

How to Program the 3499A/B/C Switch and the 34401A Digital Multimeter for Scanning Measurements

Application Note 1459

Introduction

This application note describes programming in Microsoft® Visual Basic for the 3499A/B/C switch/control unit, the N2260A 40-channel plug-in multiplexer module and the 34401A digital multimeter (DMM) so you can make multiple automated measurements from your PC and log the data in an array in Visual Basic. The 3499A provides up to 80 channels and the 3499B offers up to 200 channels. When you need to make a large number of measurements, you can use the 9slot 3499C switch for up to 360 channels. With all three models, you can use a N2260A multiplexer plug-in module and a 34401A DMM as the measuring instrument.

This application note tells how to:

- Close or open a relay
- Monitor a DMM reading
- Initiate a relay closure in the 3499 that triggers the DMM through a single reading
- Initiate a scan of multiple channels and read the data from the DMM with a bi-directional trigger

- Make trigger connections when a relay is closed

You also can download from the Agilent Web site:

- Example code in Visual Basic 6.0 that demonstrates fundamental measurement techniques
- Example code for Visual Basic for Applications in Excel that demonstrates inserting data into an Excel worksheet

To download these code examples, go to www.agilent.com/find/3499example.

The Visual Basic and Excel programming examples discussed in this application note require knowledge of the Visual Basic 6.0 development environment, and the use of referenced objects and instrument I/O. Some knowledge of the 3499A/B/C switch, the 34401A DMM and the intended application is assumed. If you need instrument information, refer to instrument data sheets and manuals.

Programming the 3499A/B/C switch and the 34401A DMM

To program these instruments, you need to have a GPIB or RS-232 interface on your PC. The examples included use the Agilent ISDK (Instrument Software Development Kit) in Visual Basic 6.0 with GPIB that comes with Agilent IntuiLink for the 34401A. You may substitute other libraries, such as VISA, by using the instrument commands shown. The Agilent IntuiLink I/O object library will work for GPIB cards or USB-to-GPIB converters from either Agilent or National Instruments. You can load the Agilent I/O objects to your PC by loading IntuiLink for DMM 34401A, which is available for download at www.agilent.com/find/intuilink. To use GPIB, you must have a GPIB interface installed and working prior to using this software.



Establishing a connection

Making a connection requires a reference to Agilent I/O Manager. From the **Project>References...** menu in Visual Basic, select **Agilent I/O Manager** to create the reference for your project. Use the code below to establish a connection.

Once you have executed the first set of code, you can use the object DMM and Switch to talk to the instrument. The lower set of code demonstrates how to retrieve the instrument error from the DMM.

```
Dim DMM As AgtIOServer
Dim Switch As AgtIOServer
Dim io_mgr As AgilentIOUtilsLib.AgtIOManager

Set io_mgr = New AgtIOManager
Set DMM = io_mgr.ConnectToInstrument("GPIB::22")
Set Switch = io_mgr.ConnectToInstrument("GPIB::9")
```

```
Dim Reply As String
DMM.Output "Syst:Error?"
DMM.Enter Reply
```

Setting up a function on the 34401A DMM

To set up a function that you want to monitor later, use the Configure command. The code below sets up the 34401A, range 10,000 ohms, with an integration time of 10 line cycles (6-digit resolution). Once you send the Configure command, you can use the Sense command to refine the setting. The Sense command is optional. Without the Sense command, the settings use default values of the Configure command.

```
DMM.Output "Configure:Resistance 1E4"
DMM.Output "Sense:Res:NPLC 10"
```

Closing a relay on the 3499 switch

To make a measurement, you must close the correct relay. You can close one or several relays of the MUX card with one command. Notice there is a space between the word "Close" and the first parenthesis, and the word "Route" is optional.

```
Switch.Output "Route:Close (@301:305,309)"
```

The syntax 301:305 acts on channels 301-305. Open the same relays with the Route:Open (@301:305,309) command. To open all the relays send the code below.

```
Switch.Output "Route:Open ALL"
```

To be sure all relay operations are complete, use this code and throw away the returned value:

```
Switch.Output "**OPC?"  
Switch.Enter Reply
```

Reading data with the 34401A and 3499 switch

There are three fundamental methods for reading the data:

1. *Monitor a reading, no hardware trigger:* To monitor a single channel, close the relay on the 3499 switch to route the signal. Then take a reading with the voltmeter.
2. *Single reading, hardware trigger to DMM:* When the relays are closed, the 3499 triggers the DMM to take a reading. In this mode the 34401A provides a default delay dependent on the function and range.
3. *Multiple readings, bi-directional trigger:* When a relay closes, the 3499 triggers the DMM to take a reading. The DMM triggers the 3499 when the reading is complete, which allows the 3499 to progress to the next relay and so on. This allows for successive multiple readings with the same function.

