GETTING STARTED GUIDE

Measurement Studio

Support for Visual Studio 2013, Visual Studio 2012, and Visual Studio 2010

Measurement Studio is an integrated suite of tools and class libraries designed to help developers create measurement and automation Windows Forms, Windows Presentation Foundation (WPF), and Web Forms applications using Microsoft .NET technologies.

Measurement Studio provides object-oriented measurement hardware interfaces, advanced analysis libraries, scientific user interface controls for Windows Forms, Windows Presentation Foundation (WPF), and Web Forms applications, measurement data networking libraries, project wizards, interactive code designers, and highly extensible .NET classes. You can use Measurement Studio to develop a complete measurement and automation application that includes data acquisition, analysis, and presentation functionalities.

Getting Started with Measurement Studio introduces the concepts associated with the Measurement Studio class libraries and development tools. This document assumes that you have a general working knowledge of Microsoft Visual Studio and the .NET Framework for application development.

Use this manual as a starting point to learn about Measurement Studio. Refer to the *NI Measurement Studio Help* within the Visual Studio environment for function reference and detailed information about the Measurement Studio class libraries, wizards, assistants, and other features.

Contents

Measurement Studio Edition Comparison Chart

The following table lists the features included in the Standard, Professional, and Enterprise editions of Measurement Studio. Refer to *ni.com/mstudio* for more information about the functionality and features included with each Measurement Studio edition.

Table 1. Measurement Studio Edition Comparison Chart

Product Category	Feature	Measurement Studio		
		Standard	Professional	Enterprise
Measurement Studio .NET Class Libraries	Standard Analysis[1]	✓		
	Professional Analysis[1]		✓	
	Enterprise Analysis[1]			✓
	Common	✓	✓	✓
	DataSocket		✓	✓
	Network Variable		✓	✓
	Technical Data Management Streaming (TDMS)		✓	✓
	NI-488.2[2]	✓	✓	✓
	NI-DAQmx[2]	✓	✓	✓
	NI-DMM[2]	✓	✓	✓
	NI-DCPower[2]	✓	✓	✓
	NI-RFSA[2]	✓	✓	✓
	NI-RFSG[2]	✓	✓	✓
	NI-SWITCH[2]	✓	✓	✓
	NI-VISA[2]	✓	✓	✓
	Windows Forms controls	✓	✓	✓
	ASP.NET Web Forms controls		✓	✓
	Windows Presentation Foundation (WPF) controls	✓	✓	✓

Table 1. Measurement Studio Edition Comparison Chart (Continued)

Product Category	Feature	Measurement Studio		
		Standard	Professional	Enterprise
Creating Measurement Studio Projects	Measurement Studio Menu	✓	✓	✓
	Measurement Studio project templates	✓	✓	✓
	.NET Instrument Driver Wizard	✓	✓	✓
	DAQ Assistant		✓	✓
	NI Instrument I/O Assistant[3]		✓	✓
	Parameter Assistant		✓	✓
	Measurement Studio Installer Builder		✓	✓

[1] Refer to *Measurement Studio .NET Class Libraries* for a list of the functionality included in the Analysis class library for each edition of Measurement Studio.

[2] To use the class libraries that interface to National Instruments device drivers, NI-DAQmx, NI-VISA, etc., you must install the underlying device drivers in addition to the .NET class libraries. Refer to the *Driver Support* section for more information.

[3] Measurement Studio 2015 support for Visual Studio 2012 and Visual Studio 2013 does not include the NI Instrument I/O Assistant.

Activating Measurement Studio Licenses

If you did not activate Measurement Studio during installation, you can use the NI Activation Wizard to activate the software after installation. To activate Measurement Studio, you need the serial number printed on the Certificate of Ownership included in your software kit. Refer to the *Licensing, Evaluation, and Activation* topic in the *NI Measurement Studio Help* for more information about how to activate Measurement Studio.

National Instruments offers a variety of Measurement Studio licenses, with certain licensed features available for each license type: Standard Edition, Professional Edition, and Enterprise

Edition. Refer to the *Measurement Studio Editions* topic in the *NI Measurement Studio Help* for more information on Measurement Studio editions, and the *Licensing Measurement Studio* topic in the *NI Measurement Studio Help* for more information about licensing Measurement Studio.

Note To move to a different Measurement Studio edition, you can activate the new edition by using NI License Manager. Launch NI License Manager from the National Instruments folder in the Start menu or from NI Launcher in Windows 8.0/8.1.

For general license activation information, refer to *ni.com/activate*. Refer to *ni.com/mstudio* to purchase a Measurement Studio license. Contact a local National Instruments representative at *ni.com/contact* for more information or for questions about specific licensing needs.

Concurrent Licensing

Measurement Studio supports concurrent licensing as part of a volume license agreement. If your software is a part of a Volume License Agreement (VLA), contact your VLA administrator for installation and activation instructions.

Driver Support

To use .NET class libraries that interface to NI device drivers, you must install the underlying device drivers in addition to the .NET class libraries. You can run the underlying device driver installers from the NI Device Drivers media included with Measurement Studio or you can download the device driver from *ni.com*. Refer to *National Instruments .NET Driver Support* for information on what drivers are supported for each version of the .NET Framework and the download location of the installer files.

Note The DAQ Assistant and the Instrument I/O Assistant are installed from the NI Device Drivers DVD. You must install the NI Device Drivers DVD to use the assistants. The Instrument I/O Assistant is available only with Measurement Studio 2015 support for Visual Studio 2010.

Deployment Requirements

To deploy an application built using Measurement Studio .NET class libraries, refer to the following table to determine the operating system and .NET Framework version required by the target machine.

Measurement Studio Support Used to Build Application	Target Machine	
	Operating Systems	.NET Framework Version
Measurement Studio support for Visual Studio 2013	Windows 8.1/8/7/Server 2008 and 2012 (R2 editions)	4.0, 4.5, or 4.5.1
Measurement Studio support for Visual Studio 2012	Windows 8.1/8/7/Server 2008 and 2012 (R2 editions)	4.0 or 4.5
Measurement Studio support for Visual Studio 2010	Windows 8.1/8/7/XP SP3/ Server 2003, 2008, and 2012 (R2 editions)	4.0

Note Measurement Studio support for Visual Studio 2013, Visual Studio 2012, and Visual Studio 2010 uses the same `NationalInstruments.Common.dll` assembly, which is built against the .NET Framework version 4.0.

Deploying an Application

National Instruments recommends using the Measurement Studio Installer Builder to deploy your Measurement Studio application. For more information on deploying applications, refer to the *Measurement Studio Installer Builder Help*, which is accessible from within the Installer Builder application.

Deploying 64-bit Applications

All Measurement Studio class libraries that support 64-bit include both 32-bit and 64-bit deployment merge modules or runtime installers. This is true regardless of whether the class library includes platform-specific (i.e., x86 or x64) or platform-agnostic (i.e., Any CPU) assemblies. Refer to *Deploying Windows Applications* in the installed *NI Measurement Studio Help* for more information on using 64-bit merge modules or runtime installers.

Learning Measurement Studio

As you work with Measurement Studio, you might need to consult additional resources. For detailed Measurement Studio help, including function reference and in-depth documentation on developing with Measurement Studio, refer to the *NI Measurement Studio Help* within the Visual Studio environment.

The *NI Measurement Studio Help* is integrated into the Visual Studio help. To view the integrated *NI Measurement Studio Help*, you must have the local Visual Studio help installed and your Help settings set to launch the help in the local help viewer.

You can view the *NI Measurement Studio Help* from within Microsoft's Help Viewer by:

- Launching the Help Viewer from the Measurement Studio folder within the NI Launcher,
- Selecting **NI Measurement Studio Help** from the Measurement Studio menu within Visual Studio, or
- Selecting **Help»View Help** from the Visual Studio menu.

Where to Go Next

The following resources are also available to provide you with information about Measurement Studio:

- Getting Started information—Refer to the installed *NI Measurement Studio Help* topic to get started with Measurement Studio. For a list of Measurement Studio resources, refer to the *Using the Measurement Studio Help* topic in the *NI Measurement Studio Help*.
- Examples—Measurement Studio installs examples organized by class library, depending on the component, the version of Visual Studio or the .NET Framework that the example supports, the version of Measurement Studio installed on the system, and the operating system. For more information on example locations, refer to the *Where to Find Examples* topic in the *NI Measurement Studio Help*.
- Measurement Studio website, *ni.com/mstudio*—Contains Measurement Studio news, support, downloads, white papers, and product tutorials.
- NI Technical Support, *ni.com/support*—Provides access to online example programs, tutorials, technical news, and Measurement Studio discussion forums.
- Review the information from the Microsoft website on using Visual Studio.

How to Create Applications with Measurement Studio Controls and Class Libraries

The following online topics include overview information and step-by-step instructions on developing applications with Measurement Studio tools and features. Refer to the *Developing Projects with Measurement Studio* section of the *NI Measurement Studio Help* for more information about the functionality of these tools and features.

Measurement Studio Support for Visual Studio 2013 Walkthroughs

- *Creating a Measurement Studio Application with Windows Forms Controls and Analysis*
- *Creating a Measurement Studio Application with Windows Forms Controls and Network Variable*
- *Creating a Measurement Studio NI-DAQmx Application*

Measurement Studio Support for Visual Studio 2012 Walkthroughs

- *Creating a Measurement Studio Application with Windows Forms Controls and Analysis*
- *Creating a Measurement Studio Application with Windows Forms Controls and Network Variable*
- *Creating a Measurement Studio NI-DAQmx Application*

Measurement Studio Support for Visual Studio 2010 Walkthroughs

- *Creating a Measurement Studio Application with Windows Forms Controls and Analysis*
- *Creating a Measurement Studio Application with Web Forms Controls and Analysis*
- *Creating a Measurement Studio Application with Windows Forms Controls and Network Variable*
- *Creating a Measurement Studio Application with Web Forms Controls and Network Variable*
- *Creating a Measurement Studio NI-DAQmx Application*
- *Creating a Measurement Studio Instrument I/O Application*

Measurement Studio .NET Class Libraries

This section provides overview information about the .NET class libraries included with Measurement Studio. Refer to the *Using the Measurement Studio .NET Class Libraries* section of the *NI Measurement Studio Help* for detailed information about these libraries.

Measurement Studio Support for Visual Studio .NET Class Library Overview

Measurement Studio provides .NET class libraries that you can use to develop complete measurement and automation applications in Visual Basic .NET and Visual C#.

Measurement Studio includes the following .NET class libraries:

- Analysis
- Common
- Data Socket
- Network Variable
- Technical Data Management Streaming (TDMS)
- User Interface

The following .NET class libraries that interface with National Instruments device drivers are also available on the NI Device Drivers DVD or online from *ni.com/drivers*:

- NI-488.2
- NI-DAQmx
- NI-DCPower
- NI-DMM
- NI-RFSA
- NI-RFSG
- NI-SCOPE
- NI-SWITCH
- NI-VISA

Refer to the following sections for information about each Measurement Studio .NET class library.

64-bit Support in Measurement Studio .NET Assemblies

You can use Measurement Studio .NET assemblies in 64-bit applications to take advantage of the increased processing power and memory capabilities that are available to 64-bit applications. For more information about working with 64-bit class libraries, review the following:

- Measurement Studio includes 64-bit support for all class libraries except DataSocket .NET class libraries and modular instruments .NET wrappers for other NI drivers. For a list of hardware class library support, refer to the *Driver Support* section.
- Projects created from project templates included with Measurement Studio 8.6.1 or earlier will run as 32-bit unless you manually change the project settings to AnyCPU.
- Not all NI-DAQ hardware supported by NI-DAQmx is supported in 64-bit user mode. Refer to the *NI-DAQ Readme* for more information, installed at **Start»All Programs»National Instruments»NI-DAQ»NI-DAQ Readme**.

Analysis

The Measurement Studio Analysis .NET class library is in the `NationalInstruments.Analysis` namespace. The Analysis class library includes a set of classes that provides various digital signal processing, signal filtering, signal generation, peak detection, and other general mathematical functionality. Use this library to analyze acquired data or to generate data. Additionally, the documentation for the Analysis class library includes analysis code snippets that you can copy and paste into an application and use immediately.

The functionality included in the Analysis class library varies based on the Measurement Studio edition you purchase. Refer to the following sections for information about the Standard, Professional, and Enterprise Analysis class libraries.

Standard Analysis

The Standard Analysis class library, which ships with Measurement Studio Standard Edition, includes the sawtooth, sine, square, triangle, and basic function wave generators.

Professional Analysis

The Professional Analysis class library, which ships with Measurement Studio Professional Edition, includes the Standard Analysis functionality as well as the following functionality:

- Signal processing functions such as convolution, deconvolution, correlation, decimation, integration, and differentiation
- FFT, Inverse FFT, Real FFT, Fast Hartley, Inverse Fast Hartley, Fast Hilbert, Inverse Fast Hilbert, DST, Inverse DST, DCT, and Inverse DCT transformations
- Lowpass, Highpass, Bandpass, and Bandstop types of Bessel, Chebyshev, Inverse Chebyshev, Windowed, Kaiser, and Elliptic filters for real and complex data
- Narrowband FIR filters for real and complex data
- Linear algebra functions such as determinant, check positive definiteness, calculate dot product, and other various matrix functions
- Scaled and unscaled windowing classes for real and complex data
- Common statistical functions such as mean, median, mode, and variance
- Exponential, linear, and polynomial curve fitting functions
- Signal generation functions
- Basic level trigger detection

Enterprise Analysis

The Enterprise Analysis class library, which ships with Measurement Studio Enterprise Edition, includes the Standard and Professional Analysis functionality as well as the following advanced functionality:

- EquiRipple filters for real and complex data
- Linear algebra functions such as forward and back substitution, LU factorization, Cholesky factorization, Schur decomposition, and Hessenberg decomposition
- Probability and analysis of variance
- Sine, impulse, pulse, ramp, and chirp patterns
- General least square curve fit, power fit, log fit, Gauss fit, cubic spline fit, and interpolation functions
- Measurement functions such as transition measurements, pulse measurements, cycle RMS average, and single tone and multiple tone information
- Special functions

Refer to Table 2 to determine the type of measurements available in the Professional and Enterprise Analysis .NET libraries.

