

# ***S-COM™ Repeater Controller Manager***

Configuration Management and Programming Software for S-COM™ Repeater Controllers

End-User Documentation

Version 2.10

© 2002 - 2014 Tom A. Wheeler

[tom.n0gsg@gmail.com](mailto:tom.n0gsg@gmail.com)

S-COM™ is a trademark of S-COM Industries, Inc.

## ***Terms Of Use***

Controller Manager is not freeware! You may use it for evaluation purposes during a trial period. The trial period begins when you first install Controller Manager (hereafter referred to as the Software) on a computer, and ends 30 days later. When the trial period is over, you may continue to use the Software, contingent upon purchase and registration of your working copy. If you choose not to register the Software, you must remove it from your computer at the end of the 30-day trial period.

By installing Controller Manager on your computer, you automatically agree to the above, as well as the following:

1. You agree not to distribute the Software in any form (including modified executable code) and by any means of reproduction (including web sites) to any third party without express written permission from us.
2. You agree not to disassemble or reverse-engineer the Software.
3. If you're using the Software as an individual, you agree not to install it on more than two machines (which may include a portable or laptop computer) without purchasing an additional license. You are licensed to manage up to two unique S-COM repeater controllers with each software license, and will purchase further licenses if you will be managing additional units.
4. Corporate users agree not to install the Software on more than a single machine without purchasing a license for all computers and S-COM repeater controllers on which the Software will be used. (Special rates are available for corporate users; please contact us directly.)

## ***Disclaimer***

The Software is provided 'as is' without warranty of any kind, either express or implied, including, but not limited to, the implied warranties of fitness for a purpose, or the warranty of non-infringement. Without limiting the foregoing, we make no warranty that the Software will meet your requirements, the service provided by the Software will be uninterrupted, timely, secure or error-free, the results that may be obtained from the use of the Software will be effective, accurate or reliable, the quality of the Software will meet your expectations, or that any errors in the Software will be corrected.

The Software and its documentation could include technical or other mistakes, inaccuracies or typographical errors. We may make changes to the Software or documentation. We assume no responsibility for errors or omissions in the Software or its documentation.

In no event shall we be liable to you or any third parties for any special, punitive, incidental, indirect or consequential damages of any kind, or any damages whatsoever, including, without limitation, those resulting from loss of use, data or profits, whether or not we have been advised of the possibility of such damages, and on any theory of liability, arising out of or in connection with the use of this Software.

The use of the Software is done at your own discretion and risk and with agreement that you will be solely responsible for any damage to your computer system or loss of data that results from such activities. No advice or information, whether oral or written, obtained by you from us or from our web site shall create any warranty for the Software.

# Contents

1.	Introduction, Installation, and Modes of Operation .....	1
2.	Programming Directly with a Serial Connection .....	8
3.	Programming by DTMF .....	12
4.	The Control Language .....	16
5.	Using XMODEM to Transmit and Receive Files .....	18
6.	Planned Features for Future Implementation .....	22

## *1.0 Introduction*

Controller Manager is a custom-designed software package to help control operators manage S-COM 7K and 7330 repeater controllers. S-COM controllers are industry-standard, high-performance units that are found in a wide variety of radio services ranging from amateur radio, police, fire, government, and many others. It is intended to be a unified environment for working with these controllers; no other programs other than the built-in "Notepad" utility of Windows are required. The program even includes a built-in "mini-terminal" that is fully integrated, eliminating the hassles of locating a terminal emulation program and sharing a serial communications COM port with it.

The software is designed to make accurate programming of these controllers an easy and repeatable task. It is designed to work with the feature set of S-COM controllers and can be used directly on a test bench, or remotely through modem connections.

Controller Manager supports robust structuring and documentation of S-COM programs through a built-in control language. The control language supports modular program construction (and reusable components), and allows symbolic programming of the controllers, making programs much easier to read, understand, and maintain.

With S-COM 7330 controllers, the software allows direct serial (RS232) programming, and allows fully error-checked uploads. Errors are clearly explained and marked for correction in the program's status window while programming is taking place, and the program provides affirmative confirmation of serial uploads.

## *1.1 Installation*

To install the program, proceed as follows:

1. Double-click the "manager.exe" file you downloaded from the web site.
2. The installer program will begin running. Controller Manager will automatically be installed onto your computer.

To remove the program from the computer, navigate to the Windows Control Panel and choose Controller Manager under "Add/Remove Programs."

### 1.2 Modes of Operation

Controller Manager has three primary modes of operation - Ready, Programming, and Mini Terminal. The user interface makes adjustments to itself based on the operating mode.

When the program is first started, it attempts to locate a useable serial communications port (COM port) on the machine, and initiates communications with a modem through that port, if possible. (If there is no modem connected, no harm is done.) The program then reports that result and waits for commands in its Ready mode, as shown in Figure 1.

