

SavoyHsms ActiveX Control
User Guide

1 Revision History

Version	Date	Name	Description
1.00	Jul, 31 st , 2009	Hikaru Okada	Created as new document
1.00a	Aug, 22 nd , 2009	Hikaru Okada	Splitted into separate document, since number of pages became large.
1.00b	Aug, 30 th , 2009	Hikaru Okada	Mended some incorrect description.

2 Table of Contents

1	Revision History	2
2	Table of Contents	3
3	SavoyHsms	4
3.1	Properties	5
3.1.1	Appearance	5
3.1.2	BorderStyle	6
3.1.3	Connect	7
3.1.4	IniFileName	8
3.1.5	IniSection	9
3.1.6	IPAddress	10
3.1.7	Log	11
3.1.8	LogBakCount	12
3.1.9	LogFileNames	13
3.1.10	LogSize	14
3.1.11	LogVerbose	15
3.1.12	MyHostName	16
3.1.13	MyPortNumber	17
3.1.14	PortNumber	18
3.1.15	Selected	19
3.1.16	Server	20
3.1.17	T3	21
3.1.18	T5	22
3.1.19	T6	23
3.1.20	T7	24
3.1.21	T8	25
3.2	Methods	26
3.2.1	AboutBox	26
3.2.2	Disconnect	27
3.2.3	LoadIniFile	28
3.2.4	Send	29
3.2.5	Setup	30
3.2.6	ToIPAddress	33
3.3	Events	34
3.3.1	Connected	34
3.3.2	Disconnected	35
3.3.3	Problem	36
3.3.4	Received	38
3.3.5	SelChanged	39

3 SavoyHsms

SavoyHsms control is an assistant product to develop SEMI E37 (HSMS) compliant communication application software. SavoyHsms control can be used for either equipment side development or host side development. Usually SavoyHsms control will be used with SavoySecsII control.

Properties

Name	Description
Appearance	Gets or sets the value that determines the appearance of a SavoyHsms control.
BorderStyle	Gets or sets whether the SavoyHsms control has a border.
Connect	Gets or sets the HSMS connection status.
IniFileName	Gets or sets INI file name to read/write settings.
IniSection	Gets or sets section name in INI file to read/write settings.
IPAddress	Gets or sets the IP address of passive entity computer for HSMS connection.
Log	Gets or sets whether logging is enabled.
LogBakCount	Gets or sets the number of back-up file for logging.
LogFileName	Get or sets the log file name.
LogSize	Gets or sets the log file size in kilobyte.
LogVerbose	Get or sets whether logging for detailed part is enabled.
MyHostName	Gets the host name of local computer.
MyPortNumber	Gets or sets local portnumber for TCP/IP connection.
PortNumber	Gets or sets the port number for TCP/IP connection.
Selected	Gets or sets the selection status of HSMS communication.
Server	Gets or sets the entity type.
T3	Gets or sets the T3 time out for HSMS in milliseconds.
T5	Gets or sets the T5 time out for HSMS in milliseconds.
T6	Gets or sets the T6 time out for HSMS in milliseconds.
T7	Gets or sets the T7 time out for HSMS in milliseconds.
T8	Gets or sets the T8 time out for HSMS in milliseconds.

Methods

Name	Description
AboutBox	Opens version information dialog box on the screen.
Disconnect	Disconnects specified TCP/IP connection.
LoadIniFile	Loads settings from INI file and initialize properties.
Send	Send specified message.
Setup	Opens setup dialog box on the screen.
ToIPAddress	Converts specified computer name to IP address.

Events

Name	Description
Connected	HSMS connection has been established.
Disconnected	HSMS connection has been disconnected.
Problem	Error has occurred.
Received	SavoyHsms control received message.
SelChanged	Another HSMS connection has been selected on the screen.

3.1 Properties

3.1.1 Appearance

Gets or sets the value that determines the appearance of a SavoyHsms control.

