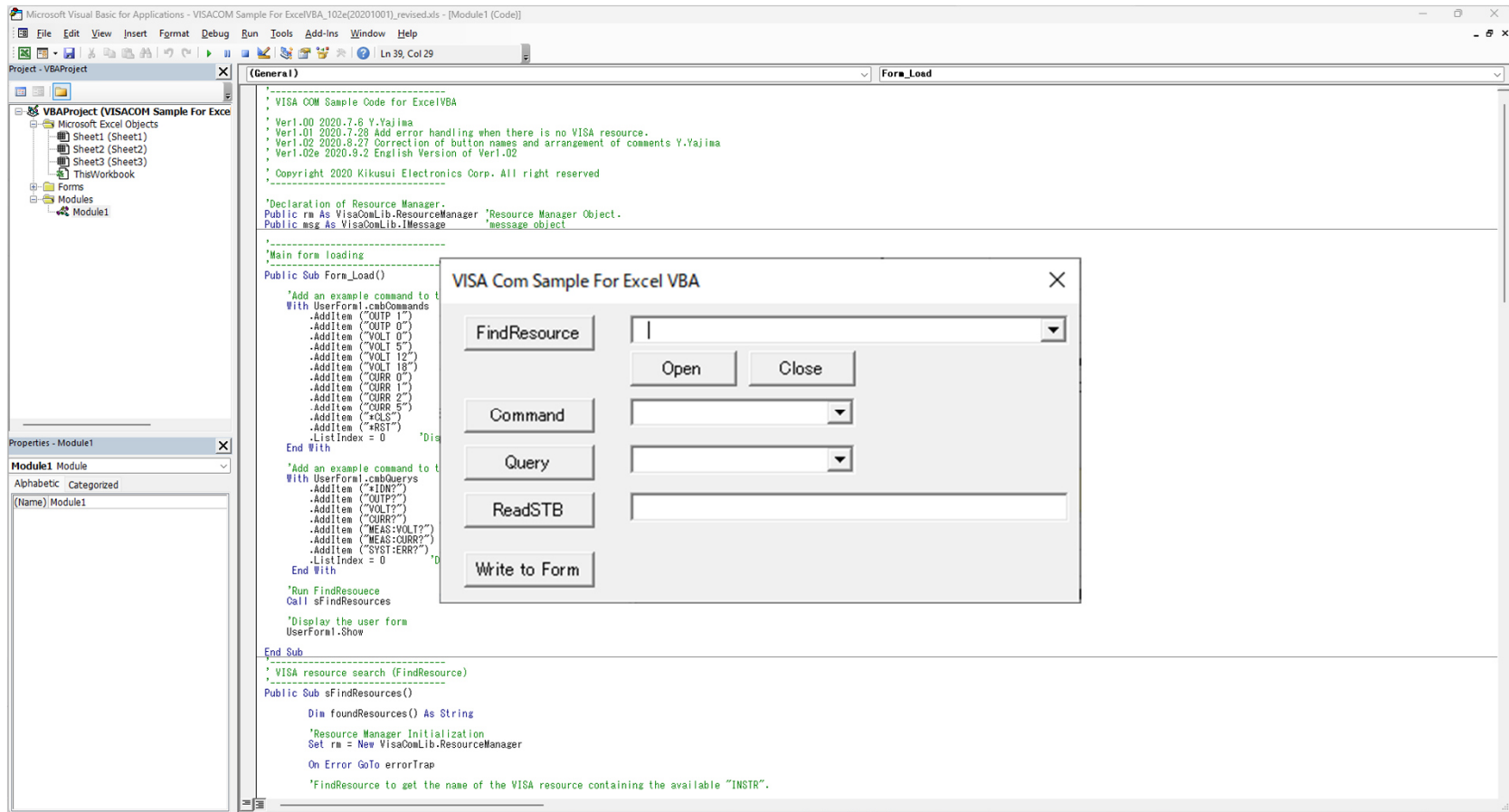


Introduction to sample code for device control using KI-VISA Library (for Excel VBA)



Contents

- Introduction to our sample code
- What can our sample code do?
- What is needed to run our sample code?
- Details of our sample code
- How to start and edit our sample code
- How to start Visual Basic Editor
- Form and Event Handler
- How to manage the reference
- Guidebook / Contact Us

Introduction to our sample code

- **Purpose of our sample code**

Our sample code simplifies the control of measurement instrumentation by utilizing the KI-VISA library for Microsoft Excel VBA.