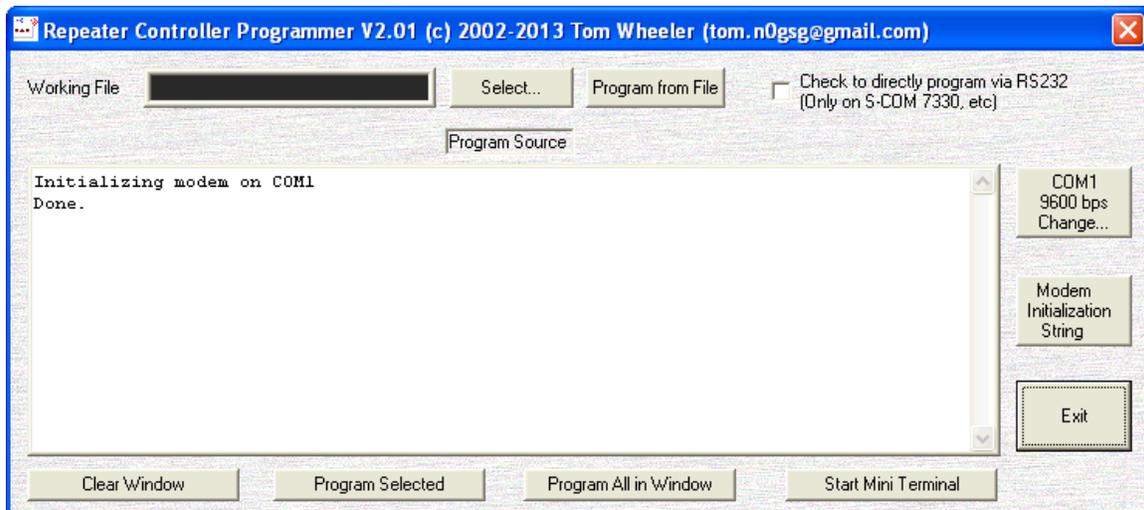


Figure 1: Ready Mode

In Ready mode, the program is prepared to accept commands; as shown in Figure 1, the program has successfully found an available communications port (COM1) and reported that status in the Program Source window.

Note: If Controller Manager is unable to locate an available COM port, the following message will appear in the Program Source window:

Failed to open serial port COMx. Can't program controller under this condition!

If you get this message, click the **Change** button to choose another port.

The software goes into Programming mode when any programming command button is clicked. These command buttons are **Program from File**, **Program Selected**, and **Program All in Window**.

Finally, Mini Terminal mode is entered by clicking **Start Mini Terminal**. This is only allowed from Ready mode.

### 1.3 Details of Controls - Ready Mode

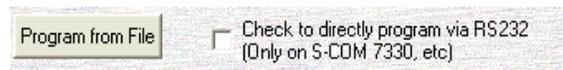
The Controller Manager program has been designed to make operation as intuitive and flexible as possible. Once the program has been installed and configured, programming a controller can be achieved in just a few mouse clicks.

#### 1.3.1 Working File



This shows the selected working file to be programming from. If the window is blank, no file has yet been selected. Clicking the **Select** button opens a file selection dialog for choosing the input file.

#### 1.3.2 Programming and Programming Mode Commands



Clicking **Program from File** initiates the programming process and switches to Programming mode. If a file has not been previously chosen, the software will prompt for an input file (just as if the **Select** button had been clicked.)

If you are programming an S-COM 7330 (or equivalent) controller through a direct RS-232 connection, check the "directly program" option to utilize direct programming mode. You must do this before beginning programming. *Using a #direct command within a file forces the program to always use direct serial programming, regardless if this box is checked.*



You can command the Controller Manager to use the Program Source window as a temporary scratchpad when programming controllers, and program the controller from results in this window instead of from a file. This capability is useful when you're experimenting with new commands and configurations, as it allows you to instantly upload commands to the controller and evaluate their effects.

The **Program Selected** control will transmit only the highlighted (selected) text in the Program Source window to the attached S-COM controller. To use this command, you must first highlight the desired text, then click **Program Selected**.

The **Program All in Window** control transmits all data from the Program Source window to the controller.

**CAUTION:** Remember that text in the Program Source window is not saved by the program, and will be lost when the program exits. You can copy information in this window to the clipboard as in other Windows applications; highlight the text you wish to copy, and type **CTRL** - C.

### 1.3.3 COM Port and Modem Configuration Controls



The **Change...** button is used to select an alternate communication port and speed. It causes the Serial Communications Settings dialog to open, as shown in Figure 2.

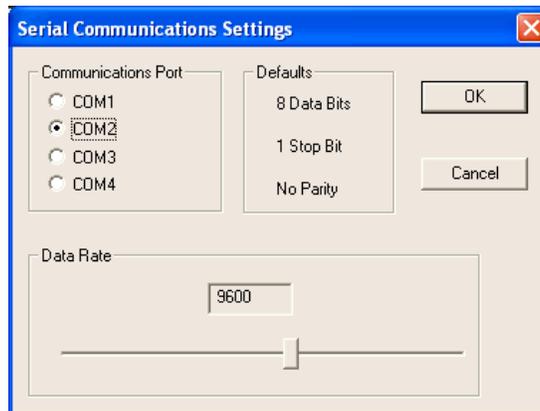


Figure 2: Serial Communications Settings Dialog

In the Serial Communications Settings dialog, the data rate is controlled with the slider control at bottom, and the port number (COM 1 through COM 4) is chosen by clicking one of the four radio buttons at the upper left.

#### Tips:

- All modes of operation (Ready, Programming, Mini Terminal) within Controller Manager share the settings chosen within this dialog. Changing it from any of these places changes it for all the program's remaining modes. The program remembers your settings when it closes.
- It is strongly recommended to use no data rate greater than 9600 bps for S-COM 7330 controllers, especially if this is being accomplished through a remote connection.

**IMPORTANT:** For direct serial programming, the S-COM 7330 must be configured with the same serial data rate as Controller Manager, or programming efforts will fail. This configuration is accomplished using Mini Terminal mode and the S-COM 7330 "SBOOT" command. The default data rate for S-COM 7330 controllers is 57,600 bps.

The **Modem Initialization String** button allows you to input a text command that will be automatically sent to the modem at power up, if needed. This is useful if your modem needs additional configuration to work with an S-COM controller. This information is also remembered by Controller Manager, so it will be sent to the modem each time the program starts.

Controller Manager uses Hayes™ AT-commands to generate DTMF ("touch tone") programming signals using an attached smartmodem. Some non-Hayes modems may require additional commands for them to work properly in this application; in particular, the modem must not time out and disconnect for a span of many minutes while the program uses it to send DTMF data to a controller.

Refer to your modem's documentation to determine if you will need a modem initialization string for your particular modem.

#### 1.4 Mini Terminal Mode

Start Mini Terminal

Clicking the **Start Mini Terminal** button launches the built-in Mini Terminal. The Mini Terminal is a simplified teletype emulator that provides easy access to devices connected to serial ports on the computer. It appears as shown in Figure 3.