Table 2. Analysis .NET Library Measurement Types included in the Professional and Enterprise Packages

Analysis .NET Library	Professional Package	Enterprise Package
Measurements		
AC and DC Estimator		✓
Amplitude and Phase Spectrum		✓
Auto Power Spectrum		✓
Cross Power Spectrum		✓
Cycle RMS Average		✓
Harmonic Analyzer		✓
Harmonic Analyzer Using Signal		✓
Impulse Response Function	✓	✓
Network Functions (avg)	✓	✓
Power and Frequency Estimate		✓
Pulse Measurements		✓
Scaled Time Domain Window		✓
Single Tone Information		✓
Multiple Tone Information		✓
Spectrum Unit Conversion		✓
State Levels		✓
Transfer Function		✓
Transition Measurements		✓
Signal Generation		
Arbitrary Wave	✓	✓
Array Riffle		✓
Bernoulli Noise	✓	✓
Binomial Noise	✓	✓
Chirp Pattern		✓

Table 2. Analysis .NET Library Measurement Types included in the Professional and Enterprise Packages (Continued)

Analysis .NET Library	Professional Package	Enterprise Package
Gamma Noise	✓	✓
Gaussian White Noise	✓	✓
Halton Sequence	✓	✓
Impulse Pattern		✓
Periodic Noise	✓	✓
Poisson Noise	✓	✓
Pulse Pattern		✓
Ramp Pattern		✓
Richtmeyer Sequence	✓	✓
Sawtooth Wave		✓
Sine Pattern	✓	✓
Sine Wave	✓	✓
Square Wave	✓	✓
Triangle Wave	✓	✓
Uniform White Noise	✓	✓
Windowing		
Blackman Window	✓	✓
Blackman-Harris Window	✓	✓
Blackman-Nuttall Window	✓	✓
Cosine Tapered Window	✓	✓
Dolph-Chebyshev Window	✓	✓
Exact Blackman Window	✓	✓
Exponential Window	✓	✓
Flat Top Window	✓	✓
Force Window	✓	✓

Table 2. Analysis .NET Library Measurement Types included in the Professional and Enterprise Packages (Continued)

Analysis .NET Library	Professional Package	Enterprise Package
Gauss Window	✓	✓
General Cosine Window	✓	✓
Hamming Window	✓	✓
Hanning Window	✓	✓
Kaiser-Bessel Window	✓	✓
Scaled Time Domain Windows	✓	✓
Symmetric Time Domain Windows	✓	✓
Triangle Window	✓	✓
Filters		
Bessel	✓	✓
Butterworth	✓	✓
Cascade	✓	✓
Chebyshev	✓	✓
Elliptic	✓	✓
Equiripple		✓
FIR	✓	✓
FIR Windowed	✓	✓
IIR Cascade	✓	✓
IIR	✓	✓
Inverse Chebyshev	✓	✓
Kaiser	✓	✓
Narrowband FIR	✓	✓
Signal Processing		
Autocorrelation	✓	✓
Basic Level Trigger Detection	✓	✓

Table 2. Analysis .NET Library Measurement Types included in the Professional and Enterprise Packages (Continued)

Analysis .NET Library	Professional Package	Enterprise Package
Convolution	✓	✓
Cross Power	✓	✓
Cross Correlation	✓	✓
Decimate	✓	✓
Deconvolution	✓	✓
Derivative x(t)	✓	✓
Discrete Cosine Transform	✓	✓
Discrete Sine Transform	✓	✓
Fast Hilbert Transform	✓	✓
Fast Hartley Transform	✓	✓
Integral x(t)	✓	✓
Inverse Real and Complex Fast Fourier Transform (FFT)	✓	✓
Inverse Fast Hilbert Transform	✓	✓
Inverse Fast Hartley Transform	✓	✓
Peak Detection	✓	✓
Power Spectrum	✓	✓
Pulse Parameters	✓	✓
Real and Complex FFT	✓	✓
Threshold Peak Detector	✓	✓
Unwrap Phase	✓	✓
Linear Algebra		
Back Transform Eigen Vectors		✓
Backward Substitution		✓
Cholesky Factorization		✓

Table 2. Analysis .NET Library Measurement Types included in the Professional and Enterprise Packages (Continued)

Analysis .NET Library	Professional Package	Enterprise Package
Complex Back Transform Eigen Vectors		✓
Complex Cholesky Factorization		✓
Complex Determinant	✓	✓
Complex Dot Product	✓	✓
Complex Eigen Vectors and Eigen Values		✓
Complex General Eigen AB		✓
Complex Hessenberg Decomposition		✓
Complex Inverse Matrix		✓
Complex Linear Equations		✓
Complex LU Factorization		✓
Complex Matrix Balance		✓
Complex Matrix Condition Number	✓	✓
Complex Matrix Norm	✓	✓
Complex Matrix Rank	✓	✓
Complex Outer Product	✓	✓
Complex Pseudo Inverse Matrix	✓	✓
Complex QR Factorization		✓
Complex QR Factorization with Pivot Matrix		✓
Complex QR Factorization with Pivot Vector		✓
Complex QZ Decomposition		✓
Complex Schur Decomposition		✓
Complex Solve Linear Equations (Multiple Right Hand)		✓

Table 2. Analysis .NET Library Measurement Types included in the Professional and Enterprise Packages (Continued)

Analysis .NET Library	Professional Package	Enterprise Package
Complex Solve Linear Equations (Single Right Hand)		✓
Complex SVD Factorization		✓
Complex Vector Norm		✓
Determinant	✓	✓
Dot Product	✓	✓
Forward Substitution		✓
General Eigen AB		✓
Hessenberg Decomposition		✓
Inverse Matrix	✓	✓
Linear Equations		✓
LU Factorization		✓
Matrix Balance		✓
Matrix By Vector Multiplication	✓	✓
Matrix Condition Number	✓	✓
Matrix Multiplication	✓	✓
Matrix Norm	✓	✓
Matrix Rank	✓	✓
Outer Product	✓	✓
Pseudo Inverse Matrix	✓	✓
QR Factorization		✓
QR Factorization with Pivot Matrix		✓
QR Factorization with Pivot Vector		✓
QZ Decomposition		✓
Schur Decomposition		✓

Table 2. Analysis .NET Library Measurement Types included in the Professional and Enterprise Packages (Continued)

Analysis .NET Library	Professional Package	Enterprise Package
Solve Linear Equations (Multiple Right Hand)		✓
Solve Linear Equations (Single Right Hand)		✓
Special Matrix	✓	✓
SVD Factorization		✓
Test Positive Definite Matrix	✓	✓
Trace	✓	✓
Transpose	✓	✓
Array and Numeric Operations		
1D and 2D Array Arithmetic	✓	✓
1D and 2D Linear Evaluation	✓	✓
1D and 2D Polynomial Evaluation	✓	✓
1D Polar to Rectangular	✓	✓
1D Rectangular to Polar	✓	✓
Complex Number Arithmetic	✓	✓
Find Polynomial Roots	✓	✓
Scale 1D and 2D	✓	✓
Curve Fitting		
Cubic Spline Fit		✓
Exponential Fit	✓	✓
Exponential Fit Interval		✓
Gauss Fit		✓
Gauss Fit Interval		✓
General Least Squares Linear Fit		✓

Table 2. Analysis .NET Library Measurement Types included in the Professional and Enterprise Packages (Continued)

Analysis .NET Library	Professional Package	Enterprise Package
General Polynomial Fit	✓	✓
Goodness of Fit		✓
Linear Fit	✓	✓
Linear Fit Interval		✓
Logarithm Fit		✓
Logarithm Fit Interval		✓
Nonlinear Fit		✓
Power Fit		✓
Power Fit Interval		✓
Remove Outliers		✓
Statistics		
1D, 2D, and 3D ANOVA		✓
Chi-Square Distribution		✓
erf(x) and erfc(x)		✓
F-Distribution		✓
Histogram	✓	✓
Inverse Chi-Square Distribution		✓
Inverse F-Distribution		✓
Inverse Normal Distribution		✓
Inverse T-Distribution		✓
Mean	✓	✓
Median and Mode	✓	✓
Moment about Mean	✓	✓
Normal Distribution		✓
Polynomial Interpolation		✓

Table 2. Analysis .NET Library Measurement Types included in the Professional and Enterprise Packages (Continued)

Analysis .NET Library	Professional Package	Enterprise Package
Root-Mean_Square (RMS)	✓	✓
Spline Interpolant		✓
Spline Interpolation		✓
Standard Deviation	✓	✓
T-Distribution		✓
Variance		✓
Special Functions		
Airy		✓
Bessel 1st		✓
Bessel 2nd		✓
Beta		✓
Complimentary Gamma		✓
Cosine Integral		✓
Dawson’s Integral		✓
Dilogarithm		✓
Elliptic 1st		✓
Elliptic 2nd		✓
Exponential Integral		✓
Factorial		✓
Fresnel Integrals		✓
Gamma		✓
Gauss Hypergeometric		✓
Hyperbolic Cosine Integral		✓
Hyperbolic Sine Integral		✓
Incomplete Beta		✓

Table 2. Analysis .NET Library Measurement Types included in the Professional and Enterprise Packages (Continued)

Analysis .NET Library	Professional Package	Enterprise Package
Incomplete Elliptic 1st		✓
Incomplete Elliptic 2nd		✓
Incomplete Gamma		✓
Jacobian Elliptic Function		✓
Kelvin 1st		✓
Kelvin 2nd		✓
Kummer		✓
Logarithm of Factorial		✓
Modified Bessel 1st		✓
Modified Bessel 2nd		✓
Parabolic Cylinder		✓
Psi		✓
Sine Integral		✓
Spherical Bessel 1st		✓
Spherical Bessel 2nd		✓
Stirling		✓
Struve		✓
Tricomi		✓
Zeta		✓

Tip For more information about analyzing or generating data with the Analysis class library, refer to the *Using the Measurement Studio Analysis .NET Library* topic in the *NI Measurement Studio Help*. For more information about the functionality included in the Analysis class library, visit *ni.com/analysis*.

Common

The Measurement Studio Common .NET class library is in the `NationalInstruments` namespace. The Common class library provides a set of classes that facilitates the exchange of data between the acquisition, analysis, and user interface portions of your application. The Common class library includes the following features:

- A `ComplexSingle` data type. This data type represents a complex number of type `Single` that is composed of two single-precision floating-point numbers.
- A `ComplexDouble` data type. This data type represents a complex number of type `Double` that is composed of a real part and an imaginary part.
- A `ComplexInt16` data type. This data type represents a complex number of type `Int16` that is composed of a real part and an imaginary part.
- A `DigitalWaveform` data type. This data type represents a set of digital states that are grouped by samples or signals.
- A `ComplexWaveform` data type. This data type represents an analog signal that varies over time and is composed of complex data values.
- A `ComplexWaveformCollection` data type. This data type represents a strongly typed collection of `ComplexWaveform<TData>` objects. This data type encapsulates raw complex data, scaling information, and timing information.
- An `AnalogWaveform` data type. This data type represents an analog signal that varies over time.
- A `DataConverter` class that converts data from one data type to another data type, such as converting an array of integers to an array of doubles.
- An `EngineeringFormatInfo` class that defines a custom formatter to format numeric values as strings with engineering notation and International System of Units (SI) prefixes and symbols.
- A `PrecisionDateTime` structure that you can use to represent an instance in time, typically expressed as a date and time of day, that is accurate to the nearest 2^{-64} second.
- A `PrecisionTimeSpan` structure that you can use to represent a time interval, or duration of time, that is accurate to the nearest 2^{-64} second.
- A `PrecisionWaveformTiming` class that you can use to represent the timing of an analog or digital waveform that is accurate to the nearest 2^{-64} second.
- An `AnalogWaveformCollection` class that contains a strongly typed collection of `AnalogWaveform<TData>` objects; one object for each channel and record combination. You can access these objects through the 1D indexer or the 2D indexer.

Tip For more detailed information about the Common class library, refer to the *National Instruments* section in the *NI Measurement Studio Help*.

Data Transfer

You can use the NetworkVariable class library or the DataSocket class library to transfer live measurement data between applications over a network. You can use NetworkVariable or DataSocket to exchange different types of data between Measurement Studio, LabVIEW, LabWindows/CVI, and other applications that support NI-Publish Subscribe Protocol (`psp:`). NetworkVariable is the preferred method for transferring data between these applications, and, in these cases, NetworkVariable supersedes DataSocket. You can also use NetworkVariable and DataSocket to exchange different types of data between OLE for Process Control (`opc:`) servers. Exchanging data between Measurement Studio applications and OPC servers with

NetworkVariable requires the LabVIEW DSC Run-Time System. Use DataSocket to communicate directly with an OPC server.

Network Variable

The Measurement Studio Network Variable .NET class library includes three namespaces: `NationalInstruments.NetworkVariable`, `NationalInstruments.NetworkVariable.WindowsForms`, and `NationalInstruments.NetworkVariable.WebForms`. You use the Network Variable class library to transfer live measurement data between applications and servers over the network. You use `NationalInstruments.NetworkVariable.WindowsForms` and `NationalInstruments.NetworkVariable.WebForms` data sources to expose Network Variable data items that you can bind to properties of a Windows Forms or a Web Forms control.

Use the features in the Network Variable class library to perform the following operations:

- Exchange different types of data between Measurement Studio, LabVIEW, LabWindows/CVI, and other applications that support NI-Publish Subscribe Protocol (`psp:`) and OLE for Process Control (`opc:`) servers. Exchanging data between Measurement Studio applications and OPC servers requires LabVIEW DSC.

Note Measurement Studio and LabWindows/CVI refer to variables as network variables and LabVIEW refers to variables as shared variables. However, you can read to and write from Measurement Studio and LabWindows/CVI network variables with LabVIEW shared variables.

- Use Windows Forms and Web Forms data sources to expose Network Variable data items that you can bind to properties of a Windows Forms or a Web Forms control.
- Use the `NationalInstruments.NetworkVariable.ServerProcess`, `NationalInstruments.NetworkVariable.ServerProcessInfo`, `NationalInstruments.NetworkVariable.ServerVariable`, and `NationalInstruments.NetworkVariable.ServerVariableInfo` classes to explicitly create network variables. You can configure the description and type for explicitly created network variables.
- Use the Network Variable Browser dialog box to quickly locate and select data items on other computers and servers. The Browser Dialog is included in the WindowsForms class.

