

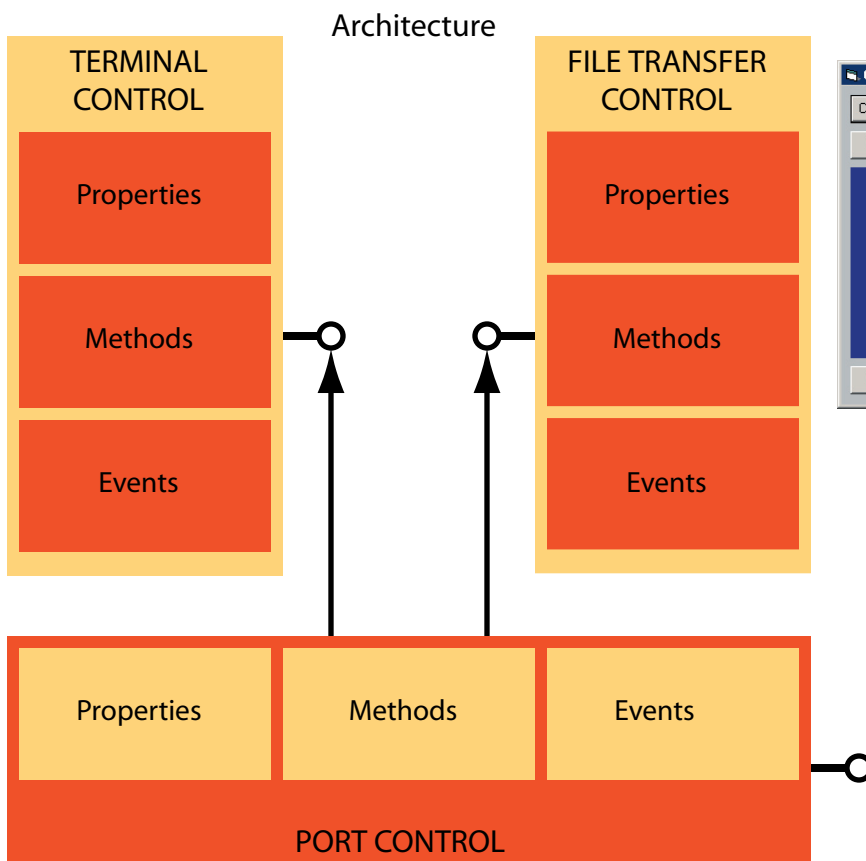


# Greenleaf CommX 1.1

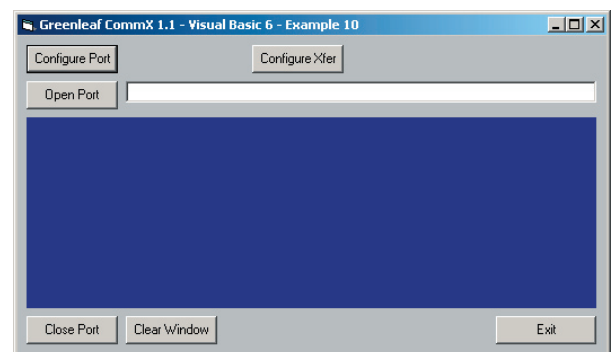
## Supports Windows with Port, Terminal & File Transfer Controls

### Easily the most Feature Rich...

- ❑ Greenleaf CommX is a set of three controls—Port, Terminal, and File Transfer—that offers rich, event driven, and robust serial comm capability for your Windows applications.
- ❑ CommX is a dual-mode ATL based ActiveX control. It is NOT a warmed-over C library like some products in the market.
- ❑ You get all the ease of drag-drop controls onto your form or dialog face plus full design-time programmability. You can encapsulate CommX controls into your DLL, Windows Service, or GUI application using Visual C++, Delphi, or Visual Basic.
- ❑ Unlimited number of Ports, Terminals and/or File Transfers simultaneously.
- ❑ Use in single or multithreaded apps; also has IDispatch interface and is MFC compatible.
- ❑ Property pages offer one way to configure each control; you can call Configure() for programmatic use or, for GUI-less projects, set each property with code.
- ❑ Optional progress displays for file transfer status.
- ❑ Events for each control: Port events individually enableable—add handlers for what you need.
- ❑ Baud Rates to 921.6 Kbps, given hardware support.
- ❑ Supports Visual C++ 5,6,7.1, Borland C++Builder 3,4,5, and 6, Visual Basic 5,6, and 7.1, Visual FoxPro 5,6, Borland Delphi 3-6, PowerBuilder 6, 6.5, and 7, and VBA for Microsoft Office application development.
- ❑ CommX enhanced apps restore state of serial port(s) when app ends.
- ❑ Compilable example programs included, all with source, for each supported development environment.
- ❑ Tutorials aid in adding CommX components to your toolbox.
- ❑ You can define any number of trigger strings for the Terminal, each tied to a routine in your application.
- ❑ You can define macros (strings to send) tied to a key.
- ❑ You can define any number of local commands. A command ties a key to a specific KeyEvent event.
- ❑ Includes recursive subdirectory search engine with wildcard expansion.
- ❑ Includes ZModem and ASCII file transfer with options.
- ❑ Terminal emulations include ANSI and TTY.