Method 1: Monitor a reading, no hardware trigger

To monitor a single channel, close the relay on the 3499 switch to route the signal. Then take a reading with the DMM. The two are independent operations. Delay for settling is provided by the code between relay closures and when the reading on the DMM is initiated.

```
Dim Reply As String

' Close the required relay and wait until it is done
Switch.Output "Route:Close (@105)"
Switch.Output "*OPC?"
Switch.Enter Reply

' Set to resistance and take a reading
DMM.Output "Configure:Resistance 1E4"
DMM.Output "Sense:Res:NPLC 10"
DMM.Output "Read?"
' return reading as a string
DMM.Enter Reply
```

Method 2: Single reading, hardware trigger to DMM

When you close a relay and then read the channel with a DMM, you need a delay for settling. By connecting the trigger out of the 3499 switch to the trigger in of the DMM, the 34401A provides a default delay dependent on the function and range.

```
Dim Reply As String
Dim reading As Double
' Set DMM for DC, 10V range 51/2 digit resolution
With DMM
    .Output "Conf:volt:DC 10;:Volt:DC:NPLC 1"
    .Output "Trig:Sour ext"
    .Output "*Opc?"
    .Enter Reply
    .Output "Init"
End With

' Program and initiate the 3499 for one channel scan
' with DMM triggered by the 3499

With Switch
    .Output "Conf:Ext:Trigger:Output 1"
    .Output "Scan (@100)"
    .Output "Init"
End With

With DMM
    .Output "Fetch?"
    .Enter reading
End With
```

Method 3: Multiple readings, bi-directional trigger

When there are multiple channels with the same DMM configuration, the 3499 switch and 34401A DMM can be set to pace each other by connecting the trigger in and out of both instruments. Once these connections are made, the 3499 and 34401A will take a series of paced readings without any programming interaction.

```
Dim dummy As String
Dim readings() As Double
Dim readingCount As Long

' Program the 3499 for one scan with DMM synchronized to 3499
  With Switch
    .Output "Conf:Ext:Sour 0"
    .Output "Conf:Ext:Outp 1"
    .Output "Arm:sour Imm;:Arm:Count 1"
    .Output "Trig:Sour Mix"
    .Output "Scan (@100:102,105)"
    ' let the 3499 tell us how many channels
    .Output "Scan:Size?"
    .Enter readingCount
  End With

  With DMM
    .Output "Conf:volt:DC 10;:Volt:DC:NPLC 1"
    .Output "Trig:Sour ext;:Trig:count " & readingCount
    .Output "*Opc?"
    .Enter dummy
    .Output "Init"
  End With
  ' Start the scan
  switch.Output "Init"
  switch.Output "*TRG"
```

When the above code is executed, the following sequence takes place:

1. The first channel in the scan list closes
2. The 3499 triggers the DMM to take a reading
3. The DMM starts a default delay depending on range and function
4. The DMM takes a reading after the delay and stores the reading in memory

5. The DMM triggers the 3499 to close the next relay on the scan list
6. The 3499 closes the next relay
7. Repeat from step 2 until all relays in the scan list are completed

Once this sequence is complete, the readings for all the channels in the scan list are available in memory in the 34401A. Then you can read the data array with the code below. For long channel lists

and DMM measurements that take a long time, a delay or a longer timeout may be required before executing this code. For best performance, avoid multiple configurations on the DMM by combining channels with the same functions and configuration. When the sequence is completed, use this code to read the data array:

```
With DMM
  .Output "Fetch?"
  .Enter readings ' read into array from memory
End With
```

Making trigger connections for the 3499 switch and 34401A DMM

To perform a scan with the 3499 switch and the 34401A DMM,

connect the two instruments as shown in Figure 1. You can create the cable shown by purchasing the Agilent N2289A cable, cutting the DB9 connector on one end and connecting BNC connectors to the

blue and violet wires. Connect all the shields together. The connector that fits into the 3499A/B/C switch is a miniature DIN in-line plug, 8 positions (Digi-Key PN CP-2080-ND).

