revised value, otherwise type

RETURN. Or you can use the “S” or “R” option described below. The base expected for the new value, decimal or hex, will be that last selected, unless you change it with a \$ or space prefix.

Entering a new value causes the revised slot to be displayed below the original, and you will be prompted a second time with “Translate to Value:”. If you are satisfied, type **RETURN** or an option selection. Otherwise try again. You will not leave the slot as long as you keep giving numeric values. The original entry remains displayed in case you want to forget the whole thing and change it back to what it was.

Responding with a **RETURN** alone to the “Translate Value:” prompt clears the display frame and unless you are in Sequential or Reverse mode, brings you back to the “Enter Char Numeric Value:” prompt. You can then either repeat the cycle by selecting a new slot, go into Sequential or Reverse mode, or type **RETURN** again to get back to the Main Menu.

If, instead of typing **RETURN** alone to either prompt, you type **S**-**RETURN**, the program will enter Sequential mode. The next table entry in sequence will be displayed and you will get the Translate to Value: prompt. Each time you just respond with **RETURN**, the next slot is automatically displayed, instead of a request for a new slot. Entering a value on the other hand, terminates Sequential mode and you will be back to the normal cycle. You can resume sequential display by using “S” again. If you want to leave Sequential mode without entering a value, type: **N**-**RETURN**.

To step through the table in the reverse direction, use “R” instead of “S”. Behavior is otherwise the same as for “S”. You can also use “R” when you are in Sequential mode (or “S” when in Reverse) to change direction.

Input Translation Display

The display of the Incoming (input) Translation Table has two sections. On the left is shown the slot number in three formats: Decimal (Dec.) hexadecimal (Hx), and the ASCII character equivalent (Asc). The value contained in the slot appears on the right in four versions: decimal, hex, ATASCII equivalent (At) and screen function (Func). The special formats will be described below. Due to space restraints on screen, these may be a bit cryptic, but they should be understandable.

There are two special slot values. Most entries will usually be zero, which causes the supplied value (the slot number) to be passed through without translation. (This saves having to clutter up the table with uninteresting entries.) This passing through is indicated by \blacklozenge instead of numeric values. The ATASCII and function equivalents are still shown.

The other special value is 128 (Hex 80), which prevents a received character from being passed at all. This value is reported as "Ignored" in the display.

Output Translation Display

The Outgoing (output) Translation Table is displayed in a similar fashion to the Incoming one, but the slot number on the left is now the original character value (from memory or the keyboard), and on the right is the transmitted value.

The original character is given in five formats: Decimal (Dec), hex (Dx), ATASCII (At), key combination that generates it (key), and screen function (Func). The transmitted character is shown as decimal, hex and ASCII.

A zero entry in the table is once again shown as \blacklozenge and the value 128 (\$80) is "not sent". (See Input Translation Display above.)

Display of ASCII

Printable characters in the Asc column are displayed as themselves. The only exceptions are those that are not in the ATARI display set. There are three of these, the tilde and left and right braces. They have been given the three-letter lower-case mnemonics tld, lbr and rbr.

Non-printing control characters are denoted by their standard ASCII mnemonics of two or three letters, such as ACK, ESC, CR and so on. If you do not know them, decoding these may require a reference text on ASCII. There is no space to go into them here.

Display of ATASCII

All ATASCII characters are shown as themselves, with the single exception of `RETURN` which is not displayable. It appears as a square (an inverse space; the screen handler represents it in the same way).

Display of Key Codes

This column in the table is an attempt to compress into three characters a representation of the key combination needed to generate the code. If a key cap has the character engraved on it, that is usually used to represent the code. In most cases, this is true even if the `SHIFT` (or caps lock) is needed. Thus "a" and "A" for example. Sometimes `SHIFT` is indicated explicitly (see below). Special function keys like `RETURN` and `TAB` (unshifted) are given distinctive reverse video labels.