### Simple Example



### Languages

- Visual C++ 5,6,7.1
- Borland C++Builder 5,6
- Borland Delphi 3,4,5
- Visual Basic 5,6,7.1
- Visual C# 7.1
- Visual FoxPro 5,6
- PowerBuilder 5,6,6.5

### Platforms

- Windows 9x, Me,2000,XP Pro,2003

## Greenleaf CommX 1.1

Over 450 public methods, properties, and events, providing powerful tools for all aspects of asynchronous serial communications programming for Windows and MSDOS, including 16- and 32-bit Extended DOS. Supports all popular C++ compilers. Includes pre-built static and dynamic libraries for most compilers including debug versions. BUILD utility lets you rebuild libraries, even after changes in the source code.

### What Others Say

We get hundreds of emails from people who have tried and failed to use MComms or other control or library in their application. Usually it's because there are features missing or they need to open more ports than the competition will allow. Every one of them has been resolved by CommX. Simply put, CommX is the best of breed.

### Supports Many Kinds of Hardware

CommX supports almost any kind of serial comm device you can plug into a PC or an embedded system. If the Windows If Device Manager lists the port, CommX can use it. This is true whether the UART is physically local or remote over Ethernet, USB, or another high speed pipe.

There is one exception: a combination of hardware and software (the prime example is "WinModem") that does not properly implement a serial port.

### File Transfer Features

We're justifiably proud of our file transfer engines; they've been honed by hard use in thousands of installations, countless months of formal beta testing, and the critical court of public appeal, so that (for example) our ZModem implementation is considered by many to be the reference implementation. It turns out that public domain protocols such as ZModem are often "tweaked" by developers who want "just a little difference" and this change, not reviewed by peers, often undocumented, can get people into trouble. Consider the nightmare of maintaining a program written ten years ago using some implementation of ZModem. How do you do it? Yes, by comparison to some things such as USB, TCP/IP, etc., they're simple protocols. That's if you're not the one trying to figure out what is breaking transfers.

### Terminal Features

The terminal control in CommX is unique in many ways. You set its Port property to a Port Control and can set or clear the properties ReadingPort and ReadingKeyboard. Next you can define trigger strings that have meaning in your application; when one so defined is received, an IncomingTrigger event is fired. The definition ties the particular trigger to a value that is passed to your IncomingTrigger event handler. From there, it's up to you what your application does.

The terminal control is also the part of CommX that maintains

a list of bindings of keystrokes to commands; when reading the keyboard, the list of command keys is searched and, if the current keystroke is tied to a command, a KeyEvent event is fired and passes a value that you can use to call a user routine from within your KeyEvent handler.

### Documentation

Most software is designed, tested, then the documentation is written. CommX's development and documentation teams worked as an organic entity so that there are tips in the docs that a tech writer would not think to add—and code that writers suggested.

We ship a Windows Help manual with CommX; in addition there are numerous examples for each platform—source code that you can compile. It turns out that many beginner users find our examples do what they want, so they have little work to do.

### Commitment & Growth

Greenleaf CommLib was the first serial communications library for the PC. Released in 1984 on a dare by a customer of one of our earlier products.

CommX was another first for Greenleaf; developed later, in the heyday of ActiveX. CommX. It was an instant success.

Check out our other serial comm products: Greenleaf CommLib, Comm++, and Greenleaf ViewComm Async and ViewComm Viewer (data monitors and script driven protocol analyzers). In addition, our ViewComm Ethernet is a full-featured Ethernet Protocol Analyzer.