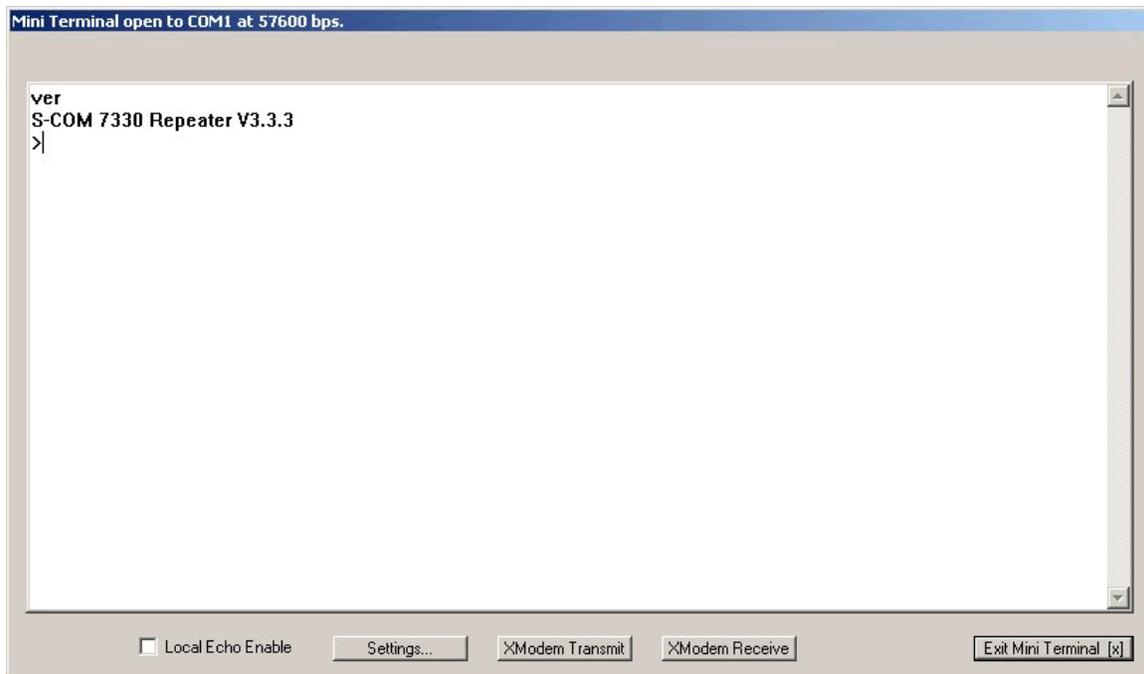


Figure 3: Mini Terminal Mode

Once the Mini Terminal is open, you may directly type commands into its window. These commands will be immediately sent to any connected device (such as an S-COM controller); the window also shows device responses to your commands.

If you're connected to an S-COM 7330 controller, you should be able to press the **Return** or **Enter** key and see a ">" response from the controller.

**Tip:** Issuing a "ver" command to the controller is an excellent way of determining if the connection is good. Pressing the reset button on the back of the controller should also result in a response showing the controller's firmware version.

A rectangular button with a light beige background and a thin black border. The text "Settings..." is centered on the button in a small, black, sans-serif font.

Controller Manager must be set to the same communication speed as the attached S-COM controller. Clicking the **Settings...** button accesses the Serial Communication Settings dialog, as described in Section 1.3.3.

If you are not able to get a response from the controller, check the following:

- The correct data rate is selected using the Serial Communication Settings dialog. If the controller is brand new, the data rate should be 57,600 bps.
- You are using a cable of the correct type between your computer's serial port and the S-COM 7330. This cable should be a "straight" cable with no crossover or "null modem" inserted, as the 7330 acts as an RS-232 DCE (Data Circuit terminating Equipment).
- If you are using a USB-to-serial converter, make sure the correct driver is installed.

**IMPORTANT:** At the time of this writing, it is known that the Windows 7 device drivers provided for USB-to-serial converters utilizing Prolific Technology chip sets are unreliable and will result in data loss. If you are using Windows 7, choose a USB to serial converter using a different chip set, such as FTDI.

If you are wondering whether the serial interface is working at all, you can temporarily set up a loopback connection to check it. This procedure works for both 9 pin and 25 pin serial ports:

- Disconnect the remote device (7330, etc) from the serial port.
- Jumper pins 2 and 3 (and no others) of the exposed male serial connection coming from your computer (or its USB-to-serial adapter).
- Type something into the mini terminal. You should see the exact same information echoing back to the screen.

If you see no information at all, something is wrong with your serial port (other than the communication settings).

Local Echo Enable

You may wish to use the Mini Terminal with devices that don't echo data back as it is sent. Checking the Local Echo Enable box causes locally typed characters to appear in the Mini Terminal window.

The S-COM 7330 always echoes data as it is sent, so you normally won't need to check this box.

If you see double characters while you're typing to a device, uncheck the box.

## 2.0 Programming Directly with a Serial Connection

Programming an S-COM 7330 controller using a direct serial (RS232) connection is an easy and reliable process. The steps to doing this are as follows:

- Generate a source code file (or files) with the configuration information in S-COM control language, usually with Notepad.
- Connect to an S-COM 7330 controller, and ensure that communications is working between your computer and the controller.
- Transmit the file information to the controller by entering Programming Mode.
- Observe the results, and correct any errors in your configuration flagged by Controller Manager.

**IMPORTANT:** Direct programming only works with the 7330 controller. Earlier S-COM controllers must be programmed using DTMF mode, as shown in Section 3.

### 2.1 Generating a Source Code File

Source code files contain the commands and configuration information that will be uploaded to the controller. These files must be in plain text format; the Windows Notepad tool is made to work with plain text files.