When `CTRL` is needed to generate the code, the key cap indicator is prefaced by an inverse video C. In cases where the shift key does more than simply shift to upper case or top-row printing characters, it may also be indicated by an inverse S. When the shift or control keys are applied to other special keys, like `TAB` or `DELETE` these are now represented by two-character mnemonics (tb, dl) in normal

video, preceded by the inverse C or S.

If the code also requires the inverse video toggle (ATARI key) to be on, this is indicated by a vertical bar at the left of the column.

Display of Screen Editor Function

For most codes, this column is empty. It is used to identify those ATASCII codes that have special meaning to the ATARI Screen Editor. Cursor movement is shown by the appropriate arrows, and other functions by three-character labels. Insertion or setting of some entity is indicated by (+), as (+)-CH for Insert Character, (+)-TB for Set Tab. Similarly deletion or clearing is marked by (-) ((-)-CH, (-)-TB). Other operations have fairly obvious three-letter mnemonics.

View Configuration

Instead of using the Input or Output translation table examination modes to step through the entries, you can get a quick look at them with the "V" selection from the Main Menu. Only non-zero entries are displayed in this mode, up to sixteen at a time. The display format, however, is identical to that of the corresponding examination mode.

You will first be asked if you want to see the Input Translation table. Answer: (Y)-(RETURN) if you do. If there are fewer than 16 non-zero slots in the table, you will be shown them all. Otherwise, you must PRESS: (SPACE BAR) to see the next batch. Press: (RETURN) to skip any further display of this table.

Then you will be asked if you want to see the Output table. Follow the same procedure. When you exit this table, you will be returned to the Main Menu.

Write Configuration to File

The “W” selection from the Main Menu allows you to write your custom Configuration file to disk with a screen of descriptive text. You will be given nineteen blank lines of screen on which you are free to enter anything you like, using standard screen editor operations. What you type goes nowhere but to the screen for the moment.

When you are satisfied with your text, which should briefly indicate special parameter settings and so on, press: **BREAK** or **CTRL-3**. You will be asked to verify that the text is okay. If you don't answer “Y”, you will be put back in the text area for more modifications, otherwise you will be asked for the name of the file to write. (If you decide to abort the operation, answer “Y” to the Text OK prompt, and then just type **RETURN** for the file query). You normally should not give an extension for the file name. It will automatically be set to .TCF, the standard extension for Configuration files.

Note that during the writing process the screen will blank for a few moments. This is normal. You will return to the Main Menu automatically.

Read from Configuration File

To set up the parameters in the Utility's tables according to an already existing file, use the “R” selection from the Main Menu. You will be asked for the name of the file to read. There is no need to give the extension, as long as it is the standard .TCF.

If the file has a descriptive text frame, this will be displayed when the file is loaded. Press **RETURN** to get back to the Main Menu. If there is no text, you will go straight back to the menu.

3. Configuration Files

CompuServe Configuration Files

For slightly greater convenience in using the CompuServe network, we have included the configuration file CSERV.TCF on the reverse side of the system diskette (with the Configuration Utility). It is also suitable for the The Source and other networks based on large machines.

The differences between this and the standard defaults are that the **(BACK SPACE)** key (code 126) sends an ASCII BS character (code 8) to delete characters instead of DEL, and that **(SHIFT)-(DELETE)** sends NAK (code 21, the same as **(CTRL)-(U)**) which is CompuServe's delete-line. In addition the XON/XOFF option is set to YES, because CompuServe supports this feature.

Default Configuration File

You can restore all parameters and translation tables to their original default values by loading DEFAULT.TCF from the reverse side of the system diskette with the Load Configuration command.

1200 Baud Configuration File

The configuration file B1200.TCF on the reverse side of the system diskette sets the baud rate to 1200. All other parameters have their default values.

4. Character Translation

ASCII is an ideal code for communication, but it didn't quite suit the designers of the ATARI who wanted to represent graphic symbols in the character set. Because of the differences between ASCII and ATASCII (ATARI ASCII), codes generated from your keyboard cannot be sent directly to another system. Some of them have to be translated first. In the same way, incoming ASCII codes must be occasionally adjusted to display correctly on your screen. *SynComm* handles this with incoming and outgoing Translation Tables. These also have some use in matching keys such as to the conventions of a particular information service.