# Some of CommX's Properties, Events, and Methods

## Port Control Properties

BaudRate	Determines baud rate for port
BreakDetect	Indicates break signal received
ByteCount	Holds current count of bytes received
Cd	Holds current state of CD signal
Cts	Holds current state of CTS signal
DataBits	Sets / Holds number of data bits per character
Dsr	Holds current state of DSR
Dtr	Sets / Holds current state of DTR
DtrDsr	Sets the current DSR/DTR handshaking mode
FramingError	Indicates incoming framing error
IssueBreakDetectEvent	Enables break detect event fire
IssueCdChangedEvent	Enables firing event on CD change
IssueCtsChageEvent	Enables firing event on CTS change
IssueDsrChangeEvent	Enables firing event on DSR change
IssueLineStatusErrorEvent	Enables event on any line error
IssueRingEvent	Enables firing event on RI signal
IssueRxTriggerEvent	Enables firing IncomingFlagChar event
IssueTxBufferEmptyEvent	Event when Tx Buffer is impty
OverrunError	An Rx Overrun error was detected
Parity	Sets / Holds parity mode for port
ParityError	Parity error on incoming data
PortHandle	Holds the Handle for the port.
PortNumber	Holds port number, e.g. 1 = COM1
Ri	Holds current state of RI line
Rts	Holds current state of FTS line
RtsCts	Sets / Holds state of RTS/CTS handshaking
RxTriggerChar	Character code to fire IncomingFlagChar event
ShowModemPorts	Property sheet can show ports with modems
ShowMousePorts	Property sheet can show ports with mice
ShowUnattached Ports	Property sheet show unattached ports
SpaceFreeInRxBuffer	Available bytes in Rx buffer
SpaceFreeInTxBuffer	Available bytes in Tx buffer
SpaceUsedInRxBuffer	Bytes used in Rx buffer
SpaceUsedInTxBuffer	Bytes used in Tx buffer
Status	Holds port status (bitfirl)
StatusString	String description of last error on port
StopBits	Get / Set number of stop bits
XonXoff	Enable / disable XON/XOFF handshaking

## Port Control Methods

ClearLineStatus	Clear all line status indications
ClearRxBuffer	Clear the receive data buffer
ClearTxBuffer	Clear the transmit data buffer
Close	Close the port & restore prior conditions
Configure	Brings up Port property sheets at runtime
Open	Open the serial port
PeekByte	Peek at the next byte in the Rx buffer
EventMode	Controls marshaling mode
ReadByte	Read the next byte in the Rx buffer
ReadByteBuffer	Read n bytes from the Rx buffer
ReadByteBufferTimed	Read n bytes, maimum timeout
ReadByteTimed	Read one byte, maximum timeout
ReadString	Read string (BSTR) from port Rx buffer
ReadStringTimed	Read string (BSTR) with timeout
SendBreak	Send a break signal
WriteByte	Send one byte to Tx buffer
WriteByteBuffer	Send n bytes to Tx fuffer
WriteByteBufferTimed	Send n bytes to Tx buffer with timeout
WriteByteTimed	Send one byte with timeout
WriteString	Send a string to Tx buffer
WriteStringTimed	Send a string to Tx buffer with timeout

## Port Control Events

CdChange	CD signal transition occurred
CtsChange	CTS signal transition occurred
DsrChange	DSR signal transition occurred
HandleChange	Port handle has been changed on open port
IncomingBreak	Break signal detected incoming
IncomingData	One or more characters received in the Rx buffer
IncomingFlagChar	Received code = RxTriggerChar
LineStatusError	One or more errors have occurred
ParameterChange	One or more properties changed on open port
Ring	Incoming RI (Ring) signal detected
TransmitterBufferEmpty	Tx buffer and UART became empty

## File Transfer Properties

AsciiReceiveTimeout	Set # ms for Rx ASCII transfer timeout
AsciiStripIncomingLF	Strips linefeed from incoming ASCII files
AsciiTransmitLineDelay	Accommodate other end needing delay
CreateProgressWindow	Enable runtime file transfer progress window
CurrentByteCount	Bytes received so far in current file
CurrentErrorCount	Line errors in current file
CurrentFileCount	Number of files sent or received so far
CurrentFileLength	Number of bytes in file being processed
CurrentFileName	Name of current file being transferred
DownloadDirectory	Pathname destination for incoming files
FileNameDelimiters	One or more delimiters between filenames
IncludeSubdirectories	Enables subdirectory recursion when sending
MaxErrors	Maximum errors before abort transfer
MonitorCarrier	Enables abort if Carrier Detect is lost
Port	Name of port associated with file transfer engine
ProcessDialogParent	Handle for parent window for progress dialog
Protocol	Selects the protocol for file transfers
ReceivedFileName	Name of current file being received
SendFileNames	Delimited string listing file(s) to be send
StripPathOnReceive	Enable storing all incoming files in same directory
StripPathOnSend	Don't include path portion of files being sent
ZModemCrashRecovery	Enables crash recovery on ZModem transfers
ZModemOverwriteMode	Select option for file overwite on ZModem
ZModemWindowSize	Set sliding window size for ZModem transfers