You can easily control devices, such as power supplies, measuring instruments and electronic loads, by modifying the initial value of our sample code, according to the communication interface manuals.

As a typical example, we provide our sample code specifically for our DC power supply PMX series.

- **Intended user**

Our sample code is designed for development engineers who need guidance in creating programming codes to control the devices using the KI-VISA library.

What can our sample code do?

- **Find a VISA resource**

Click **FindResource** to search for an available VISA resource (device).

- **Connect a PC and device**

Select the VISA resource name from the FindResource pull-down menu. Click **Open** to open the VISA session and establish a communication link with the device.

- **Send a command**

[Command] button

Select a command from the Command pull-down menu. Click **Command** to send the command to the device.

- **Send a query command**

[Query] button

Select the command from the Query pull-down menu. Click **Query** to send the query command to the device. The result data is returned and displayed in the text box.

- **Read a status byte register**

[ReadSTB] button

Click **ReadSTB** to execute the ReadSTB method. The value of the status byte register is displayed in the text box.

*Available only when connected via the GPIB or USB interface.

- **Copy the result data to the form**

[Write to Form] button

Copy the result data into cell E8 of Excel Sheet 1.

VISA Com Sample For Excel VBA

FindResource

Open Close

Command

Query

ReadSTB

Write to Form

What is needed to run our sample code?

- **Windows PC with the following requirements**

- KI-VISA library (Ver.5.5.0 or later)

<https://global.kikusui.co.jp/dri-fir-upd/ki-visa/>

**Please note that if another VISA library, such as NI-VISA or Keysight VISA, is already installed, this library is not necessary. The reference setting is required, please see [How to manage the reference](#).*

- Microsoft Excel

**Our sample code was tested on Microsoft Excel 2016.*

- **Connection cable**

You will need one of the following connection cables.

- USB connection: USB Cable (USB Type A-Type B)
- Serial connection: Serial cable (9-pin cross type)
USB-RS232C conversion cable
(e.g., REX-USB60F manufactured by RATO Systems, Inc.)
**If your PC has a serial port, the conversion cable is not required.*
- GPIB connection: USB-GPIB conversion cable
(e.g., USB-GPIB-HS manufactured by National Instruments)
**Use the cable that supports the VISA COM library.
You cannot use the GPIB-USB cable by RATO Systems, Inc.
- LAN connection: LAN cable (cross type)
**In case of AUTO-MDIX LAN interface, a straight cable is also available.*

- **Device such as measuring instrument or power supply**

- You will need the device with the following interfaces:
 - GPIB interface
 - Serial interface (RS232C)
 - USB interface (USBTMC)
 - LAN interface (VXI-11, HiSLIP)

Details of our sample code

- **Contents of the ZIP file**
 - The book file (.xls) with the VBA macro
- **Programming Language**
 - Excel VBA
- **Code Examples**
 - Finding the VISA resource name of the device (FindResource method)
 - How to open/close the VISA session (Open/Close method)
 - Setting RS-232C baud rate
 - Sending a command string (WriteString method)
 - Receiving a query message (ReadString method)
 - Reading the status byte by serial polling (ReadSTB method)
- **Precautions before implementing our sample code**
 - You can modify our sample code, but please note that they are provided only as a reference of your programming.
 - Our sample code does not guarantee complete operation in all environments.
 - Error handling has been minimized to simplify our sample code.
We strongly recommend handling all errors (On Error GoTo) on your actual system.

How to start and edit our sample code

Click on Enable Content.

Click on Open Form.

To edit the code, check Developer.