Because of these differences, codes generated from your keyboard cannot be sent directly to another system. Some of them have to be translated first. In the same way, incoming ASCII codes must be adjusted sometimes to display correctly on your screen. *SynComm* handles this with incoming and outgoing Translation Tables. These also have some use in matching keys such as to the conventions of a particular information service.

SynComm achieves this with two Translation Tables, one for transmission and another for reception. An entry is made in the appropriate table for each character that must be translated. Others are passed unchanged. The current outgoing table is always applied to characters typed on the keyboard, and the incoming table always translates characters that appear on the screen. Configuration options determine whether text being recorded to or sent from memory is also translated.

SynComm is initially loaded with default tables that will suffice for nearly all your common needs. To change them you load a Configuration File embodying the desired tables. A configuration file with a slightly different table for CompuServe is included on the system diskette. You can make files of your own with customized tables if you want,

using the Configuration Utility. The actual table format is described later for those who would like to do this.

The default outgoing translation table makes these substitutions:

Key	Code	ASCII	CODE	STD KEY CODE
Cursor-left	30	BS	8	CTRL -H
Back-Space	126	DEL	127	DELETE/RUBOUT
Tab	127	HT	9	CTRL -I
Ctrl-2 (beep)	253	BEL	7	CTRL -G

We convert cursor-left to ASCII backspace and Back-Space to the Delete character because many terminals and systems respond to the ASCII codes in this way. For others, CompuServe in particular, translating Back-Space to BS is preferable. None of the other cursor-control or editing characters are translated because there are no consistent ASCII equivalents. You can always send ASCII control characters in the conventional way, of course, by typing the appropriate letter with the CTRL key held down.

The incoming translation table defaults are:

ASCII	Code	ATASCII	Code
Nul	0	Ignored	—
BEL	7	Beep	253
BS	8	Cursor-Left	30
HT	9	Tab	127
LF	10	Ignored	—
{ lft. Brace	123	Inv. [219
{ rt. Brace	125	Inv.]	221
~tilde	126	Inv.	189
DEL	127	Back-Space	126

Some systems send NULs after RETURNS and so on, so the table is set to ignore them. Line-Feed (LF), not required in ATASCII, often follows RETURN in ASCII systems so this too is ignored. The two brace characters and tilde are not represented in ATASCII — their codes are either graphics or screen control — so arbitrary inverse-video replacements

have been chosen.

Each *SynComm* translation table is 256 bytes long, one slot for each possible character. Each candidate code for translation is looked up in the table by its numeric value. If the corresponding entry is zero, the code is passed unchanged. If the entry is hex 80 (decimal 128), the character will not be passed at all. Any other value will be taken as the translated code for the character. Unless you are doing some very unusual translation, most entries will be zero.

Do not confuse the preceding process with the 850 module's Translation modes (850 manual Appendix 6). *SynComm* uses the 850's Light Translation mode with parity ignored, unless the 8-bit (Atascii) option in the Configuration Menu is set to YES. Light Translation clears the eighth bit of each character to conform to ASCII, and will change an EOL (ATASCII) to CR (ASCII) if the translation table has not changed it first. It alters nothing else.

8-bit mode is "No Translation" to the 850, which then passes all 8-bit codes exactly as it encounters them. If you are communicating in ATASCII with another ATARI you will have to use this mode, and you will also need it if you are passing non-text data and cannot use the Christensen protocol. You will almost certainly have to turn off table translation in these situations too.

In normal operation the parity handling features of the 850 are not used. The parity bit of outgoing characters is left unchanged. It will be zero if Light Translation is in use. The parity of incoming characters is not checked, and will be set to zero in the Light Translation mode. If for some reason the information service you wish to call needs a particular parity setting, you can use the Configuration Utility to create a suitable configuration file. Be warned, though, that setting parity will interfere with the 8-bit mode.



