
CommSMSLib

Programmer's Reference



Version 1.0

for AD340-SCF AlarmCaller

Written, edited, and produced by
TOP TEAM TECHNOLOGY INC.
2025 年 08 月 12 日

Information in this document is subject to change without notice and does not represent a commitment on the part of TOP TEAM Technology Inc. The software described in this document is furnished under a license agreement or nondisclosure agreement. The software may be used or copied only in accordance with the terms of the agreement. It is against the law to copy the software on any medium except as specifically allowed in the license or nondisclosure agreement. No part of this manual may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording, for any purpose without the express written permission of the publisher.

PUBLISHED BY

TOP TEAM TECHNOLOGY INC.

8-7F, No.2, ChienPa Rd. Chung-Ho City Taipei, Taiwan, R.O.C.

Copyright © 2004 by TOP TEAM TECHNOLOGY INC. All rights reserved.

Printed and bound in the Taiwan, R.O.C.

Microsoft, MS-DOS, Windows, and Windows operating system logo are registered trademarks of Microsoft Corporation. All other trademarks and service marks are the property of their respective owners.

Index

Index	i
Chapter 1 Introduction	1
Overview	1
How to use	1
Chapter 2 Properties, Methods	2
CommTimeout Property	2
CommRetry Property	3
IsConnected Property	4
IsCommPortOpen Property	5
Status Property	6
ErrorCode Property	7
ErrorMessage Property	8
Initialize() Method	9
SendSMS() Method	10
Terminate() Method	11

Overview

CommSMSLib ActiveX is an ActiveX component design to help programmer communicate to AD340-SCF AlarmCaller. It provides the object and functions which programmers can easily incorporate into their application. Using the object and its functions, users' application can send a GSM short message by simply providing the message content string and destination phone number to the GSM AlarmCallers.

CommSMSLib ActiveX is an ActiveX code component. Unlike an ActiveX Control which can be inserted on a form to be manipulated by users, CommSMSLib must be by programmers. CommSMSLib is a DLL file, so it cannot be executed alone.

How to use

CommSMSLib ActiveX includes 3 files :

1. **ACSMS.dll** It is for CommSMSLib.dll Library file.
2. **CommSMSLib.dll** It is an ActiveX code component.
3. **GSMCodeMap.cod** It is a GSM Code Map file.

These 3 files must be put to the same directory.

CommRetry Property

Gets or sets the number of times a transaction will retry before it is considered to have failed.

Syntax

object.CommRetry [= *Integer*]

The CommRetry property syntax has these parts:

Part	Description
<i>object</i>	An object expression of commSMS object
<i>Integer</i>	An Integer expression. The number of times a transaction will retry before it is considered to have failed.

Settings/Property Value

Setting/Property Value	Description
3	(Default)

Example

The following example illustrates the usage of the **CommRetry** Property.

```

Dim SMS as commSMS
Dim aa as Integer
Dim bb as Integer

Set SMS = New commSMS
aa= SMS.CommRetry           ' Returns 3 (Default)
SMS.CommRetry=4              ' Set Retry Times to be 4
bb=SMS.CommRetry            ' Returns 4

```

CommTimeout Property

Gets or sets the value of communication timeout in sec

Syntax

object.CommTimeout [= *Integer*]

The CommTimeout property syntax has these parts:

Part	Description
<i>object</i>	An object expression of commSMS object type
<i>Integer</i>	An Integer expression.

Settings/Property Value

Setting/Property Value	Description
1	(Default)

Example

The following example illustrates the usage of the **CommTimeout** Property.

```

Dim SMS as commSMS
Dim aa as Integer
Dim bb as Integer

Set SMS = New commSMS
aa= SMS.CommTimeout           ' Returns 1 (Default)
SMS.CommTimeout=2              ' Set CommTimeout to be 2
bb=SMS.CommTimeout            ' Returns 2

```

ErrorCode Property

Gets the error code of the previous error condition

Syntax

object.ErrorCode [= *Integer*]

The ErrorCode property syntax has these parts:

Part	Description
<i>object</i>	An object expression of commSMS object type.
<i>Integer</i>	An Integer expression.

Return Value

Property Value	Description
0	Success (Default)
1	Fail, communication error.
2	Fail, queue is full.
3	Fail, no short message.
4	Fail, no telephone number.
5	Fail, memory is full.
6	Fail, the short message is blank.
7	Fail, the telephone number is blank.
Other value	Fail, unknown error.