Value	Description
0	Flat
1	Etched

Syntax

Visual Basic 6.0
Appearance As Integer

Visual C++ 6.0
short GetAppearance() void SetAppearance(short)

Example

Visual Basic 6.0
.Appearance = 0 ' flat .Appearance = 1 ' sunken

Visual C++ 6.0
m_ctrl.SetAppearance(0); // flat m_ctrl.SetAppearance(1); // sunken

Remarks

Persistent property.

See Also

3.1.2 BorderStyle

Gets or sets whether the SavoyHsms control has a border.

Value	Description
0	No border
1	Fixed single border

Syntax

Visual Basic 6.0

```
BorderStyle As Integer
```

Visual C++ 6.0

```
short GetBorderStyle()  
void SetBorderStyle(short)
```

Example

Visual Basic 6.0

```
.BorderStyle = 0 ' no border  
.BorderStyle = 1 ' with border
```

Visual C++ 6.0

```
m_ctrl.SetBorderStyle(0); // no border  
m_ctrl.SetBorderStyle(1); // with border
```

Remarks

Persistent property.

See Also

3.1.3 Connect

Gets or sets the HSMS connection status. If Connect property is set to true where Server property is false (client), SavoyHsms control will attempt to establish connection. If connection has not been established, Connect property will be set to false.

If Connect property is set to true where Server property is true (server), SavoyHsms control will attempt to open server port and start listening to prepare for future incoming connection. At this time connection has not been established until client connected server. Therefore, even no connection was made, Connect property would be True, when server port was opened.

If user sets False to Connect property, existing connection will be disconnected. User doesn't have to worry about connection to make sure it is disconnected when application is closing. SavoyHsms control will take care disconnection automatically. When Server property was set to True and Connected property is set to False, all existing connections with clients will be disconnected.

Syntax

Visual Basic 6.0

Connect As Boolean

Visual C++ 6.0

```
BOOL GetConnect()
void SetConnect(BOOL)
```

Example

Visual Basic 6.0

```
.IPAddress = "hsms_server"
.PortNumber = 5000
.MyPortNumber = 0
.Server = False
.Connect = True ' Open
If Not .Connect then
    MsgBox "Cannot connect to server!"
End If
```

Visual C++ 6.0

```
m_ctrl.SetIPAddress("hsms_server");
m_ctrl.SetPortNumber(5000);
m_ctrl.SetLocalPortNumber(0);
m_ctrl.SetServer(false);
m_ctrl.SetConnect(true); // Open
if(!m_ctrl.GetConnect())
    MessageBox("Cannot connect to server!");
```

Remarks

If Connect property on server side was set to False, connection would be disconnected. Therefore, Connect property on client side would also be set to False.

See Also

3.1.4 IniFileName

Gets or sets INI file name to read/write settings. If INI file name is either full path name or containing relative reference of folder name, INI file will be created and read in such location. Otherwise, INI file will be created in Windows OS system folder. For this reason, it is highly recommended using with folder name. If current directory is the location, add “.” at the beginning.

Either “/” (slash) or “\” (back slash) can be used for separator of folder name.

Syntax

Visual Basic 6.0

```
IniFileName As String
```

Visual C++ 6.0

```
CString GetIniFileName()  
void SetIniFileName(LPCTSTR)
```

Example

Visual Basic 6.0

```
.IniFileName = “./Savoy.ini”
```

Visual C++ 6.0

```
m_ctrl.SetIniFileName(“./Savoy.ini”);
```

Remarks

Persistent property.

See Also

3.1.5 IniSection

Gets or sets section name in INI file to read/write settings. If multiple SavoyHsms control were embedded in one project, it is possible to share same INI file by changing section name.

Default setting is "SavoyHsms".

Syntax

Visual Basic 6.0

```
IniSection As String
```

Visual C++ 6.0

```
CString GetIniSection()  
void SetIniSection(LPCTSTR)
```

Example

Visual Basic 6.0

```
.IniSection = "Host" ' [Host] section
```

Visual C++ 6.0

```
m_ctrl.IniSection("Host"); // [Host] section
```

Remarks

Persistent property.