Controller Manager provides a control language that augments the S-COM language, providing additional functionality and simplifying operations. Section 4 details this control language. The basic rules of the language are as follows:

- Any line that begins with the pound sign (#) is a reserved command for the Controller Manager program. All reserved commands are executed only while programs are being uploaded and otherwise have no effect on the controller.
- Comment lines start with a semicolon (;) and are ignored by both Controller Manager and the S-COM controller. Comments can also be placed at the end of each line by adding a semicolon.

We recommend that you comment your programs well. It's much easier to maintain a configuration that is properly documented!

A sample configuration file is shown in Figure 4. You should refer to the S-COM programming manual for further details about generating configuration information.

```

; WOERH - 145.290 MHz -- test file
;
; 05/19/2010: ADDED dynamic ID. If PL is on, repeater announces tone frequency with
;             voice ID message. If PL off, frequency is not announced.
;
; 06/13/2010: Removed the 6 am event to turn off PL/CTCSS. The repeater will now be on
;             full-time CTCSS access.
;
#define COPW 99                ;INITIAL COPW VALUE FROM FACTORY
#define DESIRED_COPW 1234     ;WHAT COPW WILL BE CHANGED TO ONCE LOADED

;////////////////////
;ERASE ALL MACROS
;////////////////////

COPW 22 00 *

;-----
;TORNADO WARNING
;Macro A3
;-----
COPW 20 00A3 COPW 15 9981 9960 0000 0523 0568 0000 0523 0568 *

;-----
;CANCEL TAIL WX-ALERT MESSAGE (TURNS OFF WEATHER ALERT)
;Macro 29
;-----
COPW 20 0029 COPW 3113 *

;-----
;TIME & DATE
;Macro C8
;-----
COPW 20 00C8 COPW 15 9981 9960 0000 0377 9820 9821 9829 9828 *

```

*Figure 4: A Sample Configuration*

Note the use of the **#define** directive to conveniently define symbolic constants, such as the control password **COPW**. Once a constant is defined in this manner, Controller Manager will automatically substitute it into expressions as they occur in subsequent lines. For example, the line "COPW 20 00A3 COPW . . ." will be sent to the controller as "99 20 00A3 99" since **COPW** is defined as the constant 99.

## 2.2 Connecting to an S-COM Controller

Connect a straight serial cable between the RS232 (COM) port on your computer and the console port of the S-COM 7330, and verify that the serial communication speeds of both devices match using the Mini Terminal mode.

It is recommended that you set both systems to 9600 bps for the most reliable transfers. You can set the S-COM 7330 data rate by entering its SBOOT utility at the command prompt.

SBOOT is a built-in utility of the S-COM 7330 controller that lets you manage the controller's overall configuration.

Once you've configured the controller to your liking, exit SBOOT using the Boot from Flash command, then exit Mini Terminal by closing the Mini Terminal window.

**IMPORTANT:** The S-COM controller must running normally, *not* within SBOOT mode, for programming to be completed in the following steps.

### 2.3 Transmitting Files in Programming Mode

Before attempting to transmit the file, make sure the "Directly Program" box is checked. Otherwise, Controller Manager will assume it is connected through a modem, and will attempt to send the file by DTMF.

You can also include a **#direct** command anywhere within the file; this will cause Controller Manager to always transmit the file directly to the controller, regardless of whether the "Directly Program" box is checked.

To begin programming the controller from a known file, click **Program from File** to begin the process. Controller Manager will prompt you for the file if one is not already selected. The programming mode screen will then open as shown in Figure 5, and data transmission will immediately begin.

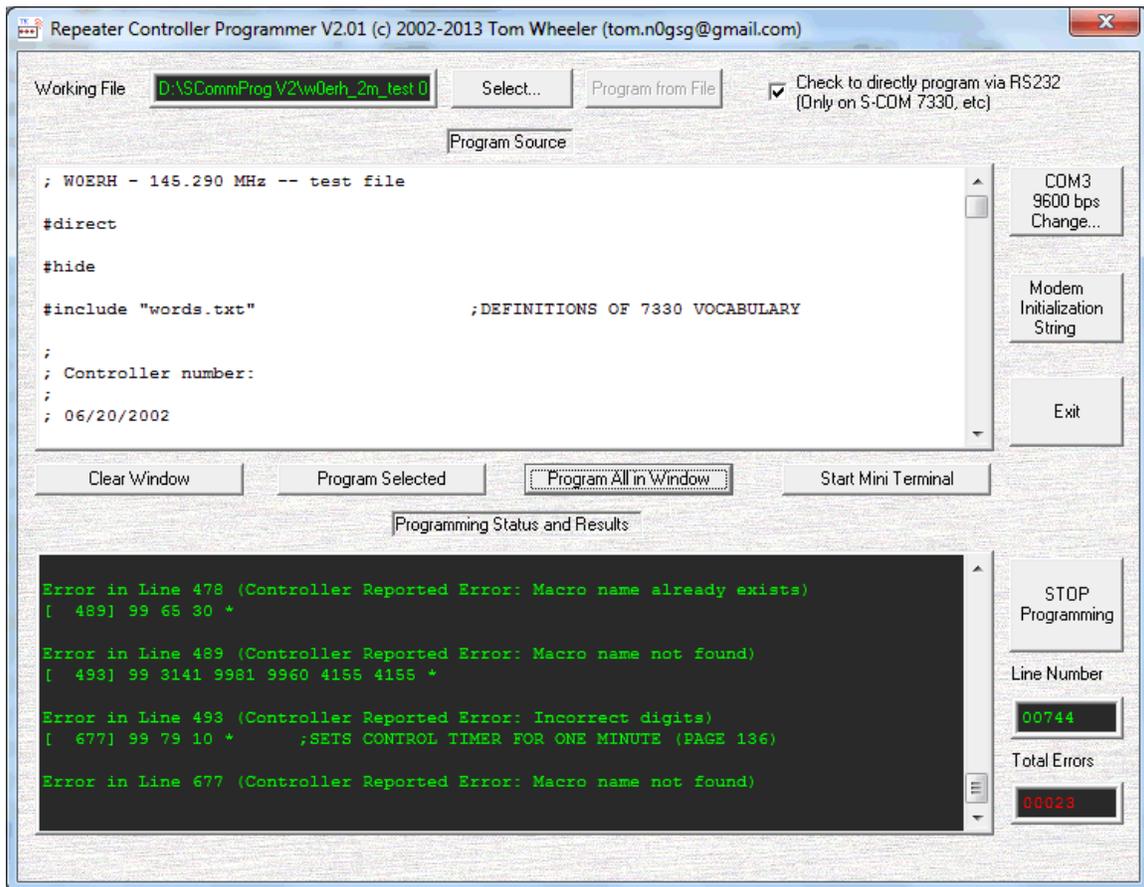


Figure 5: Programming Mode Screen

In programming mode, Controller Manager helps you keep track of the process in two ways:

- The real-time flow of data to the controller is displayed in the Programming Status and Results window.
- The Line Number and Total Errors counters track the line number in the file being processed, and the total number of errors accumulated.

Any errors that occur are relayed in two ways:

- The first error always causes Controller Manager to pause, display the error, and ask whether it should continue attempting to program the controller. If you opt to continue programming, subsequent errors will not cause Controller Manager to pause.
- Error messages accompany the output in the Programming Status and Results window. They tell you what is wrong with the line of code from the controller's point of view.

Note that Controller Manager does not pre-screen commands being sent to the controller in any way.

Controller Manager also alerts you if the attached controller fails to respond. This could be due to a bad cable, wrong COM port number selection, controller in wrong mode (or not powered on), incorrect data rate settings, and so forth.

**IMPORTANT:** Errors that occur in carrying out directives (such as **#include**) are always fatal and stop the programming process immediately. You must correct these errors to continue the programming process.

You can stop programming at any time by clicking the **STOP Programming** button. Controller Manager responds by immediately stopping the programming session in progress, and returns to Ready mode. The Programming Status and Results window will reflect that the programming process was halted before it could be completed.

Successful programming of a controller is reflected by a "Programming complete with 00000 errors" message in the Programming Status and Results window.

Since Controller Manager checks line-by-line for correct transmission of data to the controller, this message indicates that the controller now has an exact copy of the desired configuration loaded into its memory, and is ready to be placed into service.

### 3.0 Programming by DTMF

All S-COM controllers can be programmed by using a sequence of DTMF tones. For controllers prior to the 7330, this is the only supported method of programming.

DTMF programming is accomplished in one of two ways:

- A Hayes-compatible smartmodem is connected directly to the S-COM controller using a fixture similar to that of Figure 6. This is the typical setup used for bench programming a controller prior to installation.
- For remotely-located controllers, a telephone connection is used along with the **#session** directive. When a **#session** directive is included in the configuration source code, Controller Manager automatically uses a Hayes-compatible smartmodem to dial the remote controller, enter the lines of code, and terminate the connection.

There is no error detection in DTMF mode. The controller responds with "OK" in Morse code after each line is entered, but Controller Manager has no built-in way of recognizing that signal.

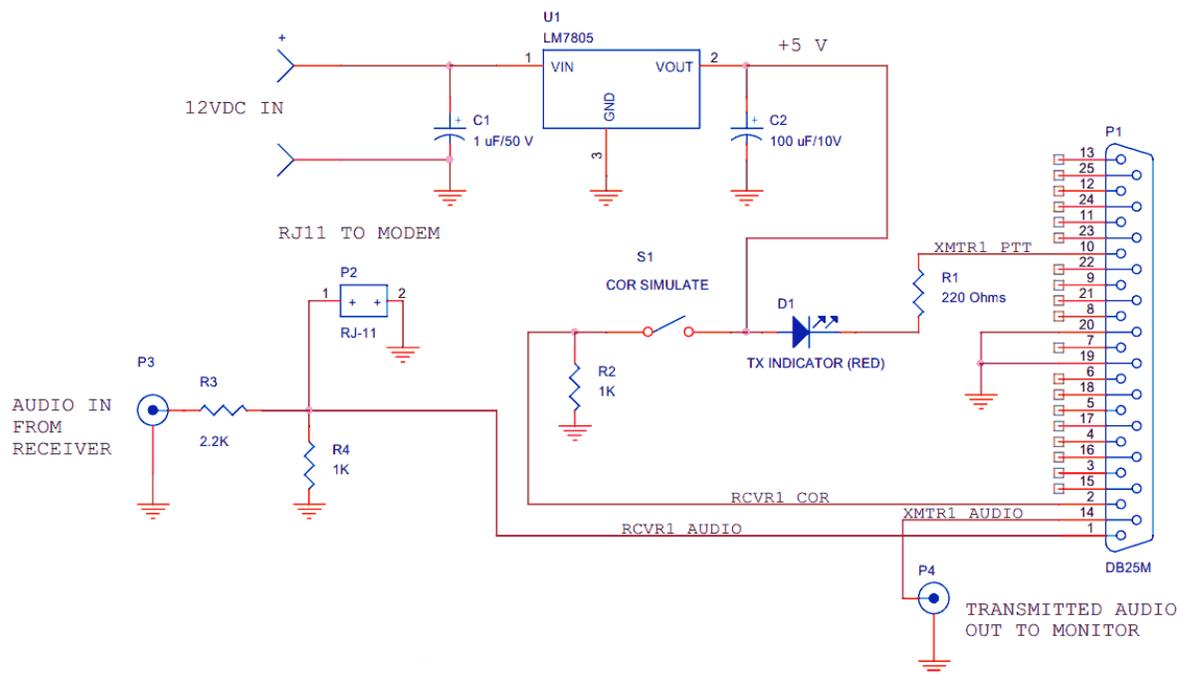


Figure 6: Custom Fixture for Bench Programming S-COM Controllers in DTMF Mode

**IMPORTANT:** Since there's no error detection in this mode, you should monitor the programming process in the event a problem arises.