VI Reference

1. A Word On Modems

Acoustic Modem

If you have an acoustic modem (the kind with rubber cups for the telephone handset) you should do it this way: Dial the number and wait for the computer at the other end to answer. The ringing tone should stop after one to three rings, and a moment later you should hear a high-pitched answer tone. (If you do not, the connection has not been made for some reason. Try again.) Now, after establishing the connection, place the handset in the cups on the modem taking care to get the cord at the correct end. The connection is now complete.

Direct Connect Modem


If your modem is the direct connect type without autodialing, follow the same steps as for the acoustic modem, but instead of putting the handset in the cups turn the modem switch to the on-line position. With most modems you should then replace the handset in its cradle. However, some modems connect through the telephone itself, so check the modem manual to be sure how to operate yours.

Autodial Modem

If you have a "smart" modem that has autodialing controlled by characters sent to it through its RS232 connector, you do not have to touch the phone at all. You can type the character strings on the ATARI keyboard. Any responses from the modem will appear on the screen. The particular strings to use will have to be found from your modem

manual. For systems you access often you may find it convenient to put the dial-up procedure in a segmented text block as a file on disk (see section on Segmented Blocks).


However you do it, once you are connected to the information service, any characters you type on your keyboard will be received at the other end, and anything it sends you will appear on your screen. You will need to log-on to the system according to the various log-on procedures of each company. Commercial systems such as The Source and CompuServe will need an account number and a password. Their brochures have clear explanations for accessing their systems. Free Bulletin Boards can usually be accessed by anybody.

 **Important Note:** Please always be sure of the number you are trying to reach and dial it carefully... especially at 3 am.

2. Communication Basics

When you join a commercial information service, you will get a manual that explains how to use it. However, here we will include an overview of communication basics with some general information on bulletin boards.

The phone numbers of active boards can be obtained from users' groups or computer stores. A listing has also been included in the back of this manual in the Reference Section. Once you get onto one board, you will often find on it an extensive list of other boards. (Don't expect to find all the numbers still working, as boards go in and out of business pretty fast.)

When you dial up a bulletin board, you usually first get an identifying message on your screen. If you don't, see if typing  activates things. You may then be asked for a password. Some boards are truly private, but most use passwords just to keep some control over use of the system. If you do not give a password, you will probably still

be allowed access, but some features of the system may be closed to you. The system may also automatically give you a password for future use after asking for identification. Be prepared to write down your password.

Once through the preliminaries, you will be presented with either a Menu of allowable commands or instructions. One or more of these commands will lead you into the message display facility, which is likely to be a chronological list of messages left by others. Simple commands let you step through these in order. Often you can just scan the titles if you want to. *SynComm* makes it easy to record the messages you are interested in: Just start recording immediately before requesting playback of each message and stop it again immediately afterwards. (See Recording Text).

You should be able to post your own messages too. You may like to save time and telephone charges by composing your notes beforehand, either with *SynComm* or a text editor. You will have to explore the other facilities the boards provide by yourself, but signposts are always clearly placed to guide you. Don't be cautious about trying things out. The system is there for your use. One thing to remember: If a log-out (hang-up) command is included in the menu, please use it when you are done. Some systems will not reset themselves properly if you simply hang up. Later callers will then be unable to get on the board.

3. In Case Of Trouble

***SynComm* Won't Load**

If *SynComm* won't load, check the following:

1. All connection cables should be in place.
2. At least 48K of RAM must be installed.

If it still won't load, at this point it most likely has something to do with your disk drive. If you have more than one drive, try loading from the other one (don't forget to set the disk address to 1). Also try loading other programs from your disk drive (sometimes this will work, but because of speed variations, you still won't be able to load *SynComm*). If you are sure that all of your equipment is operating correctly and you are within the 90 day warranty period established at the date of purchase, then you may mail us your *SynComm* diskettes along with the sales slip or proof of purchase and we will verify their condition and replace if necessary.

Error Messages

Most common Error Messages will be fairly self-explanatory. They are signaled with a beep, but note that the command terminates at the error. The program will immediately switch back to the terminal Screen. To see the message, you will have to press .