See Also

3.1.6 IPAddress

Gets or sets the IP address of passive entity computer for HSMS connection. IPAddress property will be ignored if the Server property is set to true, because server listens incoming connection.

When connecting local computer (same computer), use "127.0.0.1" or "" (empty) string.

It is possible to use computer name instead of IP address.

Syntax

Visual Basic 6.0

```
IPAddress As String
```

Visual C++ 6.0

```
CString GetIPAddress()  
void SetIPAddress(LPCTSTR)
```

Example

Visual Basic 6.0

```
.IPAddress = ""  
.PortNumber = 5000  
.LocalPortNumber = 0  
.Server = False  
.Connect = True ' Open  
If Not .Connect then  
    MsgBox "Cannot connect to server!"  
End If
```

Visual C++ 6.0

```
m_ctrl.SetIPAddress("");  
m_ctrl.SetPortNumber(5000);  
m_ctrl.SetLocalPortNumber(0);  
m_ctrl.SetServer(false);  
m_ctrl.SetConnect(true); // Open  
if(!m_ctrl.GetConnect())  
    MessageBox("Cannot connect to server!");
```

Remarks

Persistent property.

See Also

3.1.7 Log

Gets or sets whether logging is enabled. If this property is enabled, processing information will be written in log file. If this property is disabled, nothing will be written in log file.

Value	Description
True	Write to log file
False	Do not write log file

Syntax

Visual Basic 6.0
Log As Boolean

Visual C++ 6.0
BOOL GetLog() void SetLog(BOOL)

Example

Visual Basic 6.0
.Log = True

Visual C++ 6.0
m_ctrl.SetLog(true);

Remarks

Persistent property.

See Also

3.1.8 LogBakCount

Gets or sets the number of back-up file for logging. If actual file size of log file exceeded LogSize property, SavoyHsms control will rename the file name and make a new empty log file. If number of back-up file reached to the value of LogBakCount property, SavoyHsms control will delete oldest back-up file.

Syntax

Visual Basic 6.0

```
LogBakCount As Integer
```

Visual C++ 6.0

```
short GetLogBakCount()  
void SetLogBakCount(short)
```

Example

Visual Basic 6.0

```
.LogBakCount = 10
```

Visual C++ 6.0

```
m_ctrl.SetLogBakCount(10);
```

Remarks

Persistent property.

See Also

3.1.9 LogFileName

Get or sets the log file name. Log file will be created in current directory.

Syntax

Visual Basic 6.0

```
LogFileName As String
```

Visual C++ 6.0

```
CString GetLogFileName()  
void SetLogFileName(LPCTSTR)
```

Example

Visual Basic 6.0

```
.LogFileName = "Savoy"
```

Visual C++ 6.0

```
m_ctrl.SetLogFileName("Savoy");
```

Remarks

Persistent property.

See Also

3.1.10 LogSize

Gets or sets the log file size in kilobyte. If actual file size of log file exceeded LogSize property, SavoyHsms control will rename the file name and make a new empty log file. If number of back-up file reached to the value of LogBakCount property, SavoyHsms control will delete oldest back-up file.

Syntax

Visual Basic 6.0

```
LogSize As Long
```

Visual C++ 6.0

```
long GetLogSize()  
void SetLogSize(long)
```

Example

Visual Basic 6.0

```
.LogSize = 1024
```

Visual C++ 6.0

```
m_ctrl.SetLogSize(1024);
```

Remarks

Persistent property.

See Also

3.1.11 LogVerbose

Get or sets whether logging for detailed part is enabled.

Syntax

Visual Basic 6.0

```
LogVerbose As Boolean
```

Visual C++ 6.0

```
BOOL GetLogVerbose()  
void SetLogVerbose(BOOL)
```

Example

Visual Basic 6.0

```
.LogVerbose = False
```

Visual C++ 6.0

```
m_ctrl.SetLogVerbose(false);
```

Remarks

Persistent property.