The steps to programming controllers using DTMF are as follows:

- Generate a source code file (or files) with the configuration information in S-COM control language, usually with Notepad, as described in Section 2.1.
- Connect a Hayes-compatible smartmodem to your computer, and interface it directly with the controller using the fixture of Figure 6, or through a phone line if remotely programming.
- Transmit the file information to the controller by entering Programming Mode.
- Observe the results, and note any error conditions that occur.

### 3.1 Transmitting a File in Programming Mode

Before attempting to transmit a file by DTMF, make sure the "Directly Program" box is *unchecked*. Otherwise, Controller Manager will assume it is directly connected to a 7330 or compatible controller and will attempt sending the data by ASCII text.

If the controller to be programmed is remotely located, you must include a **#session <control-password> <phone-number>** command at the beginning of the file; this will cause Controller Manager to dial the specified phone number before proceeding with the data transfer. A typical **#session** command looks like this:

```
#session 99 9136488888
```

The value 99 is the control password, and 913-648-8888 is the telephone number of the remote controller. *There must be no spaces in the phone number.*

To begin programming the controller from a known file, click Program from File to begin the process. Controller Manager will prompt you for the file if one is not already selected. The programming mode screen will then open as shown in Figure 7, and data transmission will immediately begin.

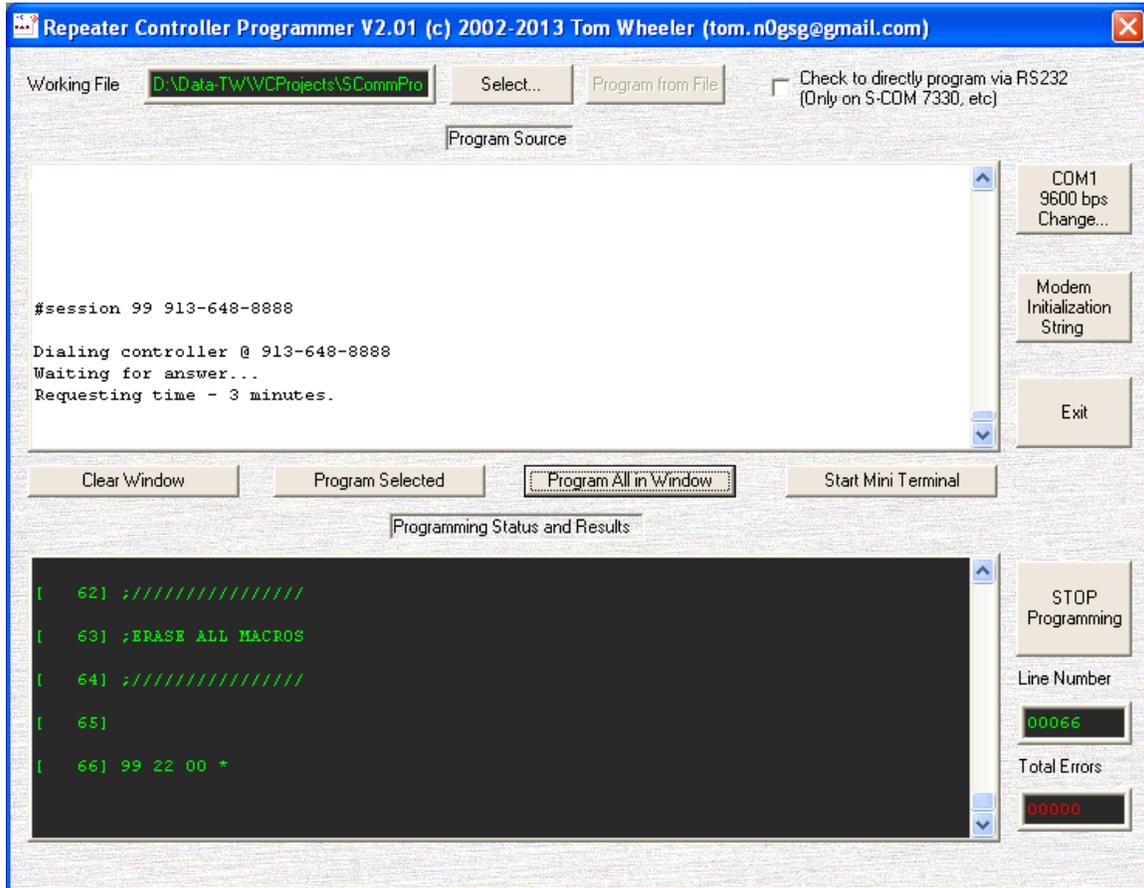


Figure 7: Programming Mode Screen in DTMF Mode

Note that the Program Source window will reflect the outgoing call status in DTMF Programming Mode. In Figure 7, controller at telephone number 913-648-8888 has been dialed, and time has been allotted for the S-COM controller to answer.

**IMPORTANT:** There's no error detection in this mode. The total error count will remain zero throughout the process, even if something severely goes wrong (such as a broken connection). Monitoring is important to assure success.

Even though there is no error detection in this mode, Controller Manager does intelligently manage the operation:

- The connection time is monitored, and since S-COM controllers will automatically disconnect after two minutes, Controller Manager automatically inserts requests for more controller time whenever they are needed.

- Transmission of data is open loop, but Controller Manager analyzes the content of each line of code to be transmitted and anticipates how much time is needed to send it to the controller, and how much time the controller will need to process that line and respond with "OK." This prevents an out-of-control "overrun" of data into the controller.

To stop a transmission in progress, click the **STOP Programming** button. Controller Manager will not stop sending immediately when this button is clicked:

- The pending line of code will be completely transmitted to the controller, to avoid an incomplete entry into the controller's memory.
- If a remote session is in progress, Controller Manager will terminate the telephone connection by forcing the modem to hang up.

**IMPORTANT:** It could take up to 45 seconds for all pending operations to complete once **STOP Programming** is clicked. Be patient; the program is not misbehaving! It is simply working to find a clean stopping point.