Tip For more detailed information about the Network Variable class library, refer to the *Using the Measurement Studio Network Variable .NET Library* section in the *NI Measurement Studio Help*.

DataSocket

The Measurement Studio Data Socket .NET class library is in the `NationalInstruments.Net` namespace. Use the DataSocket class library to transfer live measurement data over the Internet or an intranet, between applications on the same computer, and to and from files. Use the classes in the Data Socket class library to perform the following operations:

- Read and write data between different data sources and targets.
- Use a single, simple API to communicate with several types of servers, including DataSocket Servers (`dstp:`), web servers (`http:`), file transfer protocol servers (`ftp:`), file systems (`file:`), and OLE for Process Control (`opc:`) servers.
- Specify data sources and targets using a URL, the same way you access web pages in a web browser.
- Use DataSocket Transfer Protocol (DSTP) to exchange different types of data.
- Expose DataSocket data items as data sources that you can bind to properties of a Windows Forms control.
- Interactively browse to quickly locate and select data items on other computers and servers.

Note The Measurement Studio Data Socket .NET class libraries do not include 64-bit support.

Tip For more detailed information about the Data Socket class library, refer to the *Using the Measurement Studio Data Socket .NET Library* section in the *NI Measurement Studio Help*.

NI-488.2

The Measurement Studio NI-488.2 .NET class library is in the `NationalInstruments.NI4882` namespace. This class library is included when you install the NI-488.2 driver. The NI-488.2 driver is available at *ni.com/downloads* and on the NI Device Drivers DVD that is included with Measurement Studio. The NI-488.2 class library includes a set of classes for communicating with GPIB instruments, controlling GPIB devices, and acquiring GPIB status information. Use this library to design code that communicates with and controls instruments on a GPIB interface. Use the NI-488.2 class library to configure and communicate with GPIB devices using the `Device` and `Board` classes.

Tip For more information about the NI-488.2 class library, refer to the *Using the Measurement Studio NI-488.2 .NET Library* topic in the *NI Measurement Studio Help*. For more information about GPIB, visit *ni.com/gpib*.

NI-DAQmx

The Measurement Studio NI-DAQmx .NET class library is in the `NationalInstruments.DAQmx` namespace. This class library is included when you install the NI-DAQmx driver. The NI-DAQmx driver is available at *ni.com/downloads* and on the NI Device Drivers DVD that is included with Measurement Studio. Use the NI-DAQmx class library to communicate with and control NI data acquisition (DAQ) devices.

Note Not all NI-DAQ hardware supported by NI-DAQmx is supported in 64-bit user mode. Refer to the *NI-DAQ Readme* for more information, installed at **Start»All Programs»National Instruments»NI-DAQ»NI-DAQ Readme**.

Use the NI-DAQmx class library to perform the following types of tasks:

- Analog signal measurement
- Analog signal generation
- Digital I/O
- Counting and timing
- Pulse generation
- Signal switching

Tip For more information about the NI-DAQmx class library, refer to the *Using the Measurement Studio NI-DAQmx.NET Library* topic in the *NI Measurement Studio Help*. For more information about DAQ, visit *ni.com/dataacquisition*.

NI-DC Power

Note NI-DCPower .NET support is available only with Visual Studio 2010.

The Measurement Studio NI-DCPower .NET class library is in the `NationalInstruments.ModularInstruments.NIDCPower` namespace. This class library is included when you install the underlying NI-DCPower driver and the NI-DCPower .NET class libraries. Both the driver and the class library installers are available on *ni.com/drivers*. The NI-DCPower driver is also available on the NI Device Drivers DVD that is included with Measurement Studio. Use the NI-DCPower class library to communicate with and control NI DCPower Supplies and SMUs.

NI-DMM

Note NI-DMM .NET support is available only with Visual Studio 2010.

The Measurement Studio NI-DMM .NET class library is in the `NationalInstruments.ModularInstruments.NIDMM` namespace. This class library is included when you install the underlying NI-DMM driver and the NI-DMM .NET class libraries. Both the driver and the class library installers are available on *ni.com/drivers*. The NI-DMM driver is also available on the NI Device Drivers DVD that is included with Measurement Studio. Use the NI-DMM class library to communicate with and control NI Digital Multimeter devices.

NI-RFmx

Note NI-RFmx .NET support is available only with Visual Studio 2010.

Use the NI-RFmx class libraries to configure and operate NI RF spectral analysis hardware and vector signal transceivers. The Measurement Studio NI-RFmx .NET class libraries include the `NationalInstruments.RFmx.InstrMX`, `NationalInstruments.RFmx.DemodMX`, and `NationalInstruments.RFmx.SpecAnMX` class libraries. These class libraries are included when you install the underlying NI-RFmx driver and the NI-RFmx .NET class libraries. Both the driver and the class library installers are available on *ni.com/drivers*. The NI-RFmx driver is also available on the NI Device Drivers DVD that is included with Measurement Studio.

NI-RFSA

Note NI-RFSA .NET support is available only with Visual Studio 2010.

The Measurement Studio NI-RFSA .NET class library is in the `NationalInstruments.ModularInstruments.NIRFSA` namespace. This class library is included when you install the underlying NI-RFSA driver and the NI-RFSA .NET class libraries. Both the driver and the class library installers are available on *ni.com/drivers*. The NI-RFSA driver is also available on the NI Device Drivers DVD that is included with Measurement Studio. Use the NI-RFSA class library to configure and operate NI RF signal analysis hardware and to perform waveform and spectrum analysis.

NI-RFSG

Note NI-RFSG .NET support is available only with Visual Studio 2010.

The Measurement Studio NI-RFSG .NET class library is in the `NationalInstruments.ModularInstruments.NIRFSG` namespace. This class library is included when you install the underlying NI-RFSG driver and the NI-RFSG .NET class libraries. Both the driver and the class library installers are available on *ni.com/drivers*. The NI-RFSG driver is also available on the NI Device Drivers DVD that is included with Measurement Studio. Use the NI-RFSG class library to configure and operate NI RF signal generators.

NI-SWITCH

Note NI-SWITCH .NET support is available only with Visual Studio 2010.

The Measurement Studio NI-SWITCH .NET class library is in the `NationalInstruments.ModularInstruments.NISWITCH` namespace. This class library is included when you install the underlying NI-SWITCH driver and the NI-SWITCH .NET class libraries. Both the driver and the class library installers are available on *ni.com/drivers*. The NI-SWITCH driver is also available on the NI Device Drivers DVD that is included with Measurement Studio. Use the NI-SWITCH class library to communicate with and control NI SWITCH devices.

NI-VISA

The Measurement Studio NI-VISA .NET class library is in the `NationalInstruments.VisaNS` namespace. This class library is included when you install the NI-VISA driver. The NI-VISA driver is available at *ni.com/downloads*. The NI-VISA class library includes a set of classes that provides a rich, object-oriented interface to the NI-VISA driver. Use this library to quickly create bus-independent or bus-specific instrument control applications.

The NI-VISA class library supports formatted I/O operations, locking, event handling, and interface-specific extensions. With this class library you can access the functionality available in NI-VISA for communicating with message-based and register-based instruments using the following interfaces:

- GPIB
- IEEE 1394
- PXI
- Serial (RS-232 and RS-485)
- TCP/IP
- USB
- VXI

Note The Instrument I/O Assistant is available only with Measurement Studio support for Visual Studio 2010. For more information about NI-VISA, visit *ni.com/visa*.

Technical Data Management Streaming (TDMS)

Technical Data Management Streaming (TDMS) is a file format based on the National Instruments TDM data model used to stream data to disk. You can use the TDMS .NET class library to describe, store, and read measurement data that is optimized for high-speed data streaming and post processing. Additionally, you can use the TDMS .NET class library to create files that you can use in LabVIEW, LabWindows/CVI, and DIAdem, and files created by these applications can be used by the TDMS .NET class library.

You can use the Measurement Studio TDMS .NET class library to perform the following operations:

- Read and write array data in a structured format from and to a `.tdms` file.
- Read and write analog waveform data or digital waveform data, including timing information, from and to a `.tdms` file.
- Use the `TdmsProperty` class to create custom properties for each level of the hierarchy by defining a name, data type, and value.

Tip For more detailed information about the TDMS class library, refer to the *Using the Measurement Studio TDMS .NET Library* section in the *NI Measurement Studio Help*.

TDM Excel Add-In

You can use the TDM Excel Add-In to load NI `.tdm` and `.tdms` files into Microsoft Excel. Use the toolbar from within Excel to choose which properties are loaded into Excel at the file, group, and channel levels, including custom properties.

User Interface

The Measurement Studio user interface controls are in the Windows Presentation Foundation (WPF), Windows Forms, and Web Forms .NET class libraries. The following sections list the functionality included with the Measurement Studio WPF, Windows Forms, and Web Forms controls.

Windows Presentation Foundation (WPF) Controls

The WPF .NET class libraries are in the following namespaces:

- `NationalInstruments.Controls`
- `NationalInstruments.Controls.Converters`
- `NationalInstruments.Controls.Data`
- `NationalInstruments.Controls.Primitives`
- `NationalInstruments.Controls.Rendering`
- `NationalInstruments.DataInfrastructure`
- `NationalInstruments.DataInfrastructure.Descriptors`
- `NationalInstruments.DataInfrastructure.Primitives`
- `NationalInstruments.Logging`

The WPF class libraries encapsulate the following Measurement Studio user interface controls:

- Graph
- Writable graph
- Intensity graph
- Polar graph
- Digital Graph
- Mixed Graph
- Legend
- Graph interaction palette
- Knob
- Gauge
- Meter
- Slider
- Tank
- Numeric Text Box
- Radix Numeric Text Box
- Boolean Button
- Boolean Content Button
- Arrow Button
- Power Button
- Switch
- LED
- TimeStamp TextBox

Use these class libraries to add measurement-specific user interface controls to your application. You can configure the controls programmatically at design time with code or

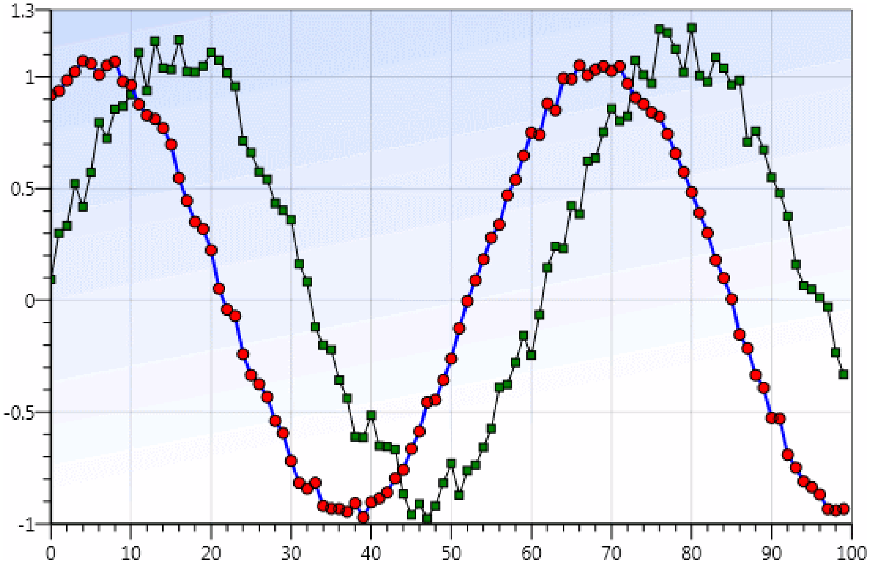
XAML or through the Properties window in the WPF design surface. The following sections describe each of the Measurement Studio WPF user interface controls.

Tip For more information about using the .NET user interface controls, refer to the *Using the Measurement Studio WPF .NET Controls* section in the *NI Measurement Studio Help*.

Graph and Writable Graph Controls

Use the Measurement Studio graph and writable graph controls to display two-dimensional Cartesian data that requires two axes. You can plot and chart with the graph and writable graph controls, and you can display data using lines, points, bars, and areas. The graph and writable graph controls support multiple plots. In addition to plotting and charting, the writable graph provides your end users with the ability to draw data into the graph.

Figure 1. Graph Control with Line and Point Renderers



With the graph and writable graph controls and the classes that interface with the controls, you can perform the following operations:

Plot Operations

- Plot and chart arrays of various types of scalar numeric values, such as double-precision floating point values; analog waveforms; and complex waveforms.
- Configure a graph to contain multiple plots to show separate but related data on the same graph.
- You can configure a graph to automatically render different styles of plots, such as line, bar, or point, on the same graph using a `DefaultPlotRenderers` collection.
- Draw bars or fills from a plot to positive infinity, negative infinity, or zero.

- Specify vector, raster, or hardware rendering for display, or set the graph to automatically switch between vector, raster, and hardware as needed.
- Keep track of how your signal is changing over time using the phosphor effect.

Axis Operations

- Configure a graph to include multiple axes or independent ranges so that plot data fits the graph plot area.
- Configure the axis to automatically scale to show data being plotted or charted.
- Use logarithmic axes with configurable bases.
- Interactively change the range of an axis at run time.
- Display labels on the axis.
- Display grid lines.
- Position the axis to display on one or both sides of the graph's plot area.
- Configure major, minor and custom divisions.

Cursor Operations

- Use cursors, multi-plot cursors, and range cursors to identify key points, values, and ranges in plots and the plot area.
- Determine which plots or axes a cursor can interact with by configuring the relevant properties.
- Use cursor labels to display X and Y data coordinates, values, or ranges in a customized format that the cursor crosshair points to, and customize the appearance of the label.
- Interactively move the cursor by clicking and dragging the vertical or horizontal crosshair or the center of the cursor.
- Programmatically move the cursor to a specified index, data value, or screen position.
- Create cursors programmatically or at design time using the graph's Children collection editor.

Annotation Operations

- Configure text labels, arrows, and shapes to annotate a point in the plot area of the graph.
- Configure range area, text labels, and arrows to annotate a range in the plot area of the graph.

Input Operations (Writable Graph only)

- Draw data into the writable graph using EditGraphInteraction at run time.
- Customize the appearance of drawn plots.
- Determine where drawn data appears on the graph with SelectedInputPlot.