See Also

3.1.12 MyHostName

Gets the host name of local computer.

Syntax

Visual Basic 6.0

```
MyHostName As String
```

Visual C++ 6.0

```
CString GetMyHostName()
```

Example

Visual Basic 6.0

```
.IPAddress = .MyHostName
```

Visual C++ 6.0

```
m_ctrl.SetIPAddress(m_ctrl.GetMyHostName());
```

Remarks

Read-only property.

See Also

3.1.13 MyPortNumber

Gets or sets local portnumber for TCP/IP connection. If SavoyHsms is running as active entity, this property should be 0. Or connection will not be re-established until TCP/IP level time-out.

When SavoyHsms is running as passive entity, MyPortNumber property indicates server port number for incoming client connection.

Since some port numbers are reserved by Windows OS, the number should be greater than 1024 in general. For example http server uses port number 80.

Syntax

Visual Basic 6.0

```
MyPortNumber As Long
```

Visual C++ 6.0

```
long GetMyPortNumber()  
void SetMyPortNumber(long)
```

Example

Visual Basic 6.0

```
.MyPortNumber = 0 ' Auto determine
```

Visual C++ 6.0

```
m_ctrl.SetMyPortNumber(0); // Auto determine
```

Remarks

Persistent property.

See Also

3.1.14 PortNumber

Gets or sets the port number for TCP/IP connection.

Since some port numbers are reserved by Windows OS, the number should be greater than 1024 in general. For example http server uses port number 80.

Syntax

Visual Basic 6.0

```
PortNumber As Long
```

Visual C++ 6.0

```
long GetPortNumber()  
void SetPortNumber(long)
```

Example

Visual Basic 6.0

```
.PortNumber = 5000
```

Visual C++ 6.0

```
m_ctrl.SetPortNumber(5000);
```

Remarks

Persistent property.

See Also

3.1.15 Selected

Gets or sets the selection status of HSMS communication.

Value	Description
False	HSMS connection was not selected
True	HSMS connection was selected

If SavoyHsms control received Select Request message, the connection would switch to “selected” status. If user doesn’t want to change to “selected” by some reason, set Selected property to false. When SavoyHsms received Select Response message, judge by “reason code” whether “selected” is appropriate. This property will affect T7 timeout.

Syntax

Visual Basic 6.0
Selected As Boolean

Visual C++ 6.0
BOOL GetSelected() void SetSelected(BOOL)

Example

Visual Basic 6.0
.Selected = False

Visual C++ 6.0
m_ctrl.SetSelected(false);

Remarks

See Also

3.1.16 Server

Gets or sets the entity type. If Server property is true, SavoyHsms control will run as server. If Server property is false, SavoyHsms control will run as client.

Value	Description
False	Active entity (client)
True	Passive entity (server)

Syntax

Visual Basic 6.0
Server As Boolean

Visual C++ 6.0
BOOL GetServer() void SetServer(BOOL)

Example

Visual Basic 6.0
.Server = True ' Server .Server = False ' Client

Visual C++ 6.0
m_ctrl.SetServer(true); // Server m_ctrl.SetServer(false); // Client

Remarks

Persistent property.

See Also

3.1.17 T3

Gets or sets the T3 time out for HSMS in milliseconds. Default value is 45 seconds.

Syntax

Visual Basic 6.0

```
T3 As Long
```

Visual C++ 6.0

```
long GetT3()  
void SetT3(long)
```

Example

Visual Basic 6.0

```
.T3 = 500 ' 500 milliseconds
```

Visual C++ 6.0

```
m_ctrl.SetT3(500); // 500 milliseconds
```

Remarks

Persistent property.

See Also

3.1.18 T5

Gets or sets the T5 time out in milliseconds. Default value is 10 seconds.