Example

ErrorMessage Property

Gets the error message of the previous error condition

Syntax

object.ErrorMessage [= *String*]

The ErrorCode property syntax has these parts:

Part	Description
<i>object</i>	An object expression of commSMS object type
<i>String</i>	A String expression.

Return Value

Property Value	Description
0	The action is success. (Default)
1	Communication error.
2	Queue is full.
3	No short message.
4	No telephone number.
5	Memory is full.
6	The short message is blank.
7	The telephone number is blank.
Other value	Unknown error.

Example

IsConnected Property

Returns whether an SMS Caller device is connected.

Syntax

object.IsConnected

The IsConnected property syntax has these parts:

Part	Description
<i>object</i>	An object expression of commSMS object type

Return Value

True if the Caller is connected, **False** otherwise.

Example

```
Sub IsCallerConnected ()  
    Dim SMS as commSMS  
    Dim Result as Boolean  
  
    Set SMS = New commSMS  
    Result = SMS.IsConnected()           ' If it is True, the Caller is connected now.  
End Sub
```

IsCommPortOpen Property

Returns whether the communication port of PC used to communicate with the Caller's is opened.

Syntax

object.IsCommPortOpen

The IsCommPortOpen property syntax has these parts:

Part	Description
<i>object</i>	An object expression of commSMS object

Return Value

True if the communication port is opened, **False** otherwise.

Example

```
Sub CallerIsCommPortOpen ()
    Dim SMS as commSMS
    Dim Result as Boolean

    Set SMS = New commSMS
    Result = SMS.IsCommPortOpen()
End Sub

If Result is True, the Caller's communication port is opened now, otherwise False.
```

Status Property

Returns whether the Caller's is ready.

Syntax

object.Status [= *Integer*]

The Status property syntax has these parts:

Part	Description
<i>object</i>	An object expression of commSMS object
<i>Integer</i>	An Integer expression.

Return Value

Property Value	Description
0	Ready
-1	Not ready.

Example

Initialize Method

Initialize the SMS Caller so it is ready to accept message for sending.

```
Public Method Initialize(ByVal CommPort As Integer, _
    ByVal CommSetting As String) As Boolean
```

Parameter

CommPort

Required. **Integer** Expression. The Caller's Communication port.

CommSetting

Required. **String** Expression. The Caller's Communication Settings.

Remarks

Before it can be used to send short message, the Caller must be initialized. Returns True, if the Caller is successfully initialized; otherwise, it returns **False**.

Example

This example uses the **Initialize** method to initialize the Caller.

```
Dim SMS as commSMS
Dim myport as Integer
Dim mysetting as String
Dim Result as Boolean

Set SMS = New commSMS
myport = 1
mysetting = "9600,n,8,1"
Result = SMS.Initialize(myport,mysetting)
```

If Result is **True**, the Caller is initialized successfully, otherwise Result is **False**.

SendSMS Method

Order the caller to send a short message to the destination mobile phone.

```
Public Method SendSMS(ByVal PhoneNo As String, _
                      ByVal Message As String, _
                      Optional ByValIsUnicode As Boolean = True) As Boolean
```

Parameter

PhoneNo

Required. A **String** expression that designates the telephone number that will receive the short message

Message

Required. A **String** expression that specifies the short message content to be sent.

The limit on the length of the message is 70 characters for message in Unicode and 140 for standard GSM alphabet.

IsUnicode

Optional. A **Boolean** Expression used to specify whether the short message string is in Unicode or not.

Remarks

SendSMS returns **True** if the sending of the short message with the Caller succeeds. Otherwise, it returns **False**.

Example

This example uses the **SendSMS** method to command the Caller to send a short message.

```
Dim SMS as commSMS
Dim strPhoneNo as String
Dim strSMS as String
Dim Result as Boolean

Set SMS = New commSMS
strPhoneNo="0932123123"
strSMS="This is a test."
Result =SMS.SendSMS(strPhoneNo,strSMS, False)
```

If Result is **True**, send the short message with the Caller succeeds; otherwise it is **False**.

Terminate Method

Terminate the Active to end the using of the Caller.

Syntax

```
object.Terminate()
```

Return Value

No return value.

Remarks

Before ending the application calling the CommSMSLib ActiveX, the programmer must use **Terminate** method to stop the communication with the Caller.

Example

This example uses the **Terminate** method to stop communicating to the Caller.

```
Dim SMS as commSMS
Dim myport as Integer
Dim mysetting as String
Dim myslaveid as Integer
Dim Result as Integer

Set SMS = New commSMS
SMS.Terminate()           ' End the Caller.
```