Couldn't Find It

The file requested from disk is not in the directory

File Already Exists, Do You Want To Replace It?

(No Beep). This is a warning rather than an error. To replace the original file with the new one, Type: -.

Full Up!!

(Appears on the lower status line). There is no more room in memory for text. You will have to erase something if you want to record more.

Last Segment

The last free block number has been allocated. You will have to erase something before giving another BLOCK DELIMIT command, otherwise later text may be lost.

No Blocks Free

There are no unallocated block numbers.

No Blocks To . . .

Where “. . .” depends on the command (no beep). There are no blocks of text currently in memory.

You tried to load configuration parameters from some other kind of file.

Partial Load — File Too Big For Memory

The file you read was too large for the available space. As much of the file as possible was read into the block.

Read Error . . .

Write Error . . .

RS232 Error . . .

Operating system errors are indicated by their ATARI error code (“. . .” above). You may get these when you make an error in the Disk Operations command, or if, for example, you try to write to a file that is locked. Refer to the front of your BASIC manual or elsewhere for the meaning of the numeric codes.

4. Quick Reference Page

Function Keys

- OPTION** Displays Command Screen and clears command display area.
- SELECT** Displays Command Screen and last set of commands used.
- START** Executes any valid command without displaying the Command Menu.

Command Screen

- (OPTION) - (M) Memorize Data: records data in a text block.
- (OPTION) - (N) No Memorization: stops recording data in text block.
- (OPTION) - (B) Block Delimit: closes text block when recording is completed.
- (OPTION) - (R) Read Block: brings file from disk into memory.
- (OPTION) - (W) Write Block: saves blocks to disk or printer.
- (OPTION) - (S) Send Block: transmits text blocks.
- (OPTION) - (A) Abort Send: aborts transmission.
- (OPTION) - (G) Go on Sending: resumes transmission at point stopped.
- (OPTION) - (V) View Block: displays any text block on the Command screen.
- (OPTION) - (E) Erase Block: erases unwanted text blocks.
- (OPTION) - (C) Configuration Menu: displays options which can be used to set new transmission parameters.
- (OPTION) - (L) Loads Configuration file from disk.
- (OPTION) - (X) X-Modem Transfer: used to transmit and
- (OPTION) - (D) Disk Operations: contains DOS functions for managing files.
- (OPTION) - (Z) Zero Clock Time: sets the time clock to zero.
- (OPTION) - (F) Fresh Screen: clears the screen.

Disk Operations Sub-Menu

- (D) Directory: lists all files on disk.
- (E) Erase: deletes specified file or files.
- (R) Rename: renames file or files.
- (S) Set default drive: writes file to specified drive.

Configure Sub-Menu

W Wrap Words:

Yes: overflowing text is moved to beginning of next line.

No: characters written to right margin and words split as necessary.

X XON%XOFF:

No: halts transmission of data from remote end.

Yes: resumes transmission of data from remote end.

D Duplex (half)

No: when remote end is echoing characters.

Yes: when remote end is not echoing characters.

I Incoming Translation:

Yes: translates incoming ASCII characters to ATASCII equivalents.

No: does not translate incoming ASCII characters to ATASCII equivalents.

O Outgoing Translation:

Yes: translates outgoing ATASCII to ASCII.

No: does not translate outgoing ATASCII to ASCII.

8 8-bit (Atascii):

No: strips eighth bit from ASCII characters.

Yes: does not strip eighth bit from ASCII characters.

E Echo Cntrl Chars:

No: Characters typed with the control key are not displayed on the echo line.

Yes: Characters typed with the control are displayed on the echo line.

S Slow transmission: 0/:

Slows transmission when program at remote end is unable to keep up with the stream of characters coming over.

The units are 5 millisecond intervals. Typing 10 will put 50

milliseconds between transmitted characters.

(C) Configure to file :

Stores configuration to file for future use.