Syntax

Visual Basic 6.0

```
T5 As Long
```

Visual C++ 6.0

```
long GetT5()  
void SetT5(long)
```

Example

Visual Basic 6.0

```
.T3 = 5000 ' 5 seconds
```

Visual C++ 6.0

```
m_ctrl.SetT3(5000); // 5 seconds
```

Remarks

Persistent property.

See Also

3.1.19 T6

Gets or sets the T6 time out in milliseconds. Default value is 5 seconds.

Syntax

Visual Basic 6.0

```
T6 As Long
```

Visual C++ 6.0

```
long GetT6()  
void SetT6(long)
```

Example

Visual Basic 6.0

```
.T6 = 5000 ' 5 seconds
```

Visual C++ 6.0

```
m_ctrl.SetT6(5000); // 5 seconds
```

Remarks

Persistent property.

See Also

3.1.20 T7

Gets or sets the T7 time out in milliseconds. Default value is 10 seconds.

Syntax

Visual Basic 6.0

```
T7 As Long
```

Visual C++ 6.0

```
long GetT7()  
void SetT7(long)
```

Example

Visual Basic 6.0

```
.T7 = 10000 ' 10 seconds
```

Visual C++ 6.0

```
m_ctrl.SetT7(10000); // 10 seconds
```

Remarks

Persistent property.

See Also

3.1.21 T8

Gets or sets the T8 time out in milliseconds. Default value is 5 seconds.

Syntax

Visual Basic 6.0

```
T8 As Long
```

Visual C++ 6.0

```
long GetT8()  
void SetT8(long)
```

Example

Visual Basic 6.0

```
.T8 = 5000 ' 5 seconds
```

Visual C++ 6.0

```
m_ctrl.SetT8(5000); // 5 seconds
```

Remarks

Persistent property.

See Also

3.2 Methods

3.2.1 AboutBox

Opens version information dialog box on the screen.

Syntax

Visual Basic 6.0

```
Sub AboutBox()
```

Visual C++ 6.0

```
void AboutBox()
```

Return Value

None.

Example

Visual Basic 6.0

```
.AboutBox
```

Visual C++ 6.0

```
m_ctrl.AboutBox();
```

Remarks

See Also

3.2.2 Disconnect

Disconnects specified TCP/IP connection.

Syntax

Visual Basic 6.0

```
Function Disconnect(IpszIPAddress As String, IPortNumber As Long) As Boolean
```

Visual C++ 6.0

```
BOOL Disconnect(LPCTSTR IpszIPAddress, long IPortNumber)
```

Argument	Description
IpszIPAddress	IP address to be disconnected
IPortNumber	Port number to be disconnected

Return Value

Return true if disconnection was successfully done. Otherwise return false. If false was returned, IP address or port number might be incorrect. Or connection had already disconnected.

Example

Visual Basic 6.0

```
.Disconnect "", 5000
```

Visual C++ 6.0

```
m_ctrl.Disconnect("",5000);
```

Remarks

See Also

3.2.3 LoadIniFile

Loads settings from INI file and initialize properties. If loading was failed, values in persistent resource will be set.

LoadIniFile method probably is called at the beginning of application, since it retrieves saved parameters by Setup method.

Syntax

Visual Basic 6.0

```
Function LoadIniFile() As Boolean
```

Visual C++ 6.0

```
BOOL LoadIniFile()
```

Return Value

Return true if loading was successful. Otherwise return false. If false was returned, IniFileName property or IniSection property might be incorrect.

Example

Visual Basic 6.0

```
Private Sub Form_Load()  
    SwingHsms1.LoadIni  
End Sub
```

Visual C++ 6.0

```
void CxxxView::OnInitialUpdate()  
{  
    ...  
    m_ctrl.LoadIni();  
    ...  
}
```

Remarks

See Also

3.2.4 Send

Send specified message.

Syntax

Visual Basic 6.0

```
Function Send(IpszMessage As String) As Boolean
```

Visual C++ 6.0

```
BOOL Send(LPCTSTR IpszMessage)
```

Argument	Description
IpszMessage	Message to be sent

Return Value

Return true if transmission was successful. Otherwise return false.