If the code doesn't run, please check:
- VISA installation
- References in VBA.

Excel VBA Code:

```
Sub OpenForm()  
    Application.Run "KIKUSUI.PMX!OpenForm"  
End Sub
```

Response example:
(when clicking Write To Form)

***How to show the Developer tab:**
The Developer tab isn't displayed by default, but you can add it to the ribbon:
1. On the File tab, go to Options > Customize Ribbon.
2. Under Customize the Ribbon and under Main Tabs, select the Developer check box.

***How to enable macros:**
If you see a security warning when you open this document, you can choose to make it a trusted document and enable macros by selecting Enable Content.

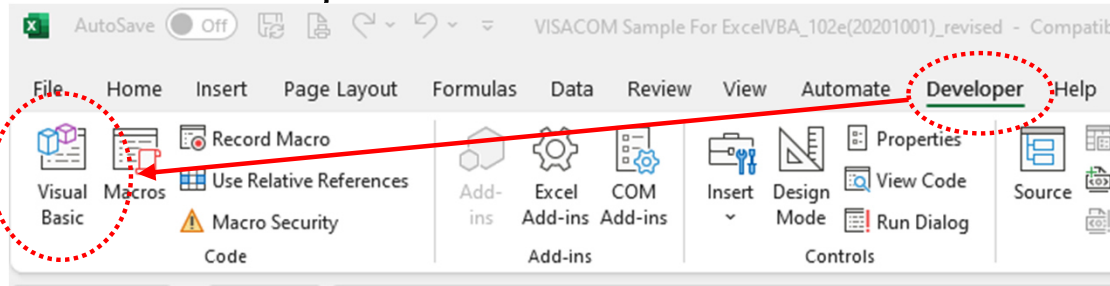
***How to manage the reference:**
If you are experiencing issues with the visa library reference due to conflicts with the visa version or type, follow these steps:
1. In the Visual Basic Editor, go to Tools > References... > Browse...
2. Select "C:\Program Files (x86)\IVI Foundation\VISA\VisaCom\GlobMgr.dll".
The VISA COM 5.xx Type Library will then be available.
(Note: The numbers in "xx" vary depending on your VISA version.)