## File Transfer Control Methods

AbortXfer	Abort file trnsfer process
ReceiveFiles	Initiate process of listening & receiving file(s)
SendFiles	Initiate process of sending one or more files

## File Transfer Control Events

Message	Event to display a string in a message box
ReceiveFileComplete	Fired at end of receiving (each) file
ReceiveFileStart	Fired when incoming file begins to be read
SendFileComplete	Fired when completing (each) file in Tx process
SendFileStart	Fired when starting to send (each) file
StatusUpdate	Fired when status has changed (may require action)

## Terminal Control Properties

ActualColumns	Number of actual columns in emulation area
ActualRows	Number of rows in emulation area
BackgroundColor	Set background color for terminal
Blink	Set / Get state of blink attribute
Bold	Set / Set state of bold attribute
BorderColor	Set terminal area border color
BorderXPixels	Width of vertical border in pixels

## Terminal Control Properties (Cont'd)

BorderYPixels	Height of horizontal border in pixels
Cols	Number of text columns in terminal
CursorColumn	Current text cursor x-position
CursorRow	Current text cursor y-position
DesiredColumns	Number of text columns requested
DesiredRows	Number of text rows requested
Emulation	Emulation type, e.g. ANSI, TTY
Font	Font size specification
ForegroundColor	Foreground text color
Invisible	Set / Clear invisible text attribute
Port	Associated communications Port
ReadingKeyboard	Set / Clear reading keyboard state
REadingPort	Set / Clear reading from port state
ReverseVideo	Set / Clear reverse video attribute
Rows	Set / Get number of rows on terminal area
TestMode	When set, characters typed appear directly on terminal
Wrap	Wrap words at right