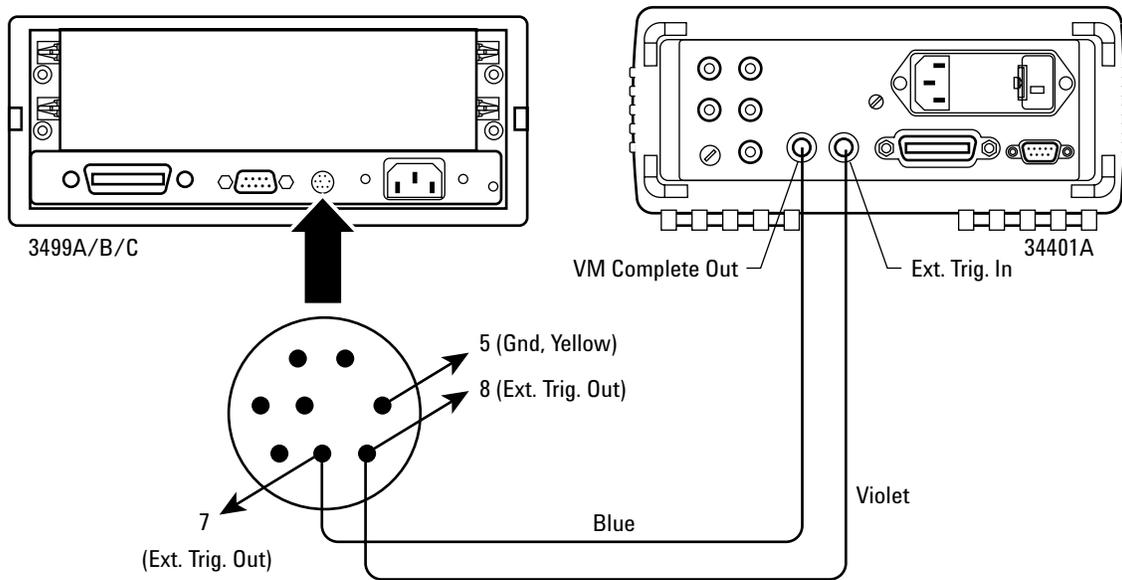


Figure 1: How to make the trigger connections

Connecting the MUX to the DMM

Connect the DMM input to the common bus 0 of the N2260A module. If you want to make 4-wire ohms measurements or RTD temperature measurements, also connect the DMM sense terminals to the common bus 1 of the N2260A module.

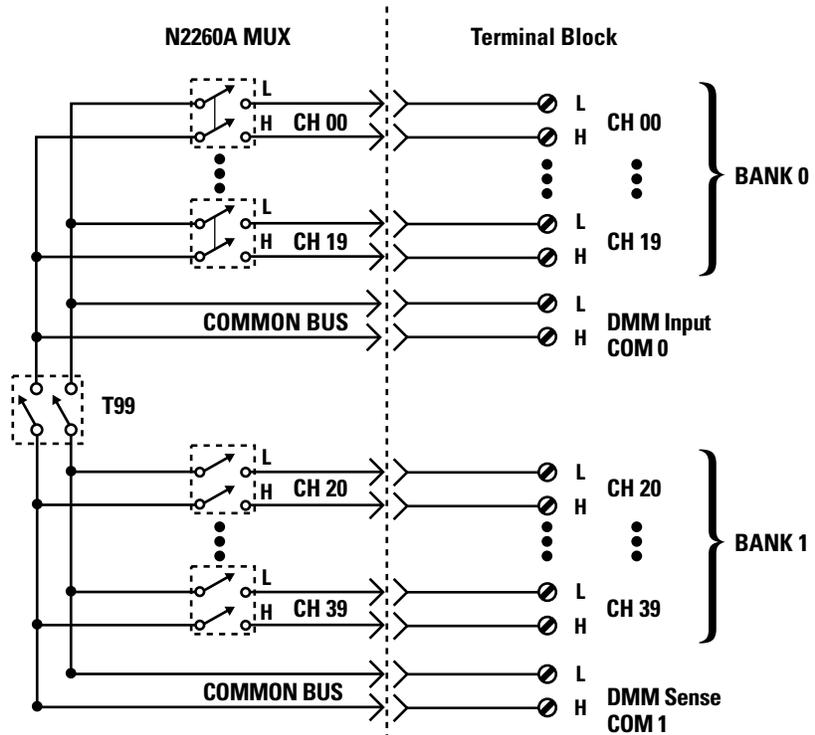


Figure 2: How to connect the MUX to the DMM

Tips for making better measurements

Reducing noise in your system

Tree switching is a method of organizing the channels of a scanner into groups, each with its own main switch. Use tree

switching to reduce noise in your system.

Without tree switching, every channel can contribute noise directly through its stray capacitance. With tree switching, groups of parallel channel

capacitances are in series with a single tree switch capacitance. The result is greatly reduced crosstalk in a large data acquisition system, due to the reduced inter-channel capacitance.