#### 4.0 Control Language

Controller Manager understands a limited set of commands that are executed whenever a controller is being actively programmed. These commands constitute the control language.

The first character on each line of code determines how Controller Manager interprets it:

- A semicolon (;) tells Controller Manager that the line is a comment. The remainder of the line is ignored.
- A pound sign (#) marks a **directive**. Directives perform specific actions during the programming process, such as automatically setting the time on the attached repeater controller.

Any other lines are interpreted as data that is to be sent to the controller. Before data lines are transmitted, they are scanned for defined *symbols* generated with the **#define** directive. The appropriate substitutions for the symbols are inserted in each line before it's transmitted.

A common use for symbols is to define operating constants such as the control password in a global fashion. This way the control password can be changed on subsequent runs by updating only the definition, instead of the many times it might appear in the source code.

#### 4.1 Controller Manager Directives

The following are the supported directives as of this version of the product. The following conventions are assumed:

- [ *dv* ] marks identify optional portions of a command. *dv* is the default value that's assumed if the optional portion is omitted.
- < *element* > identifies a required operand or option on the line. *element* describes the type of information that belongs.
- Quotes are not generally required for command line arguments, however, they should be used consistently. They must be used in places where the argument contains spaces, such as certain file or pathnames.

Directive and Syntax	Purpose	Example Usage
#define <const> <value>	Defines a symbolic value. <const> must not already be defined, or an error occurs. Case-insensitive.	#define PWORD 1315 #define MYFILE words.txt
#direct	Forces direct ASCII programming of a controller to take place, regardless of selection status on control panel.	#direct  (May be any place within code)
#hide	Causes Controller Manager to not print subsequent source lines during the programming process. Scope is only for the current file.	#hide ; ;These lines will not show ;during programming

Directive and Syntax	Purpose	Example Usage
#include <file>	Causes statements from the specified <file> to be included in the stream of information going to the controller. Useful for including tables of common definitions, or commonly-used routines.	#include words.txt #include "c:\my file.txt"
#session <pw> <phone>	Initiates a remote DTMF session with a controller. Should be near the beginning of the code.	#session 99 913-648-8888 #session PWORD PHONE_NUMBER
#show	Causes Controller Manager to begin showing program statements again during the programming process.	#show
#timeset <pw>	Automatically synchronizes the attached controller's time with the clock of the personal computer. The year, month, day, and time-of-day information are translated into the S-COM equivalents, and sent to the controller. The control password <pw> is mandatory, as the controller must have it to execute this command. Note: Although this command will work on the 7330 controllers, we recommend using #timeset7330 for 7330 controllers, as this will ensure that the seconds field is synchronized.	#timeset 99 #timeset PWORD
#timeset7330 <pw>	Synchronizes the attached controller's time with the clock of the personal computer. The year, month, day, and time-of-day information are translated into the S-COM equivalents, and sent to the controller. The control password <pw> is mandatory, as the controller must have it to execute this command. Note: This command will NOT work with 7k or earlier controllers.	#timeset7330 99 #timeset7330 PWORD

### 5.0 Using XMODEM to Transfer Files

Controller Manager has a built-in utility for sending and receiving files using both the checksum and Cyclic Redundancy Check (CRC) versions of the XMODEM protocol. The transmit utility detects the client's initial request for data and automatically switches to the correct mode; the receiver always attempts to use the CRC version, which is much more effective at trapping and correcting transmission errors. SCOM controllers always use the CRC protocol mode.

#### 5.1 Transmitting a File to the SCOM 7330 Controller

To send a file to the controller, perform the following steps:

1. Start the Mini Terminal and issue the SBOOT command to the SCOM 7330 repeater controller. The controller will respond by displaying the SBOOT menu.
2. From the SBOOT menu, enter "L" to tell the controller that you wish to upload a file to its internal flash memory.
3. From the next menu, choose which file in the flash memory you wish to write to.

**WARNING:** Make sure you are uploading to the correct flash memory file and that you are sending appropriate information. Uploading the wrong file (or a damaged file) to flash can disable your repeater controller, which may require factory intervention to correct.

4. Answer "Y" to confirm that you do indeed wish to write to the flash file you just specified.
5. Click the **Xmodem Transmit** button. Controller Manager will prompt you for the file to be sent as shown in Figure 8. Highlight the file you wish to send and click **Open**.

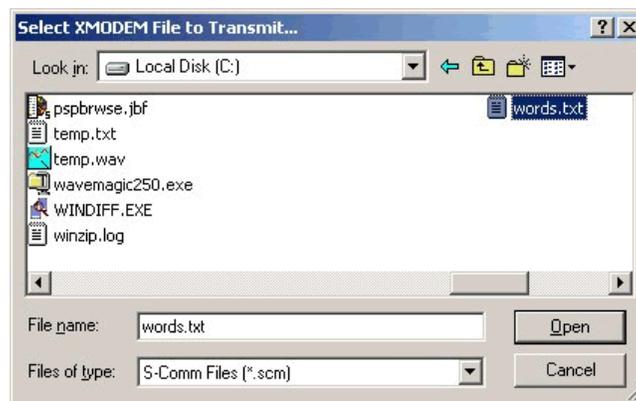


Figure 8: Choosing the Transmit File

6. Once you've chosen the file, Controller Manager will immediately begin transmitting the file to the repeater controller as shown in Figure 9. Controller Manager continually updates you regarding the status of the transfer.