Additional Operations

- Choose graph interaction at run time with the Graph Interaction Palette.
- Pan and zoom interactively, as well as programmatically.
- Perform hit testing of mouse cursor coordinates.
- Pan with scroll bars.

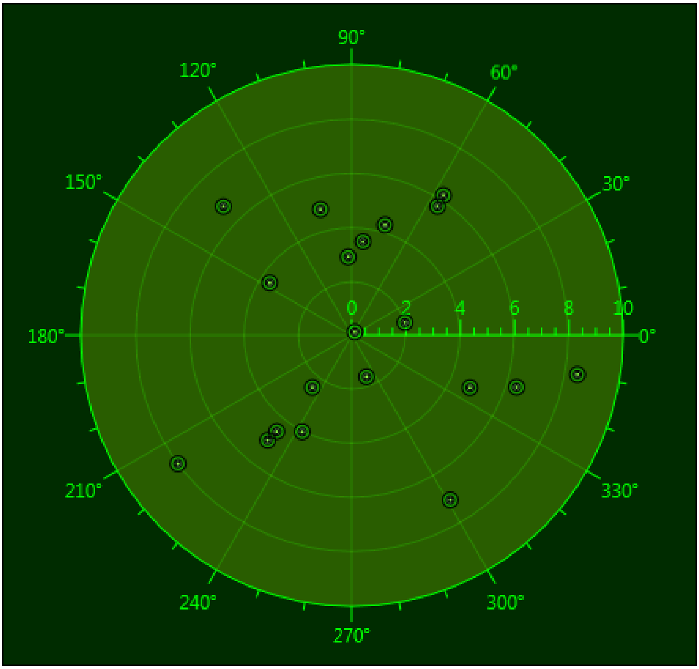
- Configure the style and mode of scroll bars.
- Bind a graph to a data source.

Tip For more information about using the graph and writable graph controls, refer to the *Measurement Studio WPF Graph Controls Overview* section in the *NI Measurement Studio Help*.

Polar Graph Control

Use the Measurement Studio polar graph control to display 2D complex or point data. For point data, the polar coordinates are given as (theta, *r*), where theta represents the measure of the angle and r represents the distance of the point from the pole. For complex data, polar coordinates are given as (phase, magnitude), where phase represents the measure of the angle and magnitude represents the distance of the point from the pole.

Figure 2. Polar Graph Control



With the polar graph control and the classes that interface with the control, you can perform the following operations:

Plot Operations

- Plot and chart `ComplexSingle`, `ComplexDouble`, and `ComplexInt16` data, along with data types supported by other graph controls.
- Configure a graph to contain multiple plots to show separate but related data on the same graph.

- Configure a graph to automatically generate different colors and shapes for different plots using a `DefaultPlotRenderers` collection.
- Create custom point and line styles for plots.

Axis Operations

- Configure the radius axis modes to: fixed or autoscaling, including autoscaling based on the visible data only.
- Interactively change the range of the radius axis by clicking on the axis end label.
- Display grid lines.
- Configure major, minor, and custom divisions.
- Display labels on the axis.

Cursor Operations

- Configure the graph to display cursors that are used to determine the magnitude and phase data coordinates of a point on the plot area.
- Configure the graph to display multi-plot cursors that are used to determine the magnitude or phase values on the plot area.
- Determine which plots a cursor can interact with by configuring the relevant properties.
- Use cursor labels to display angle and radius data coordinates or values that the cursor crosshair points to, and customize the appearance of the label.
- Create custom point and line styles for cursors.
- Create cursors programmatically or at design time using the graph's Children collection editor.

Additional Operations

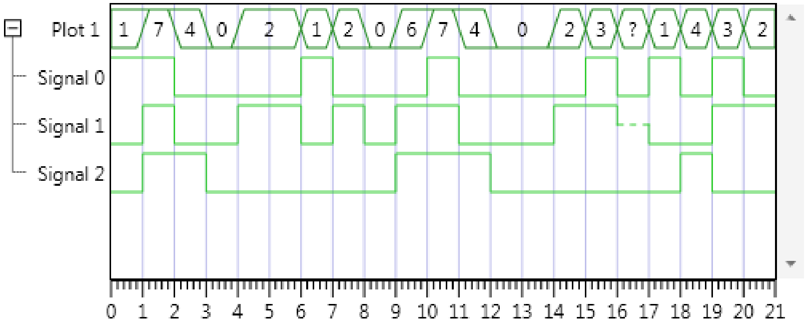
- Zoom interactively.
- Specify the angular axis scale in degrees, radians, or a custom symbol with the AngleValueFormatter class.

Tip For more information about using the polar graph control, refer to the *Measurement Studio WPF Graph Controls Overview* section in the *NI Measurement Studio Help*.

Digital Graph Control

Use this graph to plot and chart digital waveform data. You can use the digital graph to plot collections of Boolean, integer, and digital state data. Data values can represent up to eight different digital states. The tree view allows you to hide signals if you are showing both groups and signals simultaneously, and you can use properties to further configure the display.

Figure 3. Digital Graph Control



With the digital graph control and the classes that interface with the control, you can perform the following operations:

Plot Operations

- Plot digital waveform data. Data values can represent up to eight different digital states. Plot collections of Boolean, integer, and digital state data.
- Configure plot visuals using a `DefaultPlotRenderers` collection.
- Specify vector, raster, or hardware rendering for display, or set the graph to automatically switch between vector, raster, and hardware as needed.
- Customize plot rendering.
- Expand and collapse signal plots interactively or programmatically.
- Keep track of how your signal is changing over time using the phosphor effect.

Waveform Sample and Signal State Operations

- Display groups and signals simultaneously or individually.
- Configure the appearance of sample and state labels.
- Create custom waveform sample and signal state labels.

Cursor Operations

- Use cursors, multi-plot cursors, and range cursors to identify key points, values, and ranges in plots and the plot area.
- Determine which plots or axes a cursor can interact with by configuring the relevant properties.
- Use cursor labels to display X and Y data coordinates, values, or ranges in a customized format that the cursor crosshair points to, and customize the appearance of the label.
- Interactively move the cursor by clicking and dragging the vertical or horizontal crosshair or the center of the cursor.
- Programmatically move the cursor to a specified index, data value, or screen position.
- Create cursors programmatically or at design time using the graph's Children collection editor.

Axis Operations

- Display labeled groups and signals in a tree view along the left side of the plot area for digital plots.
- Interactively collapse and expand rendered groups by interacting with the tree.

- Configure group and signal labels on the digital axis.
- Configure the horizontal axis to automatically scale to show data being plotted or charted.
- Use a logarithmic horizontal axis with configurable bases.
- Interactively change the range of an axis at run time.
- Display labels on the axis.
- Display vertical grid lines.
- Position the axis to display on one or both sides of the graph's plot area.
- Configure major, minor and custom divisions.

Additional Operations

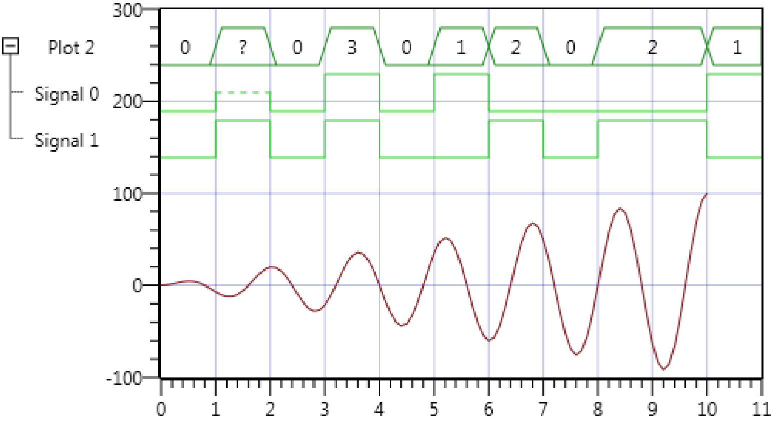
- Choose graph interaction at run time with the Graph Interaction Palette.
- Display data in sample or time format.
- Perform hit testing of mouse cursor coordinates.
- Pan with scroll bars.
- Configure the style and mode of scroll bars.
- Pan and zoom interactively and programmatically.

Tip For more information about using the graph and writable graph controls, refer to the *Measurement Studio WPF Graph Controls Overview* section in the *NI Measurement Studio Help*.

Mixed Graph Control

Use this graph to display both analog and digital data on the same graph. The digital and horizontal axes are used to scale digital data, and the horizontal and vertical axes are used to scale analog data. The tree view allows you to hide signals if you are showing both groups and signals simultaneously, and you can use properties to further configure the display. You can configure a graph to include multiple axes or independent ranges to ensure that the plot data fits the graph plot area, and you can configure the appearance of specific digital groups and states.

Figure 4. Mixed Graph Control



With the mixed graph control and the classes that interface with the control, you can perform the following operations:

Plot Operations

- Plot and chart collections of various types of scalar numeric values, such as double-precision floating point values, analog waveforms, and complex waveforms.
- Plot collections of Boolean, integer, digital sample, and digital waveform data.
- Configure plot visuals using the `DefaultAnalogPlotRenderers` and `DefaultDigitalPlotRenderers`collections to automatically generate different colors and shapes for different plots.
- Configure a graph to contain multiple analog and digital plots to show separate but related data on the same graph.
- Draw bars or fills from a plot to positive infinity, negative infinity, or zero.
- Configure appearance of digital groups and states.
- Specify vector, raster, or hardware rendering for display, or set the graph to automatically switch between vector, raster, and hardware as needed.
- Keep track of how your signal is changing over time using the phosphor effect.

Axis Operations

- Display labeled groups and signals in a tree view along the left side of the plot area for digital plots.
- Interactively collapse and expand rendered groups by interacting with the tree view.
- Configure a graph to include multiple axes or independent ranges so that plot data fits the graph plot area.
- Configure the axis to automatically scale to show data being plotted or charted.
- Use logarithmic axes with configurable bases.
- Interactively change the range of an axis at run time.
- Display labels on the axis.
- Display grid lines.
- Position the axis to display on one or both sides of the graph's plot area.
- Configure major, minor, and custom divisions.

Cursor Operations

- Use cursors, multi-plot cursors, and range cursors to identify key points , values, and ranges in plots and the plot area.
- Determine which plots or axes a cursor can interact with by configuring the relevant properties.
- Use cursor labels to display X and Y data coordinates, values, or ranges in a customized format that the cursor crosshair points to, and customize the appearance of the label.
- Interactively move the cursor by clicking and dragging the vertical or horizontal crosshair or the center of the cursor.
- Programmatically move the cursor to a specified index, data value, or screen position.
- Create cursors programmatically or at design time using the graph's Children collection editor.

Annotation Operations

- Configure text labels, arrows, and shapes to annotate a point anywhere in the plot area of the graph.
- Configure range area, text labels, and arrows to annotate a range in the plot area of the graph.

Additional Operations

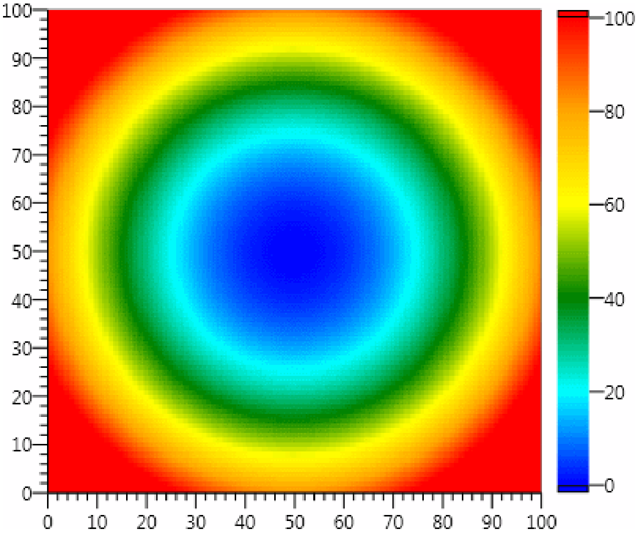
- Choose graph interaction at run time with the Graph Interaction Palette.
- Pan and zoom interactively, as well as programmatically.
- Perform hit testing of mouse cursor coordinates.
- Bind a graph to a data source.
- Pan with scroll bars.
- Configure the style and mode of scroll bars.

Tip For more information about using the mixed graph control, refer to the *Measurement Studio WPF Graph Controls Overview* section in the *NI Measurement Studio Help*.

Intensity Graph Control

The Intensity Graph is a control designed to plot 3D data on a 2D plot area using Color to represent the third dimension. The intensity graph accepts a 2D array of data, where each data value is rendered as a block of color on the graph. The mapping of data values to colors is configurable.

Figure 5. Intensity Graph Control



With the intensity graph control and the classes that interface with the control, you can perform the following operations:

Plot Operations

- Plot and chart data using the standard Data and DataSource properties available on all WPF graphs.

ColorScale Operations

- Configure the mapping from value to color for plotting.
- Configure major divisions.

Axis Operations

- Configure the axis modes to fixed, exact autoscaling, loose autoscaling, strip chart, or scope chart.
- Configure major, minor, and custom divisions.

Additional Operations

- Use Cursors to annotate points or regions in the plot area.
- Move cursor interactively.
- Interactively change the range of an axis or color scale at run time by clicking on the end labels.
- Pan with scroll bars.
- Configure the style and mode of scroll bars.

Tip For more information about using the intensity graph control, refer to the *Measurement Studio WPF Graph Controls Overview* section in the *NI Measurement Studio Help*.

Legend Control

Use the Measurement Studio legend control to display symbols and descriptions for a specific set of elements of another object, such as the plots or cursors of a graph. When you associate the legend control with another object, any changes you make to that object are automatically reflected in the legend. For example, if you associate the legend control with the plots of a graph, any changes you make in the plots collection editor are automatically reflected in the legend.

Tip For more information about using the legend control, refer to the *Measurement Studio WPF Graph Controls Overview* section in the *NI Measurement Studio Help*.

Numeric Controls

Use the Measurement Studio numeric controls to display numerical information with the look of scientific instruments. The numeric controls include a knob, gauge, meter, slide, tank, numeric textbox and radix numeric textbox. The following sections describe operations available with the controls and the classes that interface with them.