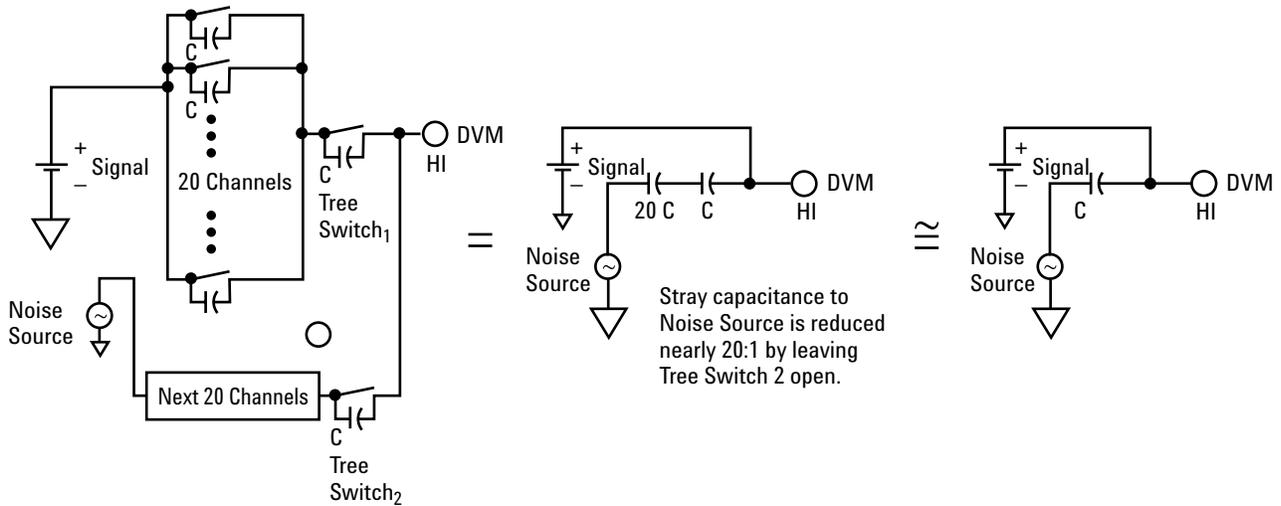


Figure 3: Tree switching

Making accurate temperature measurements

When you are measuring temperature, the 34401A measures the resistance for RTD and thermistor sensors and voltage for thermocouples. The instrument software uses these values to calculate the temperature.

For thermocouple measurements, the thermocouples must be terminated at an isothermal terminal block. The temperature of the isothermal terminal block is measured using a thermistor or other temperature device. This reference temperature value is used in the calculation of the thermocouple temperatures. One E1586A isothermal terminal

block, available from VXI Technology, is suitable for up to 39 channels of thermocouples plus one reference channel when you use it with the 3499 switch and one N2260A MUX relay card.

Making alternative measurements with large scan lists

When you are making measurements with a large scan list and functions that require long settling or take an appreciable amount of time, using a delay or increasing the timeout of the I/O will capture the bus and the PC screen. Under these circumstances, a better way to take the reading is to set the DMM to give an SRQ when readings are complete. This frees up the PC

and GPIB bus resources for other tasks while the 3499 switch and 34401A DMM are busy with scanning and measurements. If an Agilent GPIB card is installed, you can do this with an SRQ event in Visual Basic. When an SRQ event is enabled, the event will fire when the scan is complete, and the readings can be read from the DMM. Alternately, the software can poll the DMM on GPIB every fraction of a second and ask if the SRQ bit is set. Once the SRQ bit is set, the data can be read from the DMM. The examples in the 34401A SRQ directory (see below) demonstrate both techniques using the Agilent ISDK objects.

How to implement a delay

When you use the DMM and a MUX switch card, it is a good idea to add delays for settling.

```
Public Declare Sub Sleep Lib "kernel32" (ByVal dwMilliseconds As Long)
Sleep 500
```

Example Visual Basic code

The .zip file available on the Agilent Web site (www.agilent.com/find/3499example) contains example software in Visual Basic 6.0. When you extract the examples into the root of your C:\ directory, the examples will be placed into the directory:

C:\Program Files\Agilent\IntuiLink\3499\Examples.

Unless otherwise indicated, these examples use the general-purpose set of ActiveX control and I/O objects installed with the IntuiLink for DMM 34401A available at www.agilent.com/find/IntuiLink. These objects are part of the Agilent ISDK for Visual Basic 6.0.