References - VBAProject

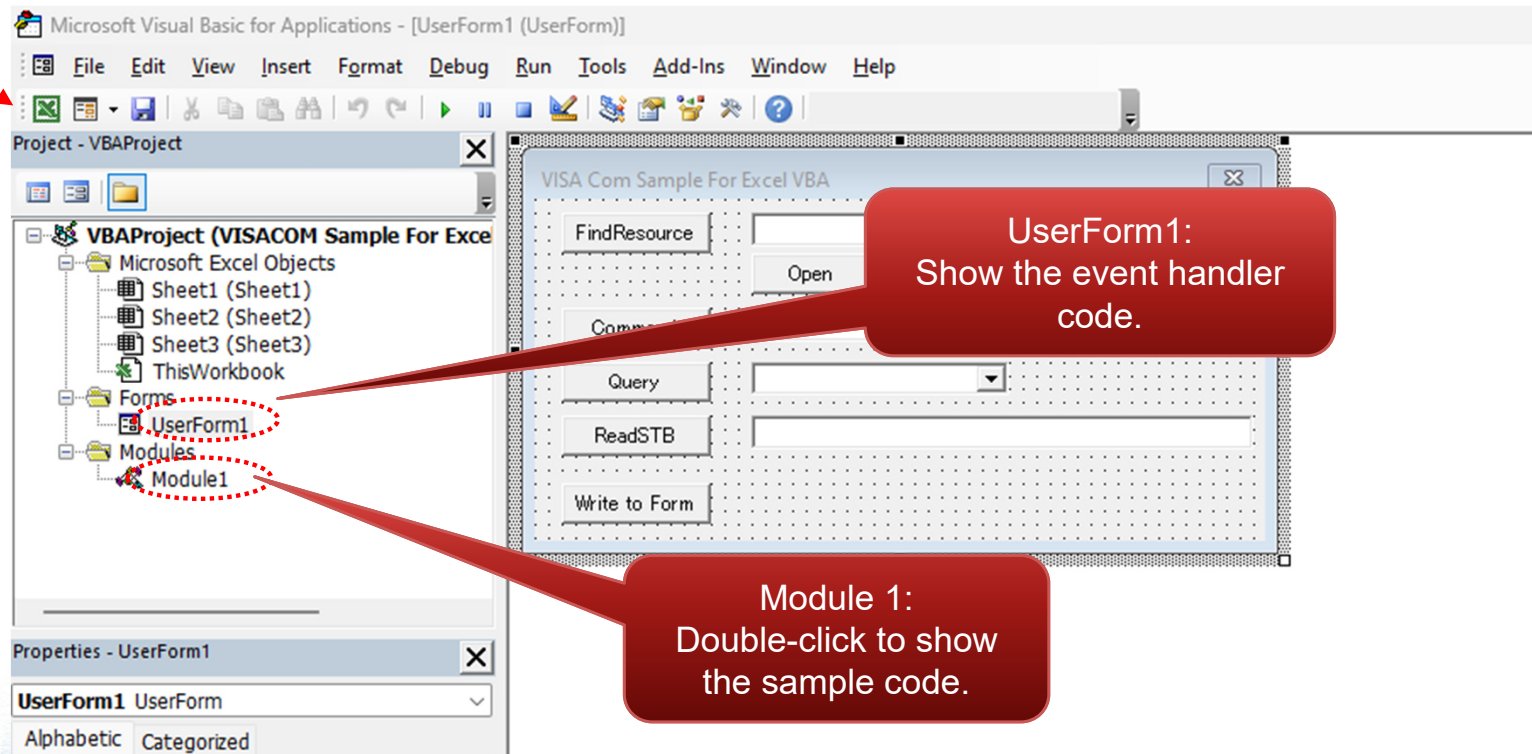
Available References:
<input checked="" type="checkbox"/> Visual Basic For Applications
<input checked="" type="checkbox"/> Microsoft Excel 16.0 Object Library
<input checked="" type="checkbox"/> OLE Automation
<input checked="" type="checkbox"/> Microsoft Office 16.0 Object Library
<input checked="" type="checkbox"/> VISA COM 5.5 Type Library
<input type="checkbox"/> Microsoft Forms 2.0 Object Library
<input type="checkbox"/> AccessibilityCplAdmin 1.0 Type Library
<input type="checkbox"/> Acrobat
<input type="checkbox"/> Acrobat Access 3.0 Type Library
<input type="checkbox"/> AcrobatLib
<input type="checkbox"/> Active DS Type Library
<input type="checkbox"/> ActiveMovie control type library

How to start Visual Basic Editor

Click on the Developer tab > Visual Basic.



The Visual Basic Editor will start up.



Form and Event Handler

UserForm1

```

'-----
' Click WriteToForm
'-----
Private Sub btnWriteToForm_Click()

    Sheet1.Range("E8").Value = txtResult.Text

End Sub

```

```

'-----
' Click FindResource
'-----
Private Sub btnFindResource_Click()

    Call sFindResources

End Sub

'-----
' Click Open
'-----
Private Sub btnOpen_Click()

    Call visaOpen(cmbFoundResources.Value)

End Sub

'-----
' Click Close
'-----
Private Sub btnClose_Click()

    Call visaClose

End Sub

'-----
' Click Command
'-----
Private Sub btnCommand_Click()

    Call sendCommand(cmbCommands.Value)

End Sub

'-----
' Click Query
'-----
Private Sub btnQuery_Click()

    txtResult.Text = fsendQuery(cmbQuerys.Value)

End Sub

'-----
' Click ReadSTB
'-----
Private Sub btnReadSTB_Click()

    txtResult.Text = freadStb()

End Sub

```

How to manage the reference

If you want to create a new project or are experiencing issues with building a code, please refer to the following steps:

- Go to **Tools > References...** and check **VISA COM 5.x Type Library** and click **OK**.
- If the VISA COM 5.x Type Library is not found, go to **Browse...** to add a reference based on your VISA version:

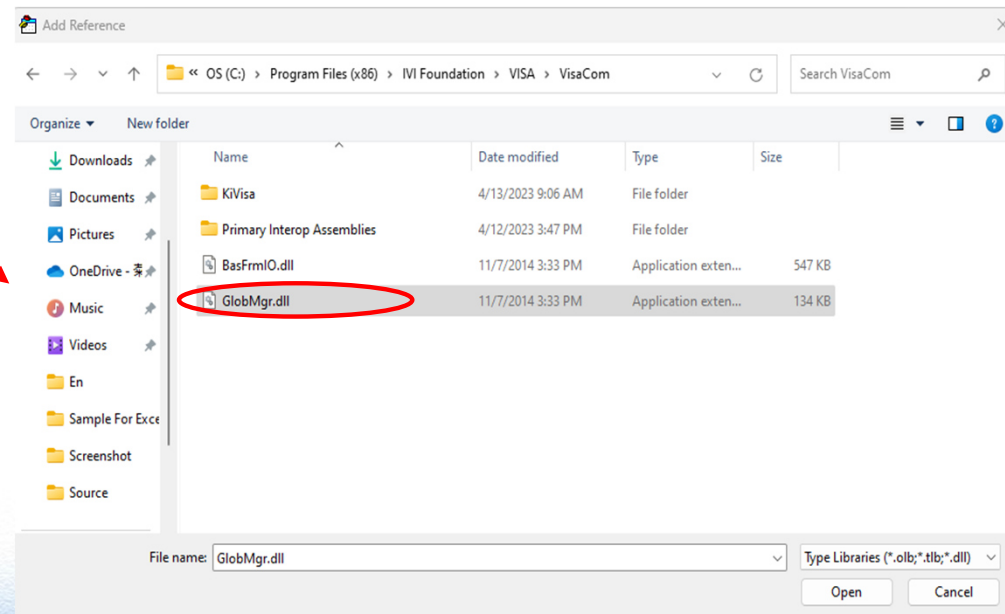
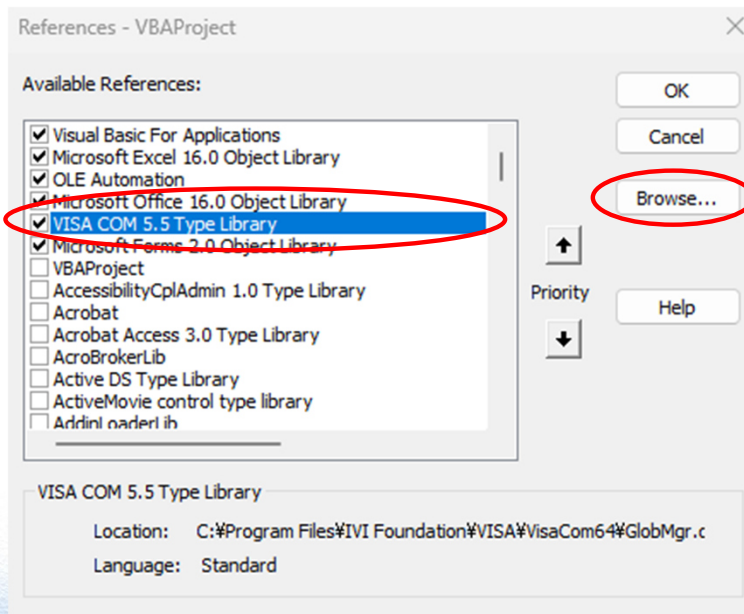
*The "x" in "5.x" depends on your VISA version.

