RMB/Win Extender

"A Rocky Mountain Basic for Windows TCP/ActiveX Extension"

Description

Why

The popular **R**ocky **M**ountain **B**asic language used with MS/Windows© is known as *'The Premier Language for Instrumentation'*. Today, usage of Test and Measurement involves related tasks like data exchange or presentation, user GUI, exchange with other software tools as well for R&D as for process testing, peripheral specialized device control, ...
Even if RMB for Windows is able to work with the operating system and enable graphical user control through Basic Plus, we must recognize that such 'accessory treatments' can be very time-consuming and take T&M engineers away from their first purpose that remains to be creating a well-designed and precise measurement control system.

The best way to keep a complete system simple enough and really reliable is to use separate software parts, as are used to create hardware with different specialized devices.

The purpose of the TCP/ActiveX extender for RMB/Win is to enable collaboration between T&M applications and other MS/Windows© software using a convenient, strong and easy to use tool.

How it works

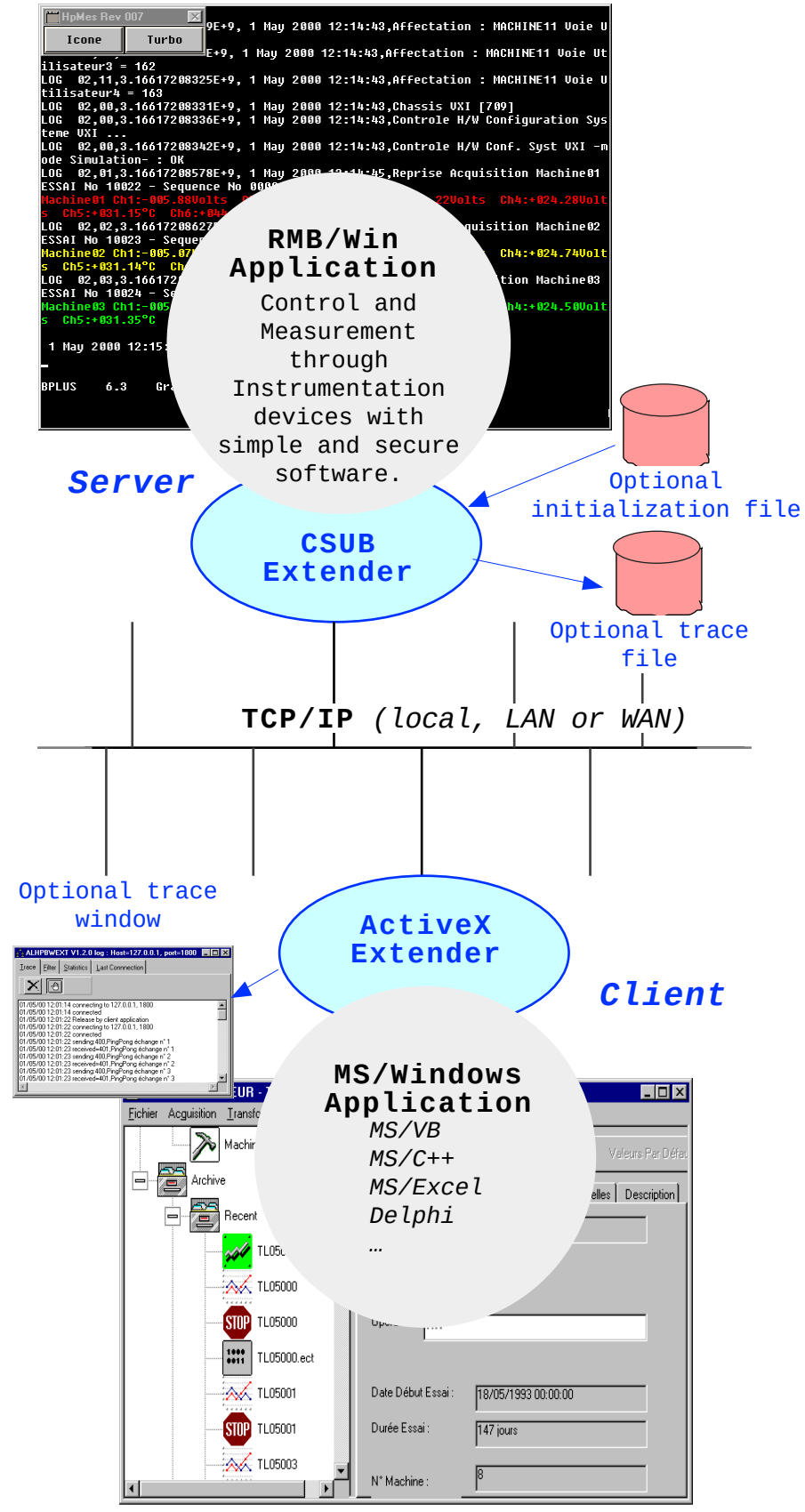
The RMB/Win Extender works with simple components used on both sides, using a strong standardized communication media:

-CSUB	on RMB side
-ActiveX	on other MS/Windows© application
-TCP/IP	to communicate between

CSUB — This is easy to use for RMB developers, and it can be used with older RMB/Win revisions as well as the latest one.

ActiveX — This type of component is largely usable from recent MS/Windows© programming tools; it is easy to use and self-documented.

TCP/IP — This is the most widely used communication protocol used for LAN and WAN. Using an asynchronous message scheme, this protocol is self-protected against communications errors.