Code directories

EZ_relay - demonstrates how to open and close a relay

MatrixSwitch - a graphical user interface for a matrix card that demonstrates how to open and close a relay, how to read the state of a relay and how to read the module type installed in the 3499A/B/C

MUX_Switch - a graphical user interface for a MUX card that demonstrates how to open and close a relay, how to read the state of a relay and how to read the module type installed in the 3499A/B/C. Requires the N2265A matrix module.

Visual Basic does not implement a delay or wait function. You can implement a delay by declaring the windows Sleep API and then

calling the sleep function. This code will cause a delay of 500 msec:

Switch_DMM_scanExample -

This directory contains several different programs. All are functionally the same, except they are implemented using different I/O in Visual Basic 6.0. The example program demonstrates the three methods described above in this application note

ISDK - uses the general-purpose set of ActiveX control and I/O objects installed with IntuiLink for DMM 34401A, available at www.agilent.com/find/intuilink and provided on the CD-ROM shipped with the 34401A

NI_488 - uses the National Instrument NI-488 syntax that comes with the GPIB interface device

SICL - uses the Agilent SICL syntax that comes with the GPIB interface device.

VISA - uses the industry-standard VISA syntax that comes with many GPIB interface cards

Temperature - demonstrates making temperature measurements using a DMM; contains the equations/routines for thermistor (5k, 10k ohm), RTD, and thermocouple (type J,K,T) measurements

Excel - The ISDK Visual Basic 6.0 example in Switch_DMM_scanExample is duplicated in Excel. Load the Excel file Switch_DMM_scanExample to see the three buttons. To see the example code click on Alt-F11. The module modScanExample is the same as in the ISDK Visual Basic example in the directory Switch_DMM_scanExample.

Summary

Microsoft Visual Basic programming lets you use the 3499A/B/C switch and the 34401A DMM as an effective scanner. The inherent ability of the 3499A/B/C and 34401A to perform bidirectional handshaking simplifies the programming to a few lines of code. For more control by the program, you can limit the handshaking to trigger the DMM only after a relay closure, or to completely control the relay closure and DMM from software.

Examples demonstrate the use of the 3499A/B/C and the 34401A in Microsoft Visual Basic 6.0. The capability of the DMM can be extended to make RTD, thermistor and thermocouple temperature measurements. Temperature conversion algorithms are included in the examples.

Glossary

Thermocouple — a temperature sensor consisting of two dissimilar metals joined together. When a temperature gradient exists (one end is a different temperature from the other), a voltage is generated. Different types of thermocouples are constructed from different metals, with differing temperature ranges and accuracies.

Thermistor — a ceramic temperature sensor that exhibits a change in resistance as a function of temperature. With most thermistors, resistance decreases as the temperature increases.

Related Agilent Literature

Agilent 3499 Switch/Control Unit:

www.agilent.com/find/3499

Agilent 34401A Multimeter:

www.agilent.com/find/34401a

Agilent Email Updates

www.agilent.com/find/emailupdates

Get the latest information on the products and applications you select.

Pathways to Exceptional Test

Let Agilent help you streamline system development and lower the true cost of test. Our world-class measurement science and support can get you accurate results in a hurry. Agilent's instruments are optimized for use in systems, and our open industry software and I/O standards take the hassle out of creating test code. To see how you can get maximum leverage from Agilent's resources, go to www.agilent.com/find/buildyourown

Was This Useful?

If you found the information in this application note useful, feel free to visit these other Agilent Web sites dedicated to helping the engineer who solves difficult test problems: Visit

www.agilent.com/find/appcentral

www.agilent.com/find/connectivity

www.agilent.com/find/adn

By internet, phone, or fax, get assistance with all your test and measurement needs

Online assistance: www.agilent.com/find/assist

Phone or Fax

United States:

(tel) 800 452 4844

Canada:

(tel) 877 894 4414

(fax) 905 282 6495

China:

(tel) 800 810 0189

(fax) 800 820 2816

Europe:

(tel) (31 20) 547 2323

(fax) (31 20) 547 2390

Japan:

(tel) (81) 426 56 7832

(fax) (81) 426 56 7840

Korea:

(tel) (82 2) 2004 5004

(fax) (82 2) 2004 5115

Latin America:

(tel) (305) 269 7500

(fax) (305) 269 7599

Taiwan:

(tel) 0800 047 866

(fax) 0800 286 331

Other Asia Pacific Countries:

(tel) (65) 6375 8100

(fax) (65) 6836 0252

(e-mail) tm_asia@agilent.com

Microsoft is a US registered trademark of Microsoft Corporation.

Product specifications and descriptions in this document subject to change without notice.

© Agilent Technologies, Inc. 2003
Printed in the USA August 27, 2003
5988-9549EN