Example

Visual Basic 6.0

```
SavoyHsms1.Send SavoySecsII1.Msg
```

Visual C++ 6.0

```
m_ctrl.Send(m_msg.GetMsg());
```

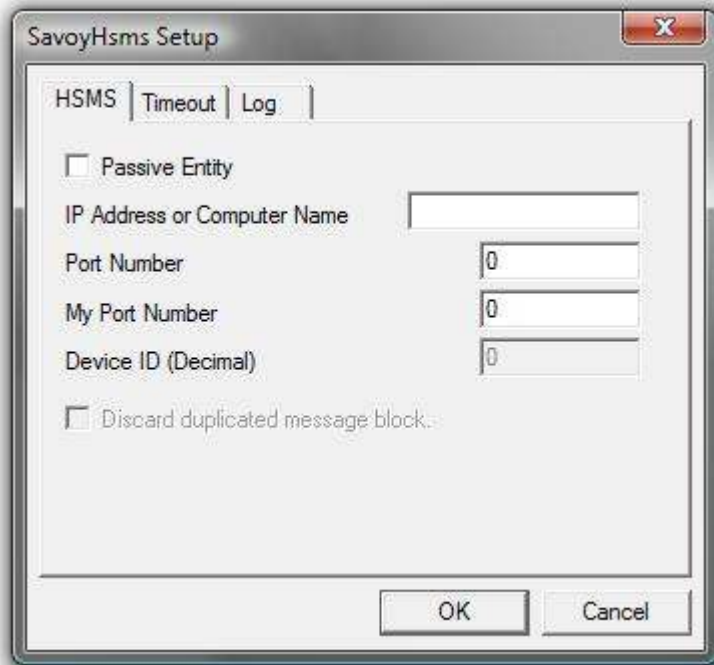
Remarks

See Also

3.2.5 Setup

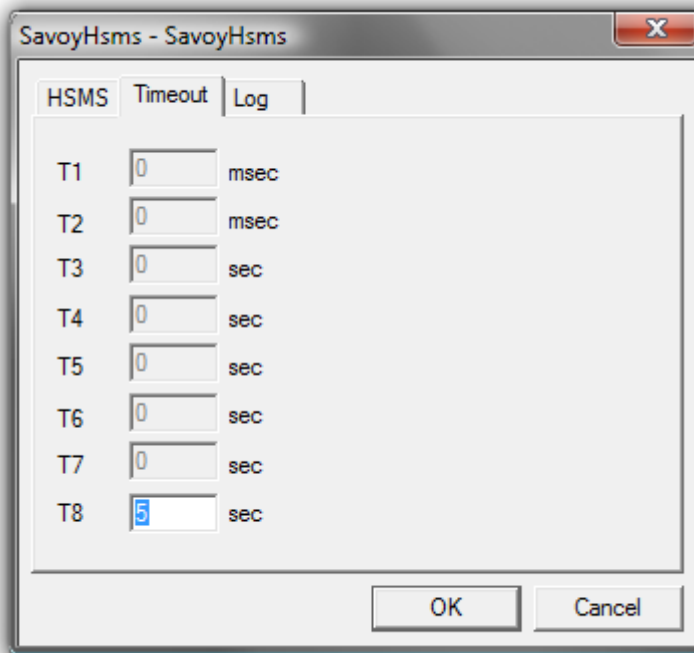
Opens setup dialog box on the screen. If user modified parameter and press OK button, data will be written in INI file.