## Terminal Control Methods

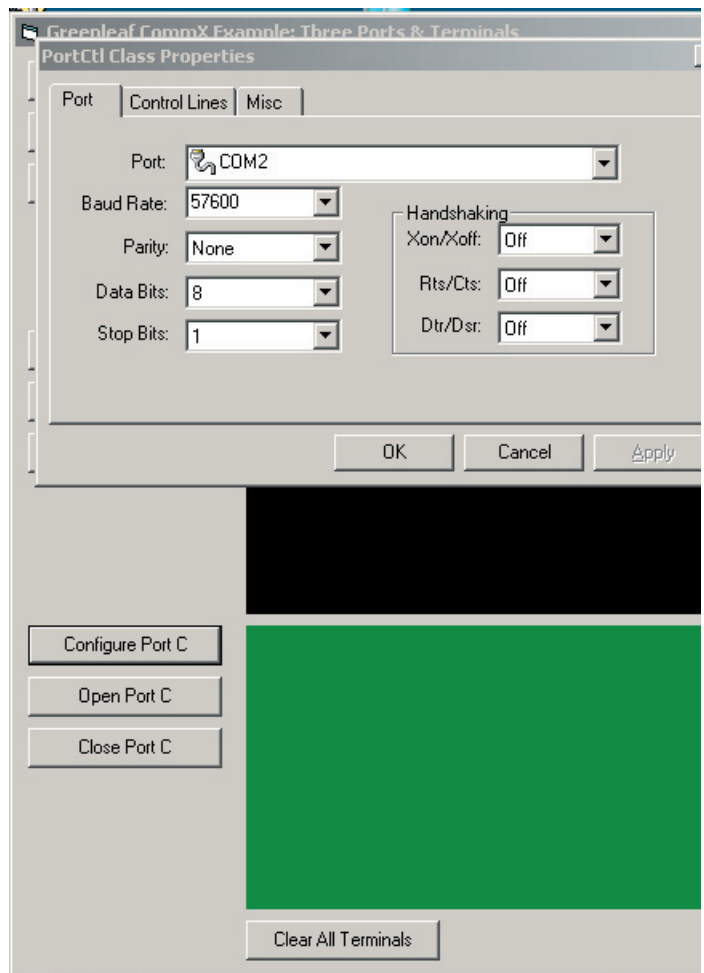
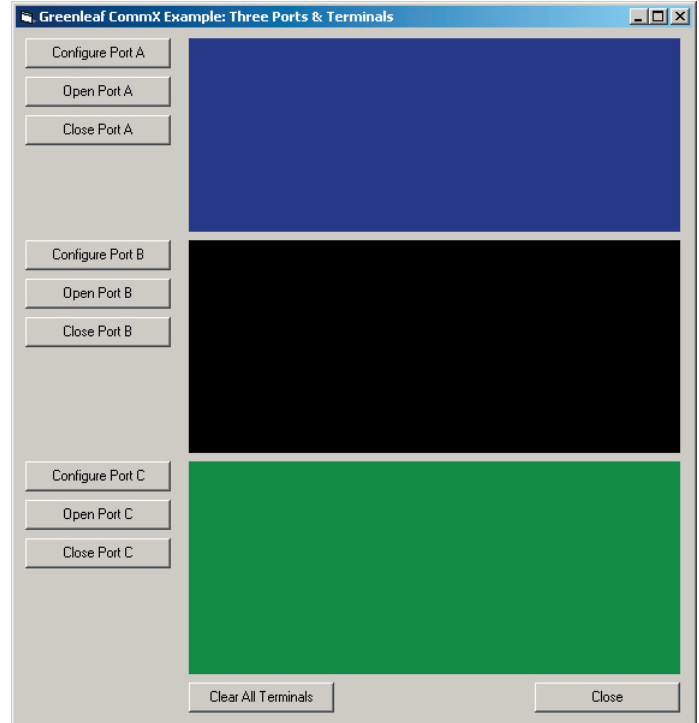
AddKeyBMacro	Add (define) keyboard macro (BSTR argument)
AddKeyCommand	Add a keyboard command definition
AddKeyMacro	Add keyboard macro string definition (C string)
AddTriggerBString	Add a trigger string definition (BSTR argument)
AddTriggerString	Add a trigger string definition (C string)
AddVirtualBMacro	Add a virtual key macro definition (BSTR)
AddVirtualKeyCommand	Add virtual key command (C string)
AddVirtualKeyMacro	Add virtual key macro (C string)
ClearToEol	Clear terminal to end of line
ClearWindow	Clear entire terminal area
Configure	Show property pages at runtime
DeleteBorder	Delete the border(s) on terminal display
RemoveKeyCommand	Remove keyboard command binding
RemoveKeyMacro	Remove keyboard macro binding
RemoveTriggerString	Remove trigger string from list
RemoveVirtualKeyCommand	Remove virtual key macro from list
SetCursorPosition	Set relative text cursor position on terminal
SetDesiredSize	Set desired number of rows and columns
SetNormal	Clear font attributes (e.g. bold, blink)
SetToDesiredSize	Set emulator window to match size properties
TestPattern	Create and display terminal test pattern

## Terminal Control Events

IncomingTrigger	Rx character sequence matches existing trigger
KeyEvent	A key (or Virtual Key) struck that is bound to a command
NewSize	Parameter change occurred that forces terminal resize

### Three Terminals Example showing Port Configure Dialog

### Three Terminals Example: Time to build---Two minutes





**Call Greenleaf Software Now:**

**1-877-218-7347**

**9550 Forest Lane  
Suite 309  
Dallas, TX 75243**

**+1 214-349-3005  
+1 214-349-3650 (FAX)**

**[www.GreenleafSoft.com](http://www.GreenleafSoft.com)**

**[sales@GreenleafSoft.com](mailto:sales@GreenleafSoft.com)**

- ❑ Greenleaf CommX supports Microsoft Visual C++ 5, 6, and 7.1, Borland C++Builder 3...6, Delphi 3-6, Visual FoxPro 5,6, PowerBulder 6, 6.5, and VBA for Microsoft Office 2000, XP, and 2003 Application development.
- ❑ Develop applications, DLLs, etc. for Windows 9x, Me, 2000, XP Pro, and Server 2003.
- ❑ Many compilable example programs for each supported compiler included, all with source for Windows. Includes tutorials showing how to import CommX into your toolbox and project.
- ❑ Coded for optimum granularity and speed.
- ❑ Product includes Windows Help documentation
- ❑ FREE Web and email based support with telephone support as needed.
- ❑ We accept MasterCard, VISA, American Express. Terms available upon credit approval.
- ❑ Did we forget to say THERE ARE NO ROYALTIES? Well, there are none.

**We do the grunt work...**