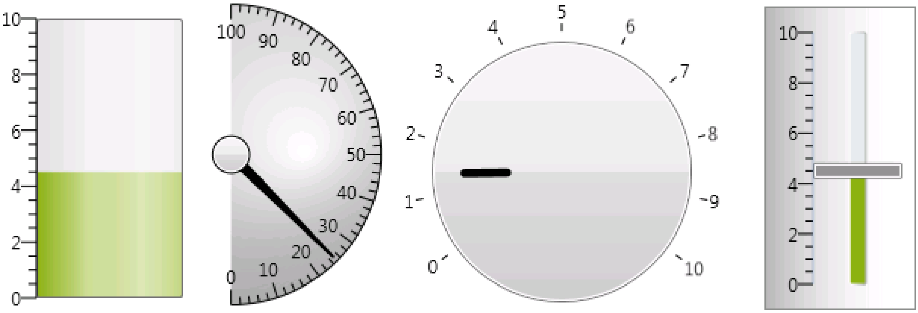
Note The default data type for the numeric controls is `Double` or `Int32`. To change the data type, right-click the control on the Visual Studio Designer, select Data Type, then select a different data type.

With all of the numeric controls and the classes that interface with them, you can perform the following operations:

- Natively handle a variety of data types, including `double`, `Int64`, and others.
- Capture any input value and coerce the captured value into range, while retaining the initial input value.
- Configure the scale to be linear or logarithmic, toggle the visibility of the scale, and customize the scale baseline width.
- Fill the scale and configure the color of the fill.
- Bind to other WPF controls so that if you change the value of one control, it changes the value of the other control.
- Customize the color, number of ticks, and length of ticks and labels.
- Configure the format of value labels.
- Interactively change the value of the control by clicking or dragging and moving the pointer with the mouse.
- Interactively change the range of an axis at run time by clicking on the axis end labels.
- For radial numeric controls, specify the start and sweep angle of the arc programmatically or from the Properties window.
- Configure major, minor, or custom divisions.

Use the Measurement Studio tank, gauge, knob, and slider controls to input and display numeric data on your user interface.

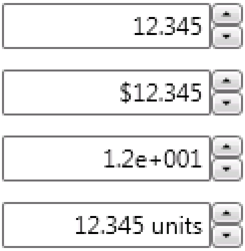
Figure 6. Tank, Gauge, Knob, and Slider Controls



Numeric TextBox and Radix Numeric TextBox Controls

Use the Measurement Studio numeric textbox and radix numeric textbox controls to display numeric values and to provide a way by which end users can edit numeric values.

Figure 7. Numeric TextBox Control with Different Value Formats



12.345

$12.345

1.2e+001

12.345 units

With the numeric textbox and radix numeric textbox controls and the classes that interface with the control you can perform the following operations:

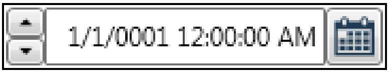
- Use up and down buttons for easy incrementing and decrementing.
- Use standard .NET and custom numeric formats for all displayed numeric values.
- Bind to other WPF controls so that if you change the value of one control, it changes the value of the other control.
- Set the `NumericTextboxInteractionModes`.

Tip For more information about using the Windows Forms numeric edit control, refer to the *Measurement Studio WPF Numeric Controls Overview* section in the *NI Measurement Studio Help*.

TimeStamp TextBox Control

You can use this numeric control to display and manipulate `DateTime` and `PrecisionDateTime` values as text and also enable use of a calendar to select dates. Use the up and down buttons to increment and decrement timestamp values, and use standard .NET and custom numeric formats for all displayed timestamp values. Use `PrecisionDateTime` to represent a point in time with sub-attosecond accuracy.

Figure 8. TimeStamp TextBox Control



1/1/0001 12:00:00 AM

With the timestamp textbox numeric control and the classes that interface with the control you can perform the following operations:

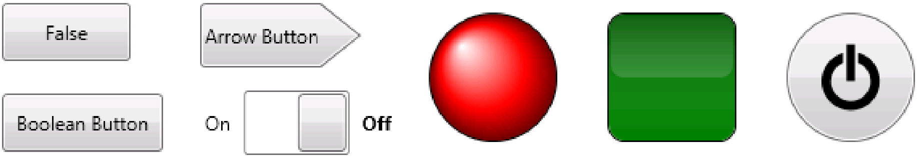
- Use up and down buttons to increment and decrement timestamp values.
- Use standard .NET and custom numeric formats for all displayed timestamp values.
- Bind to other WPF controls so that if you change the value of one control, it changes the value of the other control.
- Set the `NumericTextboxInteractionModes`.
- Use `PrecisionDateTime` to represent a point in time with sub-attosecond accuracy.

Tip For more information about using the timestamp textbox control, refer to the *Measurement Studio WPF TimeStamp TextBox Overview* section in the *NI Measurement Studio Help*.

Boolean Controls

Use the Measurement Studio arrow button, Boolean button, Boolean content button, power button, switch and LED controls as Boolean controls on a WPF user interface.

Figure 9. Boolean Controls



With the Boolean controls and the classes that interface with the controls, you can perform the following operations:

- Receive notification before or after the state of the control changes.
- Configure how the control behaves when you click the control with the mouse or press the spacebar when the control has focus.
- Configure the appearance of the control, including true and false colors.
- Specify true and false content for the Boolean content button, switch, and LED controls.

Tip For more information about using the Boolean controls, refer to the *Measurement Studio WPF Boolean Controls Overview* section in the *NI Measurement Studio Help*.

Windows Forms Controls

The Windows Forms .NET class library is in the `NationalInstruments.UI.WindowsForms` namespace. The Windows Forms class library encapsulates the following Measurement Studio user interface controls:

- Waveform graph
- Scatter graph
- Digital waveform graph
- Complex graph
- Intensity graph
- Legend
- Knob
- Gauge
- Meter
- Slide
- Thermometer
- Tank
- Numeric edit
- Switch
- LED

- Property editor
- Array controls

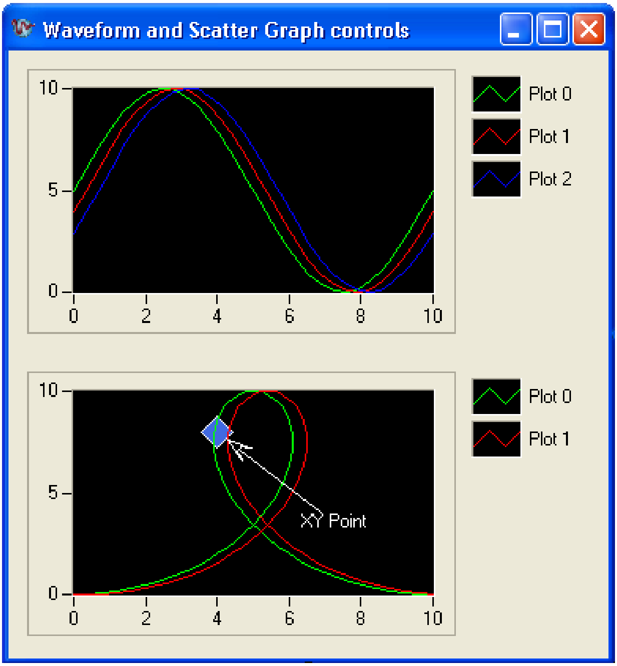
Use this class library to add measurement-specific user interface controls to your application. You can configure the controls programmatically at design time, through the Properties window in the Windows Forms Designer, or at run time with the property editor control. The following sections describe each of the Measurement Studio Windows Forms user interface controls.

Tip For more information about using the .NET user interface controls, refer to the *Using the Measurement Studio Windows Forms .NET Controls* section in the *NI Measurement Studio Help*.

Waveform Graph and Scatter Graph Controls

Use the Measurement Studio waveform graph and scatter graph controls to display two-dimensional data on a Windows Forms user interface. Use the waveform graph to display two-dimensional linear data. You explicitly specify each value in one dimension and provide an initial value and interval to implicitly specify the values in the other dimension. Use the scatter graph to display two-dimensional linear or nonlinear data: you explicitly specify each value in both dimensions.

Figure 10. Waveform Graph Windows Forms Control with Cursors and Scatter Graph Windows Forms Control with XY Point Annotation; Both Graphs Have Corresponding Legends



With the waveform graph and scatter graph controls and the classes that interface with the controls, you can perform the following operations:

Plot Operations

- Plot and chart arrays of double-precision floating point values, analog waveforms, and complex waveforms.
- Configure a graph to contain multiple plots to show separate but related data on the same graph. You can configure a graph to automatically generate different colors for different plots.
- Draw lines or fills from a plot to an X value, Y value, or another plot.
- Specify plots in the scatter graph control as X and Y data. Specify plots in the waveform graph control as X or Y data and optionally with date and time scaling.
- Use the extensible plot and plot area drawing capabilities and events to customize the graph appearance.
- Use plot data tooltips to display X and Y coordinates when a user hovers the mouse over a data point.
- Create custom point and line styles for plots.
- Specify anti-aliased plots for plot lines.
- Calculate and display error bands.

Axis Operations

- Configure a graph to include multiple axes or independent ranges so that plot data fits the graph plot area.
- Configure the axis modes to: fixed; autoscaling, including autoscaling based on the visible data only; strip chart; or scope chart.
- Use logarithmic axes with configurable bases.
- Interactively change the range of an axis and invert the axis at run time by clicking on the axis end labels.
- Display origin lines.
- Display captions on the axis.
- Display grid lines.
- Position the axis to display on one or both sides of the graph’s plot area.
- Configure major, minor, and custom divisions and origin lines.
- Display axis labels in elapsed time format.

Cursor Operations

- Use cursors to identify key points in plots and the plot area.
- Configure cursor snap modes as fixed, floating, nearest point, or to plot.
- Use cursor labels to display X and Y data coordinates in a customized format that the cursor crosshair points to, and customize the text font and colors of the label.
- Create custom point and line styles for cursors.
- Interactively move the cursor by clicking and dragging the vertical or horizontal crosshair or the center of the cursor.

- Programmatically move the cursor to the previous position, to the next position, or to a specified coordinate.
- Create custom mouse cursors programmatically or at design time using the mouse cursor style editor.

Annotation Operations

- Configure text labels, arrows, and drawing shapes to annotate a point anywhere in the plot area of the graph.
- Configure range area, text labels, and arrows to annotate a range in the plot area of the graph.
- Show tooltips configured to display data or other custom text.

Additional Operations

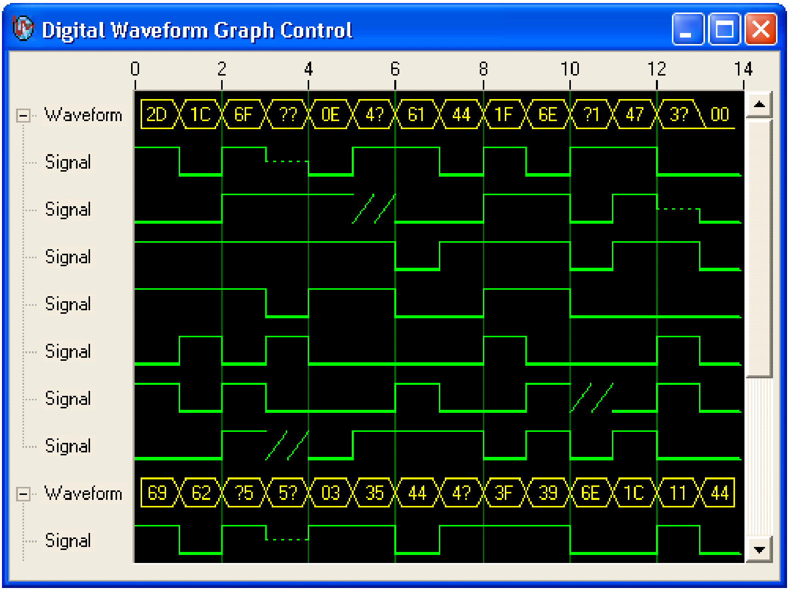
- Pan and zoom interactively, as well as programmatically.
- Copy the graph as a BMP, GIF, JPEG, or PNG image to the clipboard or a file.
- Perform hit testing of mouse cursor coordinates.
- Bind a plot to a data source on the waveform graph.
- Use the AutoFormat dialog box to configure axes.

Tip For more information about using the waveform and scatter graph controls, refer to the *Using the Measurement Studio Windows Forms Scatter and Waveform Graph .NET Controls* section in the *NI Measurement Studio Help*.

Digital Waveform Graph Control

Use the Measurement Studio digital waveform graph control to display `DigitalWaveform` data on a Windows Forms user interface.

Figure 11. Digital Graph Windows Forms Control


Digital Waveform Graph Control
0
2
4
6
8
10
12
14
Waveform
Signal
Signal
Signal
Signal
Signal
Signal
Signal
Waveform
Signal
2D 1C 6F ?? 0E 4? 61 44 1F 6E ?1 47 3? 00
69 62 ?5 5? 03 35 44 4? 3F 39 6E 1C 11 44

With the digital waveform graph control and the classes that interface with the control, you can perform the following operations:

Plot Operations

- Plot digital waveform data. Data values can represent up to eight different digital states.
- Configure plot labels on the y-axis.
- Configure plot templates to customize plots that are implicitly created from plotted data.
- Specify anti-aliased digital plots.
- Expand and collapse signal plots interactively or programmatically.
- Display tooltips.

Waveform Sample and Signal State Operations

- Simultaneously display waveforms and signals or display signals only.
- Create custom waveform sample and signal state styles.
- Configure the appearance of sample and state labels.
- Create custom waveform sample and signal state labels.

Axis Operations

- Configure the axis modes to fixed, exact autoscaling, or loose autoscaling.
- Interactively change the range of an axis and invert the axis at run time by clicking on the axis end labels.
- Display captions on the axis.
- Display grid lines.

- Position the axis to display on one or both sides of the plot area of the graph.
- Configure major, minor, and custom divisions.