HSMS Tab



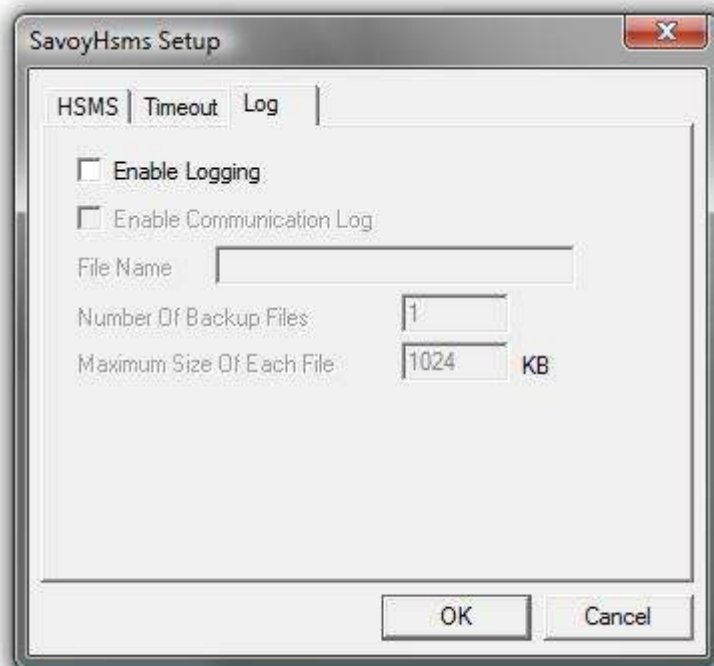
Item	Description
Passive Entity	True if passive entity. False if active entity.
IP Address or Computer Name	IP address or computer name of another computer to connect.
Port Number	Port number of another computer to connect.
My Port Number	Local port number. It is highly recommended to specify 0 for active entity.
Device ID (Decimal)	Device ID (session ID)

Timeout Tab



Item	Description
T8	T8 timer

Log Tab



Item	Description
Enable logging	True if logging is enabled.
Enable communication log	True if communication logging is enabled.
File name	Log file name.
Number of backup files	Number of backup files.
Maximum size of each file	Size of log file.

Syntax**Visual Basic 6.0**

```
Function Setup(IpszCaption As String) As Boolean
```

Visual C++ 6.0

```
BOOL Setup(LPCTSTR IpszCaption)
```

Argument	Description
IpszCaption	Caption title of dialog box. If this value is NULL or "" (empty) string, the string of IniSection property will be used for caption tile.

Return Value

If user pressed OK button and parameters were saved in INI file successfully, Setup method returns true. If user pressed Cancel button or parameter saving was failed, Setup method returns false.

Example**Visual Basic 6.0**

```
.Setup "HSMS Configuration"
```

Visual C++ 6.0

```
m_ctrl.Setup("HSMS Configuration");
```

Remarks**See Also**

3.2.6 ToIPAddress

Converts specified computer name to IP address.

Syntax

Visual Basic 6.0

```
Function ToIPAddress(lpszHostName As String) As String
```

Visual C++ 6.0

```
CString ToIPAddress(LPCTSTR lpszHostName)
```

Argument	Description
lpszHostName	Host name to be converted.

Return Value

ToIPAddress function returns IP address string if conversion was successful. Otherwise return empty string.

Example

Visual Basic 6.0

```
.IPAddress = .ToIPAddress("")
```

Visual C++ 6.0

```
m_ctrl.SetIPAddress(m_ctrl.ToIPAddress(""));
```

Remarks

See Also

3.3 Events

3.3.1 Connected

Notifies that HSMS connection has been established.

If passive entity, there is no connection established until active entity will connect.

Syntax

Visual Basic 6.0

```
Event Connected(IpszIPAddress As String, IPortNumber As Long)
```

Visual C++ 6.0

```
void OnConnected(LPCTSTR IpszIPAddress, long IPortNumber)
```

Argument	Description
IpszIPAddress	IP address
IPortNumber	Port number

Example

Visual Basic 6.0

```
Text1.Text = "Connected - " + IpszIPAddress + " [" + Format$(IPortNumber) + "]"
```

Visual C++ 6.0

```
TRACE("Connected - %s [%d]",IpszIPAddress,IPortNumber);
```

Remarks

See Also

3.3.2 Disconnected

Notifies that HSMS connection has been disconnected.