Configuration Utility Program

- (C)** Configuration Menu: displays options which can be used to set new transmission parameters.
- (T)** Terminal Configuration: displays options which can be used to set new transmission parameters for the terminal.
- (I)** Incoming Translation: sets up custom translations of characters received by *SynComm*.
- (O)** Outgoing Translation: sets up custom translations of characters transmitted by *SynComm*.
- (V)** View Translation: allows you to view the current incoming or outgoing translation.
- (W)** Write config to file: creates the descriptive text screen and writes this and the configuration to a specified file.
- (R)** Read config from file: brings in already existing file and sets up the configuration for examination and/or modification.
- (D)** Disk functions: contains DOS functions for managing files.
- (Q)** Quit to DOS: quits to DOS when a disk containing DUP.SYS is inserted in drive 1.

Configuration Sub-Menu

Has same features as Configure Sub-Menu on front side of disk with these additions:

- (L)** Left margin: 0: sets left margin to value specified.
- (B)** Background luminance: 4: sets screen brightness according to specified value between 0 and 15.
- (H)** Hue (background): 9: sets screen color according to specified value between 0 and 15.

- (C) Character luminance: 10: sets character brightness according to specified value between 0 and 15. (Choose an even number for best results).

Terminal Configuration Sub-Menu

- (B) Baud Rate: 300: changes baud rate. Use one of the following values for this setting: 110, 134, 150, 300, 600, 1200, 1800, 2400, 4800, or 9600.
- (P) Parity (In and Out): NONE: sets parity for compatibility with systems that require a particular parity for characters received.
- (D) DTR setting: UNCH: sequences DTR through unchanged, on, or off.
- (R) RTS setting: UNCH: sequences RTS setting through unchanged, on, or off.
- (L) Linefeed after CR: NO: adds a line feed to each carriage return sent out.

5. Glossary

Access: The method in which information is read from, or written to a diskette, bulletin board or information system.

ASCII: American Standard Code for Information Interchange. The standard code for communication with computers. Characters are represented as 7-bit codes, with an eighth bit for error checking.

Alphanumeric: The letters A-Z and the numbers 0-9, and/or combinations of letters and numbers. Specifically excludes graphic symbols, punctuation marks, and other special characters.

ATARI 850 Interface Module: Hardware device that allows an ATARI computer to connect to other devices, including a modem.

ATASCII: ATARI version of ASCII with eight instead of seven significant bits. Most characters, in particular the

alphanumeric ones, are the same, but ASCII control codes are graphics characters in ATASCII and inverse video characters are included.

BASIC: Beginners All-purpose Symbolic Instruction Code.

BAUD: A unit of data transmission speed usually equal to 110, 300, 1200, or 9600 bits per second (bps).

Bit: Abbreviation of binary digit. The smallest unit of information, represented by the value 0 or 1.

Block: Any group of data handled as a single unit by a program.

Boot: The process of initializing the computer for use by automatically clearing memory and loading the first few instructions which call other instructions. This gets the computer started, and brings the system you are using into action from the diskette.

Bulletin Boards: An electronic call-up service that allows users to compose and store messages to be retrieved by other users.

Byte: Eight bits treated as one unit and often used to represent a character.

Characters: Letters of the alphabet, numbers, punctuation marks, graphic symbols or any combination thereof.

Christensen Protocol: A bulletin board system where whole files are transferred (rather than transferring character by character) from disk to disk without bringing the file into memory first.

Command Screen: The screen where all transactions and prompts appear. It is accessed from the Communication Screen by depressing or for some commands. A Command Menu which displays the available program options may be displayed.

Configuration File: A saved set of transmission parameters which is used in a situation that requires settings other than the default values indicated on the Configuration Menu.

Configuration Menu: The screen which displays the options available for setting transmission parameters, including: Wrap Words, XON/XOFF, Duplex (half), Incoming Translation, Outgoing Translation, 8-Bit Atascii, Echo Cntrl Chars., Slow Transmission and Configure to File.

CPU: Central Processing Unit. The “brains” of the computer. Logic and control functions are performed in the CPU.

CRT: Cathode Ray Tube. The television receiver or monitor used to display computer output.