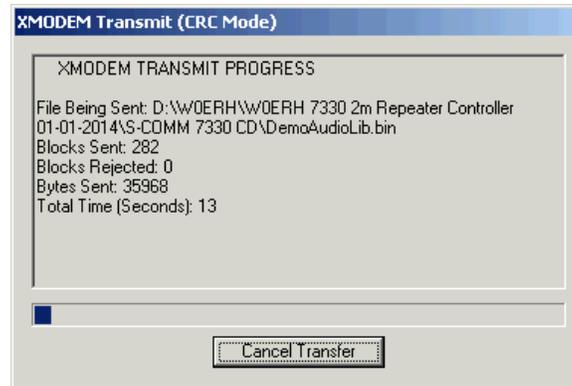


Figure 9: XMODEM Transmission Status Window

The status information can be interpreted as follows:

- **File Being Sent:** This is a confirmation of the file you are sending.
- **Blocks Sent:** XMODEM divides files into 128-byte blocks for transmission. This is a count of the number of blocks sent. It counts both "good" (accepted) and "bad" (rejected) blocks.
- **Blocks Rejected:** An XMODEM receiver tests the validity of each 128-byte block of data as it's received, and requests retransmission of bad blocks. This value shows how many blocks the repeater controller has rejected. The value should be zero (0) ideally, or at least a small number (less than 10). A high number indicates a problem with cabling or other transmission media.
- **Bytes Sent:** This is the total number of verified data bytes sent to the controller.
- **Total Time:** This indicates the time in seconds the transfer has utilized.
- **The progress bar** shows the percentage completion of the transfer. Large files can take several minutes to move.

**TIP:** To make the process go as quickly as possible, use the highest data rate available (57,600 bps if possible) to do XMODEM transfers. Since XMODEM is error-checked, there is no advantage in using lower data rates.

Note that at any time during the process you can press the **Cancel Transfer** button to stop the upload. If you do cancel a transfer, you may need to manually reset the SCOM 7330 controller - - although theoretically the repeater controller should time out after a couple of minutes and return to SBOOT on its own.

7. When the transfer completes, you will see a "successfully completed" message in the Status Window. Click the **Close** button to close the Status Window and return to the Mini Terminal. Once you're back in the Mini Terminal, you may need to press the **ENTER** key once to see the SBOOT prompt.

### 5.2 Receiving a File by XMODEM

At present the SCOM 7330 firmware does not support XMODEM transmission of controller data. It's anticipated that this feature will be implemented, and Controller Manager will be able to receive controller data when the feature is made available.

**TIP:** The XMODEM receive feature can also be used to transfer files between two computers of any type. Use a crossover cable between the two machines and use the maximum available data rate to speed the transfers.

To receive a file, do the following:

1. Command the remote system (such as the SCOM 7330 controller) to send a file. This is done within SBOOT for SCOM controllers.
2. Click the **Xmodem Receive** button from within the Mini Terminal. A dialog for choosing the receive file will appear (similar to Figure 8).

Choose the location and filename for the file you wish to receive and click the **Open** button.

3. The transfer will begin automatically. The Status Window will again open to let you monitor the progress of the file transfer.
4. When the download is complete, a success message will appear in the Status Window. Click the **Close** button to return to the Mini Terminal.

### 5.3 Troubleshooting

XMODEM is a very reliable and robust transfer protocol. However, there are several common problems you may run into.

**Transfer does not start:** Make sure that your physical cabling is correct (no crossover is needed for the SCOM 7330 on the default console port) and that the data rates are set identically on both hosts. It normally requires one or two seconds for most XMODEM transfers to begin.

**XMODEM uploads terminate with status code 1:** There is a problem with the data in the file you are trying to send and the SCOM 7330 (or other host) is rejecting the file with a CANCEL command. Make sure that you're sending the right file to the right flash memory location.

**High number of rejected blocks:** There is an issue with your cabling (or modem connection) and data is not reliably reaching the target system. Check that there are no intermittent connections. For modem connections, you may want to reduce the data rate and try again.

#### 5.4 Recovering a "Broken" SCOM 7330 Controller

It is possible to make the SCOM 7330 repeater controller inoperative by sending wrong information to it during the upload process (especially if you're uploading the repeater configuration). There is a way to force the controllers into SBOOT, however. From there you may be able to re-send the damaged information and recover the controller.

**WARNING:** If you do not fully understand the instructions in this section, contact the factory for assistance. You may physically damage the controller if you incorrectly perform some of these steps. If your controller is working normally, do not attempt this procedure.

##### 5.4.1 Forcing the Controller into SBOOT

The following are the general and undocumented steps for doing this:

1. Remove power from the repeater controller.
2. Locate the AUX1 and AUX2 jumper pins. Place a shorting jumper on both of these. This will force the controller to start an internal boot loader program when it's turned on.

**TIP:** You can temporarily borrow jumpers from one of the three ports for this - - just remember to put them back when you're done.

3. Start the Mini Terminal mode and apply power to the repeater controller.

You may see a command prompt (">") on the screen. If you don't, press the **ENTER** key a couple of times.

**TIP:** If the controller is really scrambled, it may not remember the correct data rate. If you're not getting a response, try changing the data rate value in Controller Manager.

If you get no response in this step, you should contact SCOM for more assistance.

4. Enter "b" at the command prompt to launch the SBOOT utility.
5. Once SBOOT is running, you can follow the XMODEM transmit process to re-send the damaged files.
6. Once you've recovered the controller, remove power, then remove the jumpers from the AUX1 and AUX2 pins.

### *6.0 Planned Features for Future Implementation*

This revision of the software has been created in response to the release of the S-COM 7330 controller, which unfortunately is not accompanied by software to program it. The 7330 controller is a work in progress, and as of this writing many of the SBOOT menu options look tantalizing, but are actually not yet implemented.

As more SBOOT options are rolled out by S-COM, this software will evolve to support them. Future enhancements will include:

- Support for direct upload and download of image data from a controller.

- Support for customized audio on S-COM controllers.

Controller Manager will provide a unified interface for working with customer audio on these controllers, borrowing from Analog Magic's audio processing technology. Integrated audio tools will make it much easier to generate and manage custom speech and audio.

- Internet support for remote programming and management.

Such support would allow access to any supported S-COM controller through a secured web interface.

- Graphical user interface for configuring controllers.

The next generation software could include a GUI interface for programming, which would make configuring these controllers as easy as clicking choices on a form.