For 32-bit OS:

C:\Program Files\IVI Foundation\VISA\VisaCom\GlobMgr.dll

For 64-bit OS:

C:\Program Files (x86)\IVI Foundation\VISA\VisaCom\GlobMgr.dll



Guidebook / Contact Us

- Please download the KI-VISA programming guidebook for reference (Note: The guidebook is available in Japanese text only) :
<https://kikusui.co.jp/kikusupport/downloads/guidebook/programing-guidebook/>
- For any inquiries regarding our sample code, please contact us using the following inquiry form:
<https://global.kikusui.co.jp/contact/lead/>

Please include the phrase “To the person in charge of the sample code” in the text and provide your question.

```

1 Attribute VB_Name = "Module1"
2
3 ' VISA COM Sample Code for ExcelVBA
4
5 ' Ver1.00 2020.7.6 Y.Yajima
6 ' Ver1.01 2020.7.28 Add error handling when there is no VISA resource.
7 ' Ver1.02 2020.8.27 Correction of button names and arrangement of comments Y.Yajima
8 ' Ver1.02e 2020.9.2 english Version of Ver1.02
9
10 ' Copyright 2020 Kikusui Electronics Corp. All right reserved
11
12
13 'Declaration of Resource Manager.
14 Public rm As VisaComLib.ResourceManager 'Resource Manager Object.
15 Public msg As VisaComLib.IMessage      'message object
16
17
18 'Main form loading
19
20 Public Sub Form_Load()
21
22     'Add an example command to the combo box for Command
23     With UserForm1.cmbCommands
24         .AddItem ("OUTP 1")
25         .AddItem ("OUTP 0")
26         .AddItem ("VOLT 0")
27         .AddItem ("VOLT 5")
28         .AddItem ("VOLT 12")
29         .AddItem ("VOLT 18")
30         .AddItem ("CURR 0")
31         .AddItem ("CURR 1")
32         .AddItem ("CURR 2")
33         .AddItem ("CURR 5")
34         .AddItem ("*CLS")
35         .AddItem ("*RST")
36         .ListIndex = 0      'Display the first row of the combo box
37     End With
38
39     'Add an example command to the combo box for Query
40     With UserForm1.cmbQuerys
41         .AddItem ("*IDN?")
42         .AddItem ("OUTP?")
43         .AddItem ("VOLT?")
44         .AddItem ("CURR?")
45         .AddItem ("MEAS:VOLT?")
46         .AddItem ("MEAS:CURR?")
47         .AddItem ("SYST:ERR?")
48         .ListIndex = 0      'Display the first row of the combo box
49     End With
50
51     'Run FindResoucece
52     Call sFindResources
53
54     'Display the user form
55     UserForm1.Show
56
57 End Sub
58
59 ' VISA resource search (FindResource)
60
61 Public Sub sFindResources()
62
63     Dim foundResources() As String
64
65     'Resource Manager Initialization
66     Set rm = New VisaComLib.ResourceManager
67
68     On Error GoTo errorTrap
69
70     'FindResource to get the name of the VISA resource containing the available "INSTR".
71     'For more information, see KI-VISA Library VISA COM Guidebook 8 - Dynamically Configuring VISA
Addresses (starting on page 48).