Additional Operations

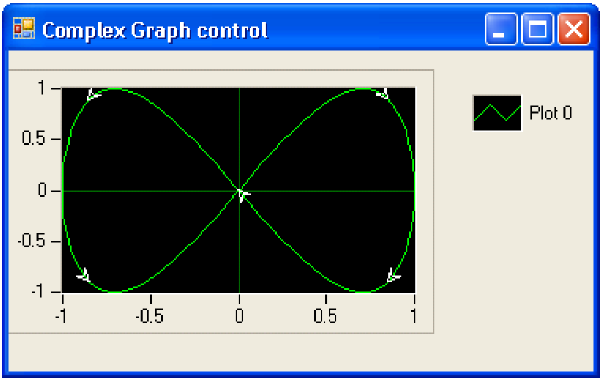
- Display data in sample or time mode.
- Perform hit testing of mouse cursor coordinates.
- Pan with scroll bars.
- Configure the style and mode of scroll bars.
- Create custom scroll bars.
- Pan and zoom interactively and programmatically.
- Copy the graph as a BMP, GIF, JPEG, or PNG image to the clipboard or a file.
- Create custom mouse cursors programmatically or at design time using the mouse cursor style editor.

Tip For more information about using the digital waveform graph control, refer to the *Using the Measurement Studio Windows Forms Digital Waveform Graph .NET Control* section in the *NI Measurement Studio Help*.

Complex Graph Control

Use the Measurement Studio complex graph control to display `ComplexDouble` data on a Windows Forms user interface. A `ComplexDouble` consists of a real part and an imaginary part. You can use a waveform graph to plot complex waveform data.

Figure 12. Complex Graph Windows Forms Control



With the complex graph control and the classes that interface with the control, you can perform the following operations:

Plot Operations

- Plot and chart `ComplexDouble` data.
- Configure a graph to contain multiple plots to show separate but related data on the same graph. You can configure a graph to automatically generate different colors for different plots.
- Draw lines or fills from a plot to an X value, Y value, or another plot.

- Use the extensible plot and plot area drawing capabilities and events to customize the graph appearance.
- Configure the plot to display arrows. The arrows indicate the direction of the complex data.
- Create custom point and line styles for plots.
- Specify anti-aliased plots for plot lines.
- Calculate and display error bands
- Display tooltips

Axis Operations

- Configure a graph to include multiple axes or independent ranges so that plot data fits the graph plot area.
- Configure the axis modes to: fixed; autoscaling, including autoscaling based on the visible data only; strip chart; or scope chart.
- Interactively change the range of an axis and invert the axis at run time by clicking on the axis end labels.
- Display origin lines and grid lines.
- Configure major, minor, and custom divisions and origin lines.
- Position the axis to display on one or both sides of the plot area of the graph.
- Display captions on the axis.

Cursor Operations

- Use cursors to identify key points in plots and the plot area.
- Configure cursor snap modes as fixed, floating, nearest point, or to plot.
- Use cursor labels to display X and Y data coordinates that the cursor crosshair points to, and customize the text font and colors of the label.
- Create custom point and line styles for cursors.
- Configure the graph to display cursors that are used to determine the real, imaginary, magnitude, and phase data coordinates of a point on the plot area.
- Create custom mouse cursors programmatically or at design time using the mouse cursor style editor.

Annotation Operations

- Configure text labels, arrows, and drawing shapes to annotate a point anywhere in the plot area of the graph.
- Configure range area, text labels, and arrows to annotate a range in the plot area of the graph.
- Annotate points and ranges of real, imaginary, and magnitude values.
- Annotate and label a range of magnitude values for a particular phase.

Additional Operations

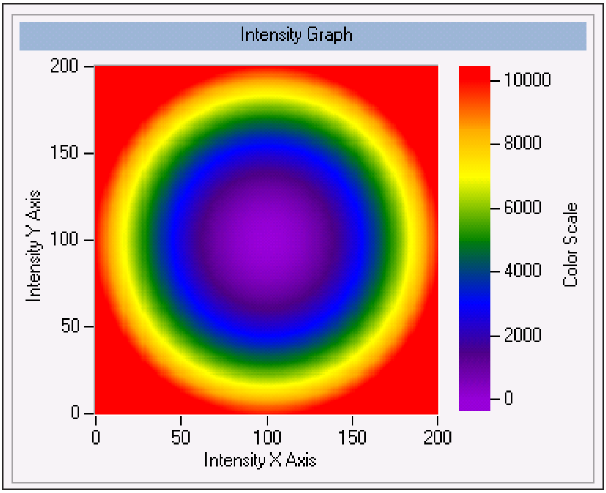
- Pan and zoom interactively.
- Copy the graph as a BMP, GIF, JPEG, or PNG image to the clipboard or a file.
- Use the AutoFormat dialog box to configure axes.

Tip For more information about using the complex graph control, refer to the *Using the Measurement Studio Windows Forms Complex Graph .NET Control* section in the *NI Measurement Studio Help*.

Intensity Graph Control

The Intensity Graph is a control designed to plot 3D data on a 2D plot area using Color to represent the third dimension. The intensity graph accepts a 2D array of data, where each data value is rendered as a block of color on the graph. The data value to color mapping can be specified by the user.

Figure 13. Intensity Graph Windows Forms Control



With the intensity graph control and the classes that interface with the control, you can perform the following operations:

Plot Operations

- Plot and chart data using Plot, PlotXAppend, and PlotYAppend methods.
- Interpolate data for every pixel from the input data to get better visual detail when the set of input data is sparse.
- Configure a graph to contain multiple plots to show separate but related data on the same graph.
- Use the extensible plot and plot area drawing capabilities and events to customize the graph appearance.

ColorScale Operations

- Configure the mapping from value to color for plotting.
- Configure a graph to include multiple color scales with independent color mappings for different plots.
- Configure major, minor, and custom divisions.

Axis Operations

- Configure a graph to include multiple axes with independent ranges to fit disparate data in a single graph plot area.
- Configure the axis modes to fixed, exact autoscaling, loose autoscaling, strip chart, or scope chart.
- Configure major, minor, custom divisions, and origin lines.
- Display axis labels in elapsed time format.

Additional Operations

- Use Cursors and Annotations to annotate points or regions in the plot area.
- Pan, zoom, move cursor, and move annotation caption interactively.
- Interactively change the range of an axis or color scale at run time by clicking on the end labels.
- Copy the graph as a BMP, GIF, JPEG, or PNG image to the clipboard or a file.
- Derive from and extend base classes to allow for easy programmatic customizability.
- Use the AutoFormat dialog box to configure axes.

Tip For more information about using the intensity graph control, refer to the *Using the Measurement Studio Windows Forms Intensity Graph .NET Control* section in the *NI Measurement Studio Help*.

Legend Control

Use the Measurement Studio legend control to display symbols and descriptions for a specific set of elements of another object, such as the plots or cursors of a graph. When you associate the legend control with another object, any changes you make to that object are automatically reflected in the legend. For example, if you associate the legend control with the plots of a graph, any changes you make in the plots collection editor are automatically reflected in the legend.

Tip For more information about using the legend control, refer to the *Using the Measurement Studio Windows Forms Legend .NET Control* section in the *NI Measurement Studio Help*.

Numeric Controls

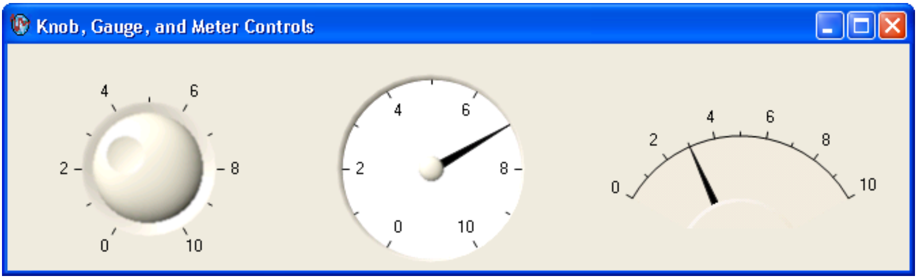
Use the Measurement Studio numeric controls to display numerical information, on a Windows Forms user interface, with the look of scientific instruments. The numeric controls include a knob, gauge, meter, slide, thermometer, and tank. The following sections describe operations available with the controls and the classes that interface with them.

With all of the numeric controls and the classes that interface with them, you can perform the following operations:

- Configure the scale to be linear or logarithmic, toggle the visibility of the scale, and customize the scale baseline width.
- Fill the scale and configure the range, color, dimensions, and style of the fill.
- Connect to the Measurement Studio .NET numeric edit control so that if you change the value of one control, it changes the value of the other control.
- Customize the appearance of the control using 3D lab styles or classic 2D styles and change the color and length of ticks and labels.
- Configure the format of value labels to engineering, date/time, or elapsed time format.
- Display tooltips reflecting the current value of the pointer.
- Interactively change the value of the control by clicking or dragging and moving the pointer with the mouse.
- Interactively change the range of an axis and invert the axis at run time by clicking on the axis end labels.
- Programmatically move the pointer to the previous or next value.
- Perform hit testing of mouse cursor coordinates.
- Specify the image format of the control as BMP, GIF, JPEG, or PNG.

Use the Measurement Studio knob, gauge, and meter controls to input and display numeric data on your user interface.

Figure 14. Knob, Gauge, and Meter Windows Forms Controls

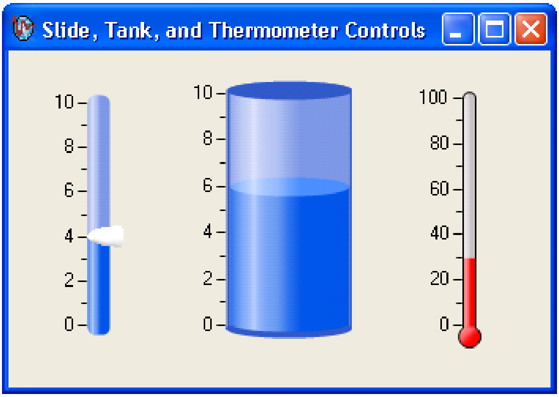


With the knob, gauge, and meter controls and the classes that interface with the controls, you can perform the following operations:

- Specify the start and sweep angle of the arc programmatically or from the Properties window.
- Use automatic division spacing, custom divisions, and invert the scale.

Use the Measurement Studio slide, tank, and thermometer controls to input and display numeric data on your interface.

Figure 15. Slide, Tank, and Thermometer Controls



With the slide, tank, and thermometer controls and the classes that interface with them, you can perform the following operations:

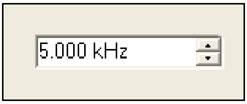
- Fill to the minimum or maximum value of the scale.
- Position the scale horizontally with left, right, or both and position the scale vertically with top, bottom, or both.

Tip For more information about using the Windows Forms numeric controls, refer to the *Knob, Gauge, Meter, Slide, Tank, or Thermometer Class* sections in the *NI Measurement Studio Help*.

Numeric Edit Control

Use the Measurement Studio numeric edit control to display numeric values and to provide a way by which end users can edit numeric values. Typically, you use a numeric edit control to input or display double numerical data instead of using a Windows Forms TextBox or NumericUpDown control.

Figure 16. Numeric Edit Windows Forms Control



With the numeric edit control and the classes that interface with the control you can perform the following operations:

- Use up and down buttons for easy incrementing and decrementing.
- Perform range checking.
- Set the minimum range value to negative infinity and the maximum range value to positive infinity.
- Create custom formats or use built-in numeric formats including generic, engineering, and simple double. You can use these numeric formats with other Measurement Studio user interface controls, such as the waveform graph and numeric pointer controls.

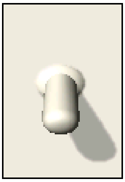
- Connect to a Measurement Studio numeric control so that if you change the value of one control, it changes the value of the other control.
- Set the coercion mode property to discrete or continuous values. This property configures the control to allow entry or display of either a discrete set of values or any value.
- Set the interaction mode to keyboard and mouse, keyboard only, mouse only, or none.

Tip For more information about using the Windows Forms numeric edit control, refer to the *NumericEdit Class* section in the *NI Measurement Studio Help*.

Switch and LED Controls

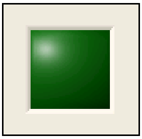
Use the Measurement Studio switch and LED controls as Boolean controls on a Windows Forms user interface. You typically use a switch control to receive and control Boolean input on an application user interface.

Figure 17. Switch Windows Forms Control in Vertical Toggle 3D Style



You typically use an LED control to indicate a Boolean value on an application user interface.

Figure 18. LED Windows Forms Control in Square 3D style



With the switch and LED controls and the classes that interface with the controls, you can perform the following operations:

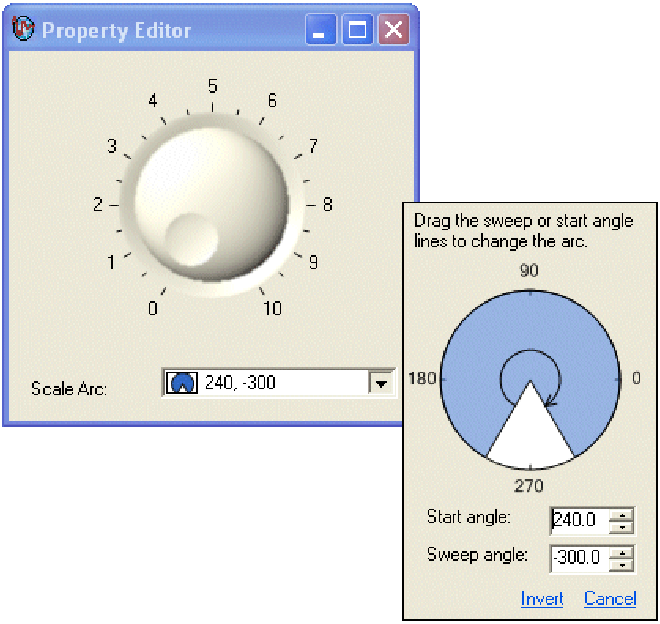
- Receive notification before or after the state of the control changes.
- Configure how the control behaves when you click the control with the mouse or press the spacebar when the control has focus.
- Configure the appearance of the control.
- Make the control background transparent.
- Configure the LED control to blink while it is on or off and configure the rate at which the LED control blinks.

Tip For more information about using the switch and LED controls, refer to the *Using the Measurement Studio Windows Forms Switch and LED .NET Controls* section in the *NI Measurement Studio Help*.

Property Editor Control

Use the Measurement Studio property editor control to configure properties for Windows Forms controls at run time.

Figure 19. Property Editor Windows Forms Control for the Knob Control Scale Arc Property



Scale Arc: 240, -300

Drag the sweep or start angle lines to change the arc.