Data: Information of any kind.

Data Bank: An organization of data files containing information or reference material on a particular subject or subjects.

Data Diskette: The diskette upon which your data is stored.

Data File: A system of information organization on disk.

Default: Condition which exists when no instructions to the contrary are given to the terminal.

Dial Up: The procedure for accessing another system through a Modem.

Disk: Same as diskette.

Disk Drive: A device that rotates magnetic disks and accesses its data by means of a read/write head.

Disk File: Blocks of text saved to file on disk under a user-designated file name.

Diskette: The 5-1/4 inch magnetic storage medium on which data is stored.

Downloading Data: To transfer programs and/or data files from a computer to another device or computer. This usually means that your computer is the one receiving the data (see uploading data).

Echo: The return of a transmitted signal back to its source.

Echo Line: See Keyboard Echo line.

Error Message: Any of a number of messages which appear on your screen when you have attempted a function out of its proper order, pressed the wrong key or incorrectly inputted a Command. Other problems such as hardware malfunctions can generate Error Messages.

File: A collection of data (usually on disk) that can be referred to by name. The structure and contents of a file depends on its use (see Text File).

Filename: A Filename is the label for a File by which it is accessed. It usually gives some indication of the File's contents.

Formatting: Preparing a new disk for information storage. Formatting a disk causes any information on that disk to be erased. This should not be confused with columnar format.

Function: A computation built into the computer that can be called for by the program.

Hardware: The electronic components of the computer system.

High Resolution: The display on the television screen or monitor is composed of little dots known as a dot matrix. High resolution is the use of a large number of these dots to increase visual clarity.

Incoming Tex: Data arriving from another source into your computer's memory for viewing, storage or printing.

Information Service: Any of a number of companies that provide on-line data information to users for a fee. It may also be a free bulletin board system.

Input: To transfer data from outside the computer into the computer. This is a data transferring operation.

Keyboard Echo Line: The line on the Terminal Screen showing the characters that you have typed.

Load System: Transferring software into the computer's memory for operational purposes.

Log-On: The procedure of entering appropriate passwords and User-ID's in order to access an information system.

Main Dialogue: The primary interaction between two systems where an exchange of data or a direct communication is taking place.

Memory: Where information is stored in the computer.

Memory Blocks: Units of data stored in memory and identified by program-generated numbers, to be transmitted, saved, or printed as needed.

Menu: A program-generated list of options usually presented on the display screen. Selections from a menu screen will cause the program to execute indicated procedures.

Modem: Modulator-demodulator (acronym). A device that converts and reconverts digital signals from a computer into tone variations for transmission over standard telephone.

On-Line: A connection established between a peripheral and the CPU of either another system or of its own system for the purposes of processing and/or transmitting information.

Operation: The practical application of various options of the software, i.e., Memorize Data or Block Delimit.

Output: The transfer of data from inside the computer to outside, such as to a printer.

Parameters: Definable characteristics whose values define the values and limitations of a system.

Password: A code that allows a user to gain authorized access to an information system or bulletin board.

Peripheral: A device connected to a computer and under its control to some degree, usually used for input/output such as terminals, printers disk drives or other similar equipment.

Program: A sequence of software instructions given to a computer for the performance of certain functions or tasks. A program must be in a language that the particular computer understands.

Prompt: Signals which appear on the screen indicating the necessity of further input or the location of an input.

RAM: Random Access Memory

Read: The inputting of data from the diskette to the computer

Remote End: The system to which data is being transmitted or from which information is being accessed.

Scrolling: Moving text vertically or horizontally on a CRT screen so that portions which do not fit on the screen at one time can be viewed.

Smooth Scrolling: Moving text up the screen in a continuous fashion rather than a line at a time, so that it remains readable while in motion.

Software: Computer programs.

Special Characters: A character that can be displayed by a computer but is neither a letter nor a numeral. Punctuation marks are special characters. So are the ATARI graphics symbols.