```


Module1.bas

```
72         foundResources = rm.FindRsrc("?*INSTR")
73
74         'Add the discovered VISA resources to the combo box
75         UserForm1.cmbFoundResources.List = foundResources
76
77         'Display the first line of the combo box
78         UserForm1.cmbFoundResources.ListIndex = 0
79
80         Exit Sub
81
82     errorTrap:
83         r = MsgBox("No VISA resource is connected", vbCritical)
84
85 End Sub
86
87 '-----
88 ' VISA Resource Open
89 '-----
90 Public Sub visaOpen(svisaResource As String)
91     'Open VISA resource
92
93
94     Dim r As Integer
95
96     Dim iHwIfType As Integer           'Hardware interface type
97     Dim gpib As VisaComLib.IGpib
98     Dim RS232 As VisaComLib.ISerial
99     Dim tcp As VisaComLib.ITcpipInstr
100    Dim usb As VisaComLib.IUsb
101
102    On Error GoTo errorTrap
103
104    'Open by specifying the VISA resource name
105    Set msg = rm.Open(svisaResource, NO_LOCK, 0, "")
106
107    'Set msg = rm.Open("ASRL3::INSTR", NO_LOCK, 0, "")
108    'It is also possible to write VISA resource names directly.
109
110    'Getting the Interface Type
111    iHwIfType = msg.HardwareInterfaceType
112
113    'Execute individual configuration for each interface.
114    Select Case iHwIfType
115        Case 1
116            ' GPIB
117            msg.TerminationCharacter = 10
118            msg.TerminationCharacterEnabled = True
119
120        Case 4
121            ' RS232
122            Set RS232 = msg                                'Configuration for via the ISerial interface in the msg
123            object.
124
125            msg.TerminationCharacter = 10
126            msg.TerminationCharacterEnabled = False
127            msg.SendEndEnabled = True
128
129            RS232.BaudRate = 19200                          'Baud rate setting 19200bps
130            RS232.DataBits = 8                             'Bit length 8 bits
131            RS232.StopBits = ASRL_STOP_ONE                 'Stop bit 1 bit
132            RS232.Parity = ASRL_PAR_NONE                   'No parity
133            RS232.FlowControl = ASRL_FLOW_XON_XOFF         'With XON/OFF flow control
134            RS232.EndIn = ASRL_END_TERMCHAR                'Waiting for a terminated character to be entered
135            RS232.EndOut = ASRL_END_TERMCHAR               'Transmit with a terminator.
136
137        Case 6
138            ' LAN
139            Set tcp = msg
140
141        Case 7
142            ' USB
```

Module1.bas

```
143         Set usb = msg
144
145     End Select
146
147     r = MsgBox(svisaResource + " is opened.", vbInformation)
148
149     Exit Sub
150
151 errorTrap:
152     r = MsgBox("An error occurred in 'visaOpen'.", vbCritical)
153
154
155 End Sub
156
157 '-----
158 ' Send command
159 '-----
160 Public Sub sendCommand(Command As String)
161
162     Dim r As Integer
163     Dim strRead As String
164
165     On Error GoTo errorTrap
166
167     r = msg.WriteString(Command + vbCrLf) ' * Command delimiter (LF) is required at the end of the message,
so vbCrLf is added.
168
169     Exit Sub
170
171 errorTrap:
172     r = MsgBox("An error occurred in 'sendCommand'", vbCritical)
173
174 End Sub
175
176 '-----
177 ' Query command transmission / result reception
178 '-----
179 Public Function fsendQuery(queryCommand As String) As String
180
181     Dim r As Integer
182     Dim strRead As String
183
184     On Error GoTo errorTrap
185
186     r = msg.WriteString(queryCommand + vbCrLf) ' * Command delimiter (LF) is required at the end of the
message, so vbCrLf is added.
187     strRead = msg.ReadString(256) ' Receive buffer size is 256 bytes
188     strRead = Left(strRead, Len(strRead) - 1) ' Remove delimiter
189
190     fsendQuery = strRead ' Returns the obtained value.
191
192     Exit Function
193
194 errorTrap:
195     r = MsgBox("An error occurred in 'fsendQuery'", vbCritical)
196
197 End Function
198
199 '-----
200 ' Receiving status bytes
201 '-----
202 Public Function freadStb() As String
203
204     Dim r As Integer
205     Dim stb As Integer
206
207     On Error GoTo errorTrap
208
209     stb = msg.readStb
210
211     freadStb = Str(stb)
212     Exit Function
```

Module1.bas

```
213|
214| errorTrap:
215|
216|     r = MsgBox("An error occurred in 'freadStb'", vbCritical)
217|
218| End Function
219|
220| '-----
221| ' Closing VISA Resources
222| '-----
223| Public Sub visaClose()
224|
225|     Dim r As Integer
226|
227|     On Error GoTo errorTrap
228|     msg.Close
229|
230|     r = MsgBox("VISA resource closed", vbInformation)
231|     Exit Sub
232|
233| errorTrap:
234|
235|     r = MsgBox("An error occurred in 'visaClose'", vbCritical)
236|
237| End Sub
```