Syntax

Visual Basic 6.0

```
Event Disconnected(lpszIPAddress As String, IPortNumber As Long)
```

Visual C++ 6.0

```
void OnDisconnected(LPCTSTR lpszIPAddress, long IPortNumber)
```

Argument	Description
lpszIPAddress	IP address
IPortNumber	Port number

Example

Visual Basic 6.0

```
Text1.Text = "Disconnected - " + lpszIPAddress + " [" + Format$(IPortNumber) + "]"
```

Visual C++ 6.0

```
TRACE("Disconnected - %s [%d]",lpszIPAddress,IPortNumber);
```

Remarks

See Also

3.3.3 Problem

Notifies that error has occurred.

Syntax

Visual Basic 6.0
Event Problem(IpszIPAddress As String, IPortNumber As Long, sErrorCode As Integer, IpszAdditionalInfo As String)

Visual C++ 6.0
void OnProblem(LPCTSTR IpszIPAddress, long IPortNumber, short sErrorCode, LPCTSTR IpszAdditionalInfo)

Argument	Description
IpszIPAddress	IP address
IPortNumber	Port number
sErrorCode	Error code (see below)
IpszAdditionalInfo	Additional information (not in use)

Example

Visual Basic 6.0
Text1.Text = "Error - " + IpszIPAddress + " [" + Format\$(IPortNumber) + "], Code : " + Format\$(sErrorCode)

Visual C++ 6.0
TRACE("Error - %s [%d], Code : ",IpszIPAddress,IPortNumber,sErrorCode);

Remarks

Error from SavoyHsms

Error code	Description
-1	Failed to send message
-2	Received message bigger than buffer size
-3	(not used)
-4	T8 timeout
-5	T3 timeout
-6	T5 timeout
-7	T6 timeout
-8	T7 timeout

Error from WinSock

Error code	Description
10093	Socket has not been initialized.
10050	Network subsystem error.
10048	Socket local address is in use.
10014	Invalid user address (like invalid character).
10036	Service provider is in progress.
10049	Remote address can not be available.
10047	Cannot use specified address family for this socket.
10061	Connection has been refused.
10039	?
10022	Invalid listening socket.

10056	Already connected.
10024	?
10051	Cannot reach to network.
10055	Buffer size is not enough.
10038	Not a socket.
10060	Time out before established connection.
10035	Cannot execute right now.

See Also

3.3.4 Received

Notifies that SavoyHsms control received message.

Syntax

Visual Basic 6.0

```
Event Received(IpszIPAddress As String, IPortNumber As Long, IpszMsg As String)
```

Visual C++ 6.0

```
void OnReceived(LPCTSTR IpszIPAddress, long IPortNumber, LPCTSTR IpszMsg)
```

Argument	Description
IpszIPAddress	IP address
IPortNumber	Port number
IpszMsg	Received message

Example

Visual Basic 6.0

```
With SavoySecsII1
  .Msg = IpszMsg
  If .Stream = 1 AND .Function = 13 Then
    ' S1F13
    ...
  End If
End With
```

Visual C++ 6.0

```
m_msg.SetMsg(IpszMsg);
if(m_msg.GetStream()==1 && m_msg.GetFunction()==13)
{
  // S1F13
  ...
}
```

Remarks

See Also

3.3.5 SelChanged

Notifies that another HSMS connection has been selected on the screen.

Syntax

Visual Basic 6.0

```
Event SelChanged(IpszIPAddress As String, IPortNumber As Long)
```

Visual C++ 6.0

```
void OnSelChanged(LPCTSTR IpszIPAddress, long IPortNumber)
```

Argument	Description
IpszIPAddress	IP address
IPortNumber	Port number

Example

Visual Basic 6.0

```
' Switch to different client
.IPAddress = IpszIPAddress
.PortNumber = IPortNumber
```

Visual C++ 6.0

```
// Switch to different client
m_ctrl.SetIPAddress(IpszIPAddress);
m_ctrl.SetPortNumber(IPortNumber);
```

Remarks

See Also