Terminal: A device (CRT & keyboard) through which a user can communicate with a computer. With *SynComm*, your ATARI is in effect being used as a terminal to another computer. See Terminal Screen.

Terminal Screen: This screen is displayed on your monitor during normal operations. Incoming text appears in the scrolling area, including any characters typed by you which are echoed from the remote end.

Text Blocks: Text or data stored in memory in the form of blocks to be transmitted, stored or printed.

Text File: A File which contains only text characters (ASCII or ATASCII).

Translation Table: A table in which the program can look up an original code value to find a replacement value. *SynComm* uses this mainly to convert ATASCII to ASCII and vice-versa.

Transmission: The movement of data over a communication line from one device to another.

Uploading Data: Transferring programs and/or data files to an information service or bulletin board from your computer.

User: An individual who is using or operating a system of hardware and software.

User-ID: A code that identifies a user for authorized access to an Information System.

Wraparound: When the cursor reaches the right edge of the screen, it disappears and "wraps around" to the beginning of the next line.

Write: The transfer of data to a magnetic diskette.
SynComm writes out records as they are entered.

Write-Protect: Protecting a disk from having data written to it.

Index

ATARI to ATARI Link, 31
1200 Baud Configuration, 45
Baud Rate, 37
Block Delimit, 13
Character Translation, 46
Christensen Protocol—X Modem, 29
Command Menu, 34
Command Screen, 9
CompuServe Configuration, 45
Configure To File, 28
Configure Utility Program, 33
Default Configuration, 45
Directory, 18
DTR Setting, 38
Display of ASCII, 42
Display of ATASCII, 42
Display of Key Codes, 42
Display of Screen Editor Function, 43
Echo Cntrl Chars, 27
Echo Line, 6
Elapsed Time Clock, 5
Erase files, 18
File Specifications, 16
Half Duplex, 26
Incoming and Outgoing Translation, 26-27, 39
Input Translation Display, 41
Line Feed After CR, 39
Lower Status Line, 6
Memorize Data, 12
Modems, 49
No Memorization, 13
Output Translation Display, 41
Parity, 38
Read Block, 19
Rename Files, 19
Read Configuration, 28, 44
RTS Setting, 39

Screen Editor, 20
Set Default Drive, 19
Slow Transmission, 28
Terminal Configuration, 37
Terminal Screen, 5
Text Blocks, 11, 21, 22
Text Window, 6
View Block, 23
View Configuration, 43
Write Block, 14
Write Configuration, 44
XON/XOFF, 26

Warranty

SYNAPSE warrants to the original consumer/purchaser that this SYNAPSE program cassette/disk (not including the computer programs) shall be free from any defects in material or workmanship for a period of 90 days from the date of purchase. If a defect is discovered during this 90 day warranty period, and you have timely validated this warranty, SYNAPSE will repair or replace the cassette/disk at SYNAPSE's option, provided the cassette/disk and proof of purchase is delivered or mailed, postage prepaid, to SYNAPSE.

This warranty shall not apply if the cassette/disk (1) has been misused or shows signs of excessive wear, (2) has been damaged by playback equipment, or (3) if the purchaser causes or permits the cassette/disk to be serviced or modified by anyone other than SYNAPSE. Any applicable implied warranties, including warranties of merchantability and fitness, are hereby limited to 90 days from the date of purchase. Consequential or incidental damages resulting from a breach of any applicable express or implied warranties are hereby excluded.

Notice

All SYNAPSE computer programs are distributed on an "as is" basis without warranty of any kind. The entire risk as to the quality and performance of such programs is with the purchaser. Should the programs prove defective following their purchase, the purchaser, not the manufacturer, distributor, or retailer assumes the entire cost of all necessary servicing or repair.

SYNAPSE shall have no liability or responsibility to a purchaser, customer, or any other person or entity with respect to any liability, loss or damage caused or alleged to be caused directly or indirectly by computer programs sold through SYNAPSE.

The provisions of the foregoing warranty are subject to the laws of the state in which the cassette/disk is purchased. Such laws may broaden the warranty protection available to the purchase of the cassette/disk.