RMB/Win Application
Control and Measurement through Instrumentation devices with simple and secure software.
Server
CSUB Extender
Optional initialization file
Optional trace file
TCP/IP (local, LAN or WAN)
Optional trace window
ActiveX Extender
Client
MS/Windows Application
MS/VB
MS/C++
MS/Excel
Delphi
...
Fichier
Acquisition
Archive
Recent
Description
Date Début Essai :
18/05/1993 00:00:00
Durée Essai :
147 jours
N° Machine :
8

Now that we know the basis, let's have a look at how RMB/Win Extender works.

•Overview and configuration

-In TCP communication there is always a 'server' that 'listens' for the 'client' connection requests. After the communication link is established, data can be sent/received by both sides.

-With RmbWin Extender the server side is always the RMB/Win application. The RMB/Win server side offers a socket port to which any client can connect. The server application can read status information, so it knows when a client is connected, and if there are messages in the reception queue (a kind of mailbox). It can read received messages when it desires to do so, and of course, it can send messages to the connected client.

-The client must know the server's name (ie: <My-computer>) or it's IP address (ie: <195.250.20.132>). The client must also know the socket port offered by the server. When the client requests a TCP connection to the server, response using a standardized protocol is returned. If the connection is accepted, the client can receive asynchronous data from the server and send data to it.

•Data Exchange

-RMB/Win Extender is designed to be simple but versatile in usage. The content that is exchanged is strongly dependent on each application and must not be limited by the communication tool. At the same time, there is usually a higher level of information that is needed (ie: remote application status, application error level, remote start/stop or continue,...).

-Data Exchange structure:

<TYPE> <STRING DATA>

<TYPE> is a two byte INTEGER in the range 1 to 19999.
This value is application dependent; it is not controlled or used by RMB/Win Extender for internal purposes. It allows a high and fast split-level from user written applications.

<STRING DATA> may contain any ASCII character and must be in the range of 0 (null string) to 19999 character length.
As any character may be used, a convenient separator can be chosen depending upon your application, to enable a second parser level scheme if needed.

•Communication Level Error Reporting

-RMB/Win Depending on user choice, an RMB ERROR may be generated when a message can't be sent. Otherwise, communication errors are reported as a returned variable on the connection status request.
Complementary error information and timestamp are written in a trace file.
RMB Error numbering is in the range 5000 to 5019.

-ActiveX An event is signaled when there is a communication error.

•Configuration Tools

-RMB/Win

MS/Windows Initialization File :
Usage of a standardized MS/Windows© initialization file is provided to help share of configuration parameters with other applications from inside your RmbWin application. You may use it or not as you find it convenient for your purpose.

Logging Service :
A log file scheme is provided on RmbWin server application side. Adjustment of trace reporting and logging is provided through an initialisation file whose name and usage are configured by user RmbWin application.

-ActiveX

It's easy to register configuration data in an ActiveX application. We provide the source code of a VB class to access private .INI files with the sample VB programs.

•Offered Functions

-RMB/Win

Communication

TcpListen	Turns server on (reset state and accept incoming connection request).
TcpStop	Turns server off.
TcpSend	Send message to connected client.
TcpReceive	Get message from connected client (FIFO sequential read).
TcpGetstate	Get connection status and number of pending messages (received and waiting in FIFO queue for application read request).
TcpSendError	*For debug and test purpose only, send a badly formatted message.*

Initialization Files

CnfFileIs	Select the initialization file to use.
CnfReadString	Get a parameter value in the selected section of .ini file.
CnfWriteString	Write a parameter value in the selected section of .ini file.

-ActiveX

Methods

Connect	Request TCP connection to server.
Disconnect	Close active connection to server.
SendData	Send message to server.
IsConnected	Report current connection status: true if message can be sent/received.
IsOpened	Report current socket status: true if *IsConnected* and during connection establishment.
SendDataWithError	*For debug and test purpose only, enable the send of badly formatted message.*

Events

ConnectionStateChanged	On connection state change.
MsgReceived	On message received from server.
Error	On communication error.

•Specifications

-Operating System : MS/Windows© 95, 98, NT 4.0 or 2000

-Supported programming tools :

-HP BASIC for Windows© V6.33 or Higher
-TransEra HT BASIC for Windows© V6.xx or Higher
-MicroSoft©/Visual Basic V5.x or Higher
Example provided in Rev 6.0.
-MicroSoft©/Visual Basic for Applications V5.0 or Higher
This MS/VBA is available with Office 97 and Office 2000
-MicroSoft©/Visual C++ V5.0 or Higher
Example provided in Rev 6.0.
-Borland Delphi© V4.0 or Higher
-…

-Socket Range Number Any Valid Socket Number (not used by other application, a value higher than 1024 is usually a good choice)

-Message Types Integer from 1 to 19999

-Message Size String from 0 to 19999 Characters

-Communication Speed Depending upon physical layer and/or system configuration and usage

•Contact & Information

For more information on RmbWin Extender, contact Bourbaky :

BOURBAKY Test & Mesure
BP 53
13, Rue des Alpes
07302 TOURNON Cedex - FRANCE
Tel (Nat.) : 04 75 07 81 21 Tel (Int.) : +33 4 75 07 81 21
Fax (Nat.) : 04 75 07 29 74 Fax (Int.) : +33 4 75 07 29 74

web : <http://www.bourbaky.com>

e-mail : <info@bourbaky.com>