Start angle: 240.0

Sweep angle: -300.0

With the property editor control and the classes that interface with the control, you can perform the following operations:

- Edit any .NET type at run time, including collections.
- Edit expandable properties that represent nested properties of another object, such as major divisions of an axis.
- Display custom editors and type converters for properties.
- Connect to a Windows Forms control so that if you change the value of a property of the control, the Property Editor will update to reflect the change.
- Configure the display mode as a visual representation of the value, text-only, or both.
- Set the interaction mode to edit values or indicator.

Tip For more information about using the property editor control, refer to the *Using the Measurement Studio Property Editor Control* topic in the *NI Measurement Studio Help*.

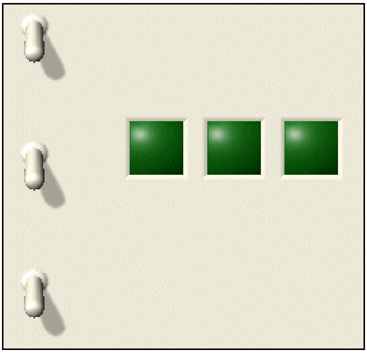
Windows Forms Array Controls

You can create an array of Measurement Studio controls that behave as a single unit. For example, you can use these array controls to visualize and control ports of a digital line or values of an array. Measurement Studio includes switch, LED, and numeric edit array controls. You can create control arrays of other controls if those controls meet the constraints of the generic type parameter `TControl`.

Switch and LED Array Controls

Use the Measurement Studio switch and LED array controls as an array of Boolean controls on a Windows Forms user interface. You typically use a switch array control to control ports of a digital line or values of an array. You typically use an LED array control to visualize ports of a digital line or values of an array.

Figure 20. Switch and LED Array Controls



With the switch and LED array controls and the classes that interface with the controls, you can perform the following operations:

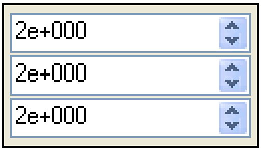
- Set values by passing an array of data.
- Modify the number of controls displayed based on the length of the specified values.
- Receive notification before or after the state of the control changes.
- Configure how the control behaves when you click it with the mouse or press the spacebar when the control has focus.
- Configure the appearance of the control.
- Make the control background transparent.
- Configure the LED controls to blink while they are on or off and configure the rate at which the LED controls blink.
- Configure the layout of the control to be horizontal or vertical.
- Bind the value of the control to a data source.
- Mark an array of Boolean controls so that only one can be true at a time.

Tip For more information about using the switch and LED array controls, refer to the *Using the Measurement Studio Control Array .NET Controls* topic in the *NI Measurement Studio Help*.

Numeric Edit Array Control

Use the Measurement Studio numeric edit array control to control and visualize values of an array of `double` values. With the numeric edit array control and the classes that interface with the control you can perform the following operations:

Figure 21. Numeric Edit Array Control



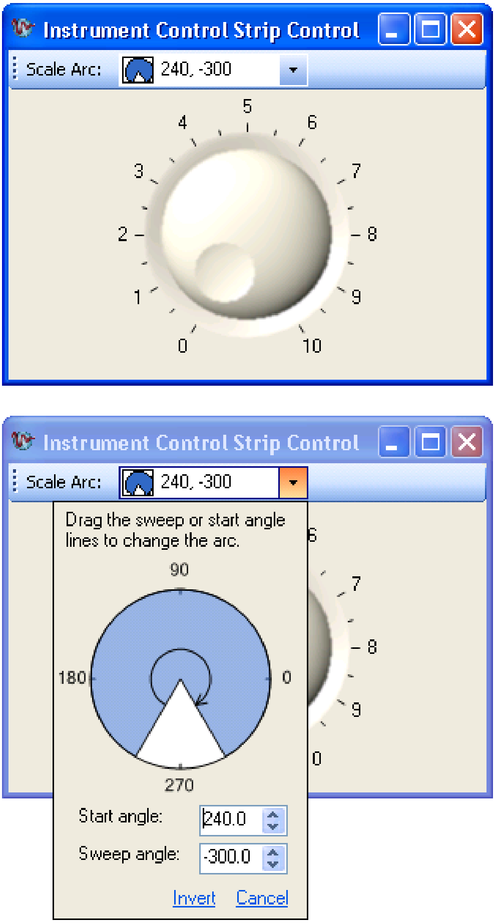
- Set values by passing an array of data.
- Modify the number of controls displayed based on the length of the array of values you specify.
- Use up and down buttons for easy incrementing and decrementing.
- Perform range checking.
- Set the minimum range value to negative infinity and the maximum range value to positive infinity.
- Create custom formats or use built-in numeric formats including generic, engineering, and simple double.
- Connect to a numeric control so that if you change the value of one control, it changes the value of the other control.
- Set the coercion mode property to discrete or continuous values. This property configures the control to allow entry or display of either a discrete set of values or any value.
- Set the interaction mode to keyboard and mouse, keyboard only, mouse only, or none.
- Use the edit box to select text programmatically and to validate text values.
- Configure the layout of the control to be horizontal or vertical.
- Bind the value of the control to a data source.

Tip For more information about using the numeric edit array control, refer to the *Using the Measurement Studio Control Array .NET Controls* topic in the *NI Measurement Studio Help*.

InstrumentControlStrip Control

You can use the InstrumentControlStrip control as a toolbar for editing property values of another control through the associated editors at run time. For example, you can populate the InstrumentControlStrip with ToolStripPropertyEditor items that edit property values of a waveform graph through the associated editors at run time. The editor displayed by the ToolStripPropertyEditor is the same editor that displays when you edit the property at design time.

Figure 22. InstrumentControlStrip Control



Scale Arc: 240, -300

Drag the sweep or start angle lines to change the arc.

Start angle: 240.0

Sweep angle: -300.0

Tip For more information about the InstrumentControlStrip control, refer to *Using the Measurement Studio Windows Forms Instrument Control Strip .NET Control* topic in the *NI Measurement Studio Help*.

ASP.NET Web Forms Controls

The Measurement Studio ASP.NET user interface controls are in the Web Forms .NET class library. The Web Forms .NET class library is in the `NationalInstruments.UI.WebForms` namespace. The Web Forms class library encapsulates the following Measurement Studio user interface controls:

- Waveform graph
- Scatter graph
- Digital waveform graph

- Complex graph
- Legend
- Knob
- Gauge
- Meter
- Slide
- Thermometer
- Tank
- Numeric edit
- Switch
- LED
- AutoRefresh

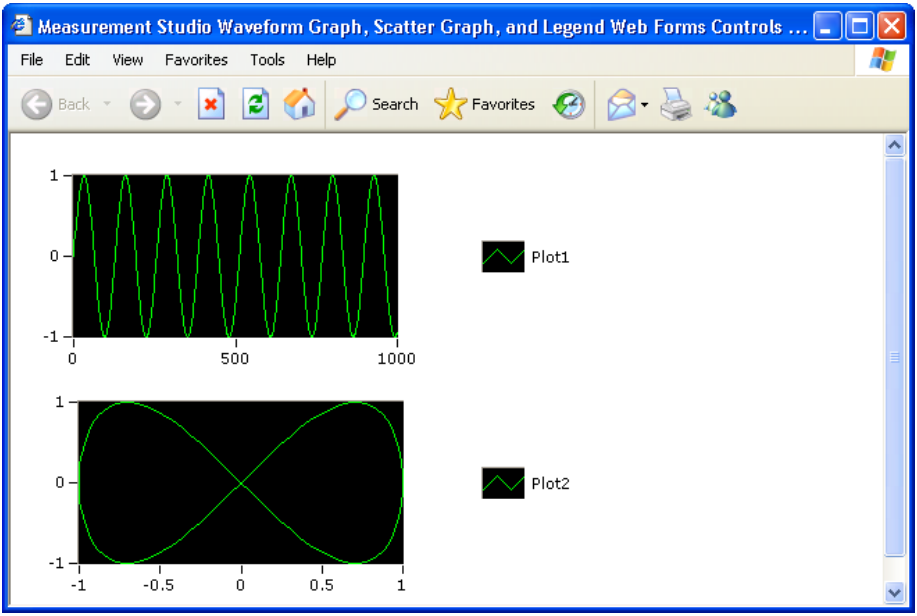
Note Measurement Studio Web Forms controls require complex UI functionality such as callbacks, view state management, and event handling to function properly. ASP.NET Model-View-Controller (MVC) does not include this functionality; therefore, Measurement Studio Web Forms controls do not support MVC.

Use this class library to add measurement-specific user interface controls to your Web application. You can configure the controls programmatically at design time or through the Properties window in the Web Forms Designer. The following sections describe each of the Measurement Studio Web Forms user interface controls.

Waveform Graph and Scatter Graph Controls

Use the Measurement Studio waveform graph and scatter graph controls to display two-dimensional data on a Web-based user interface. Use the waveform graph to display two-dimensional linear data. You explicitly specify each value in one dimension and provide an initial value and interval to implicitly specify the values in the other dimension. Use the scatter graph to display two-dimensional linear or nonlinear data: you explicitly specify each value in both dimensions.

Figure 23. Waveform Graph and Scatter Graph Web Forms Controls; Both Graphs Have Corresponding Legends



With the waveform graph and scatter graph controls and the classes that interface with the controls, you can perform the following operations:

Plot Operations

- Plot and chart arrays of double-precision floating point values, analog waveforms, and complex waveforms.
- Configure a graph to contain multiple plots to show separate but related data on the same graph. You can configure a graph to automatically generate different colors for different plots.
- Draw lines or fills from a plot to an X value, Y value, or another plot.
- Specify plots in the scatter graph control as X and Y data. Specify plots in the waveform graph control as X or Y data and optionally with date and time scaling.
- Use the extensible plot and plot area drawing capabilities and events to customize the graph appearance.
- Create custom point and line styles for plots.
- Specify anti-aliased plots for plot lines.
- Calculate and display error bands.
- Configure the plot to specify how data is saved and restored across HTTP requests.

Axis Operations

- Configure a graph to include multiple axes or independent ranges so that plot data fits the graph plot area.
- Configure the axis modes to: fixed; autoscaling, including autoscaling based on the visible data only; strip chart; or scope chart.
- Use logarithmic axes with configurable bases.
- Interactively change the range of an axis and invert the axis at run time by clicking on the axis end labels.
- Configure major, minor, and custom divisions and origin lines.
- Display axis labels in elapsed time format.

Cursor Operations

- Use cursors to identify key points in plots and the plot area.
- Configure cursor snap modes to be floating, nearest point, or to plot.
- Use cursor labels to display X and Y data coordinates in a customized format that the cursor crosshair points to, and customize the text font and colors of the label.
- Create custom point and line styles for cursors.
- Programmatically move the cursor to the previous position, to the next position, or to a specified coordinate.
- Create custom mouse cursors programmatically or at design time using the mouse cursor style editor.

Annotation Operations

- Configure text labels, arrows, and drawing shapes to annotate a point anywhere in the plot area of the graph.
- Configure range area, text labels, and arrows to annotate a range in the plot area of the graph.

Additional Operations

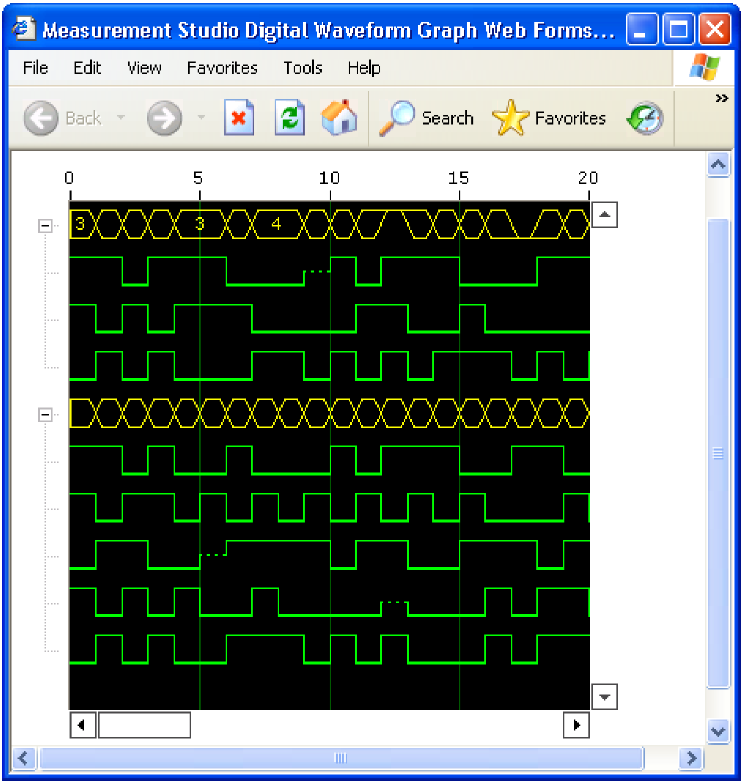
- Zoom interactively and programmatically.
- Specify the image format of the control as BMP, GIF, JPEG, or PNG.
- Use the AutoFormat dialog box to configure axes.

Tip For more information about using the waveform and scatter graph controls, refer to the *Using the Measurement Studio Web Forms Scatter and Waveform Graph .NET Controls* section in the *NI Measurement Studio Help*.

Digital Waveform Graph Control

Use the Measurement Studio digital waveform graph control to display `DigitalWaveform` data in an ASP.NET Web application.

Figure 24. Digital Waveform Graph Web Forms Control



With the digital waveform graph control and the classes that interface with the control, you can perform the following operations:

Plot Operations

- Plot digital waveform data, including digital signal state data and timing information.
- Configure plot labels on the y-axis.
- Configure plot templates to customize plots that are implicitly created from plotted data.
- Specify anti-aliased digital plots.
- Expand and collapse signal plots interactively as well as programmatically.

Waveform Sample and Signal State Operations

- Simultaneously display waveforms and signals or display signals only.
- Create custom waveform sample and signal state styles.
- Configure the appearance of sample and state labels.
- Create custom waveform sample and signal state labels.

Axis Operations

- Configure the axis modes to fixed, exact autoscaling, or loose autoscaling.
- Interactively change the range of an axis and invert the axis at run time by clicking on the axis end labels.
- Display captions on the axis and grid lines.
- Position the axis to display on one or both sides of the graph's plot area.
- Configure major, minor, and custom divisions.

Additional Operations

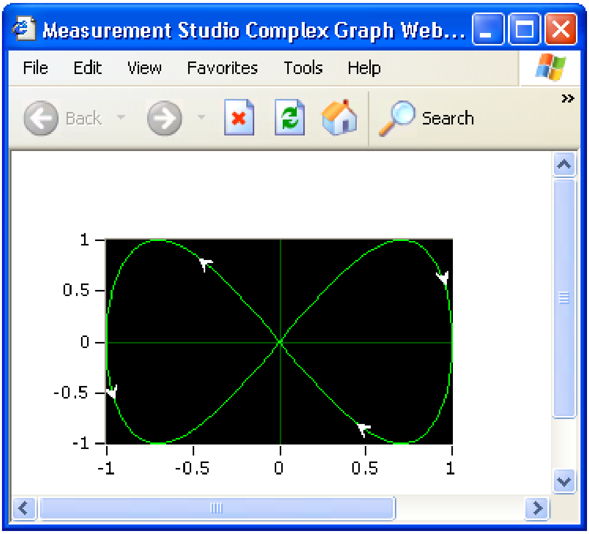
- Display data in sample or time mode.
- Configure the style and mode of scroll bars.
- Create custom scroll bars.
- Zoom interactively as well as programmatically.
- Specify the image format of the control as BMP, GIF, JPEG, or PNG.
- Create custom mouse cursors programmatically or at design time using the mouse cursor style editor.

Tip For more information about using the digital waveform graph control, refer to the *Using the Measurement Studio Web Forms Digital Waveform Graph .NET Control* section in the *NI Measurement Studio Help*.

Complex Graph Control

Use the Measurement Studio complex graph control to display `ComplexDouble` data on a ASP.NET Web application. A `ComplexDouble` consists of a real part and an imaginary part. You can use a waveform graph to plot complex waveform data.

Figure 25. Complex Graph Web Forms Control



With the complex graph control and the classes that interface with the control, you can perform the following operations:

Plot Operations

- Plot and chart `ComplexDouble` data.
- Configure a graph to contain multiple plots to show separate but related data on the same graph. You can configure a graph to automatically generate different colors for different plots.
- Draw lines or fills from a plot to an X value, Y value, or another plot.
- Use the extensible plot and plot area drawing capabilities and events to customize the graph appearance.
- Configure the plot to display arrows. The arrows indicate the direction of the complex data.
- Create custom point and line styles for plots.
- Specify anti-aliased plots for plot lines.
- Calculate and display error bands.
- Configure the plot to specify how data is saved and restored across HTTP requests.

Axis Operations

- Configure a graph to include multiple axes or independent ranges so that plot data fits the graph plot area.
- Configure the axis modes to: fixed; autoscaling, including autoscaling based on the visible data only; strip chart; or scope chart.
- Interactively change the range of an axis and invert the axis at run time by clicking on the axis end labels.
- Display origin lines, captions on the axis, and grid lines.
- Position the axis to display on one or both sides of the plot area of the graph.
- Configure major, minor, and custom divisions and origin lines.

Cursor Operations

- Use cursors to identify key points in plots and the plot area.
- Configure cursor snap modes to be floating, nearest point, or to plot.
- Use cursor labels to display real, imaginary, magnitude, or phase data that the cursor crosshair points to, and customize the text font and colors of the label.
- Create custom point and line styles for cursors.
- Programmatically move the cursor to the previous position, to the next position, or to a specified coordinate.
- Create custom mouse cursors programmatically or at design time using the mouse cursor style editor.

Annotation Operations

- Configure text labels, arrows, and drawing shapes to annotate a point anywhere in the plot area of the graph.
- Configure range area, text labels, and arrows to annotate a range in the plot area of the graph.

- Annotate and label a magnitude value.
- Annotate and label a range of magnitude values for a particular phase.

Additional Operations

- Zoom interactively as well as programmatically.
- Specify the image format of the control as BMP, GIF, JPEG, or PNG.
- Use the AutoFormat dialog box to configure axes.

Tip For more information about using the complex graph control, refer to the *Using the Measurement Studio Web Forms Complex Graph .NET Control* section in the *NI Measurement Studio Help*.

Legend Control

Use the Measurement Studio legend control to display symbols and descriptions for a specific set of elements of another object, such as the plots or cursors of a graph. When you associate the legend control with another object, any changes you make to that object are automatically reflected in the legend. For example, if you associate the legend control with the plots of a graph, any changes you make in the plots collection editor are automatically reflected in the legend.

Tip For more information about using the legend control, refer to the *Using the Measurement Studio Web Forms Legend .NET Control* section in the *NI Measurement Studio Help*.

Numeric Controls

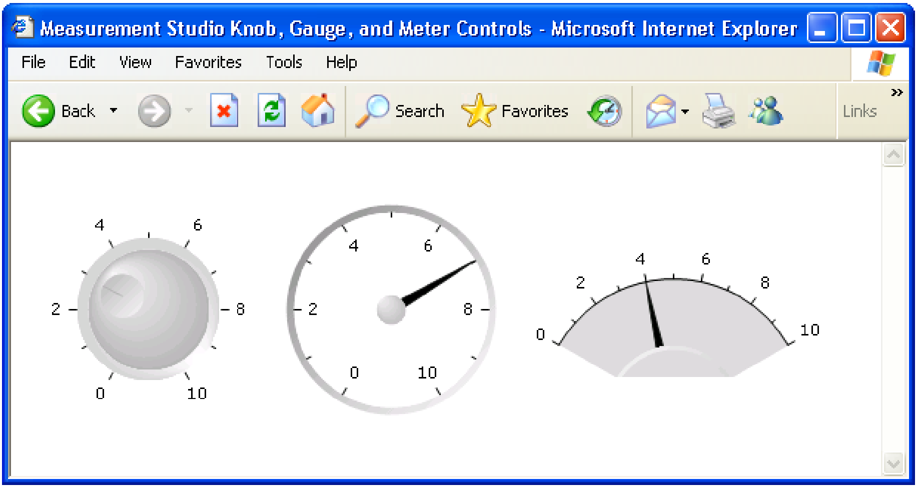
Use the Measurement Studio numeric controls to display numerical information in an ASP.NET Web application with the look of scientific instruments. The numeric controls include a knob, gauge, meter, slide, thermometer, and tank. The following sections describe operations available with the controls and the classes that interface with them.

With all of the numeric controls and the classes that interface with them, you can perform the following operations:

- Configure the scale to be linear or logarithmic and toggle the visibility of the scale.
- Fill the scale and configure the range, color, dimensions, and style of the fill.
- Connect to a Measurement Studio .NET numeric edit control so that if you change the value of one control, it changes the value of the other control.
- Customize the appearance of the control using 3D lab styles or classic 2D styles and change the color and length of ticks and labels.
- Configure the format of value labels to engineering or date/time.
- Specify the image format of the control as BMP, GIF, JPEG, or PNG.
- Interactively change the range of an axis and invert the axis at run time by clicking on the axis end labels.
- Display tooltips reflecting the current value of the pointer.
- Interactively change the value of the control by clicking the pointer with the mouse.
- Programmatically move the pointer to the previous or the next value.

Use the Measurement Studio knob, gauge, and meter controls to input and display numeric data on your user interface.

Figure 26. Knob, Gauge, and Meter Web Forms Controls

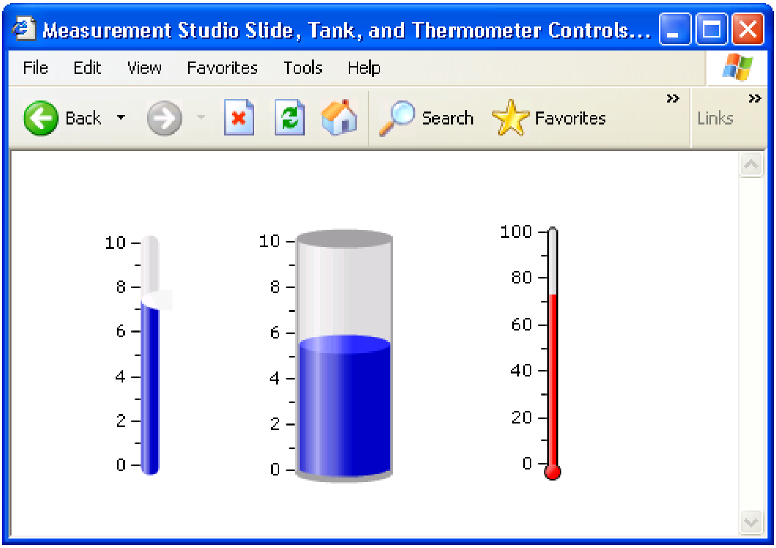

Measurement Studio Knob, Gauge, and Meter Controls - Microsoft Internet Explorer
File
Edit
View
Favorites
Tools
Help
Back
Search
Favorites
Links
4
6
2
8
0
10
4
6
2
8
0
10
0
2
4
6
8
10

With the knob, gauge, and meter controls and the classes that interface with the controls, you can perform the following operations:

- Specify the start and sweep angle of the arc programmatically or from the Properties window.
- Use automatic division spacing, custom divisions, and invert the scale.

Use the Measurement Studio slide, tank, and thermometer controls to input and display numeric data on your interface.

Figure 27. Slide, Tank, and Thermometer Web Forms Controls



With the slide, tank, and thermometer controls and the classes that interface with them, you can perform the following operations:

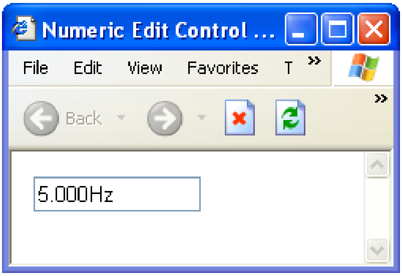
- Fill to the minimum or maximum value of the scale.
- Position the scale horizontally with left, right, or both and position the scale vertically with top, bottom, or both.

Tip For more information about using the Web Forms knob, gauge, meter, slide, tank, or thermometer controls, refer to the *Knob*, *Gauge*, *Meter*, *Slide*, *Tank*, or *Thermometer Class* sections in the *NI Measurement Studio Help*.

Numeric Edit Control

Use the Measurement Studio numeric edit control to display numeric values and to provide a way by which end users can edit numeric values. Typically, you use a numeric edit control to input or display double numerical data instead of using a Web Forms TextBox control.

Figure 28. Numeric Edit Web Forms Control



With the numeric edit control and the classes that interface with the control you can perform the following operations:

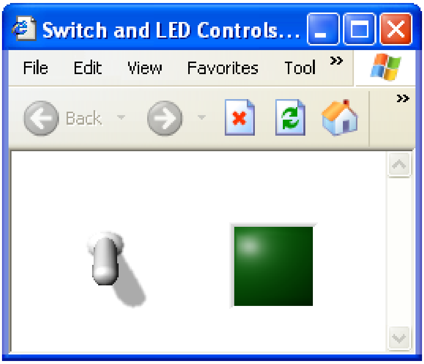
- Perform range checking.
- Set the minimum range value to negative infinity and the maximum range value to positive infinity.
- Create custom formats or use built-in numeric formats including generic, engineering, and simple double. You can use these numeric formats with other Measurement Studio user interface controls, such as the waveform graph and numeric pointer controls.
- Connect to a Measurement Studio numeric control so that if you change the value of one control, it changes the value of the other control.
- Set the coercion mode property to discrete or continuous values. This property configures the control to allow entry or display of either a discrete set of values or any value.
- Validate and format data without posting back to the Web server.

Tip For more information about using the Web Forms numeric edit control, refer to the *NumericEdit Class* section in the *NI Measurement Studio Help*.

Switch and LED Controls

Use the Measurement Studio switch and LED controls as Boolean controls in an ASP.NET Web application. You typically use a switch control to receive and control Boolean input in an ASP.NET Web application. You typically use an LED control to indicate a Boolean value on an ASP.NET Web application.

Figure 29. Switch Web Forms Control in Vertical Toggle 3D Style and LED Web Forms Control in Square 3D Style



With the switch and LED controls and the classes that interface with the controls, you can perform the following operations:

- Receive notification before or after the state of the control changes.
- Specify the image format of the control as BMP, GIF, JPEG, or PNG.

- Configure the appearance of the control.
- Configure the LED control to blink while it is on or off and configure the rate at which the LED control blinks.

Tip For more information about using the switch and LED controls, refer to the *Using the Measurement Studio Web Forms Switch and LED .NET Controls* section in the *NI Measurement Studio Help*.

AutoRefresh Control

Use the AutoRefresh control to update a Web control or a group of Web controls on the client at a specified interval.

The AutoRefresh control uses the ASP.NET client callback architecture to update a control or a group of controls at a specified interval. The AutoRefresh control sets up a timer inside the browser using Javascript. When the timer elapses, the AutoRefresh updates the controls in the AutoRefresh group. For down-level browsers, the controls update when the page posts back to the server. If the client browser supports client callbacks, the client-side script rendered by the AutoRefresh control uses a client callback to update the associated controls on the client without posting the page back to the server.

Note The AutoRefresh control is designed to work with the ASP.NET AJAX UpdatePanel and Timer controls in Visual Studio 2010.

AutoRefresh Callback

This feature provides a mechanism for updating the `Enabled` and `Interval` properties for `AutoRefresh` from within the `Refresh` callback, allowing you to turn off the `AutoRefresh` or change the `Interval` during an asynchronous HTTP request without causing a postback.

Measurement Studio Integrated Tools and Features

When you use Measurement Studio in the Visual Studio environment, you have access to measurement and automation tools and features for Visual Basic .NET and Visual C#. These integrated tools and features are designed to help you quickly and easily build measurement and automation applications.

Refer to the Creating Projects with Measurement Studio Core section in the NI Measurement Studio Help for more information about the functionality of these tools and features.

Measurement Studio Menu

The Measurement Studio Menu provides an easy way to access the following National Instruments resources and tools:

- **Measurement Studio Evaluation Home Page**—Launches the Try NI Measurement Studio for Visual Studio site on *ni.com*. This menu item appears only if you do not have a Measurement Studio license already activated.
- **Parameter Assistant**—Launches the Measurement Studio Parameter Assistant. Use the Measurement Studio Parameter Assistant to discover and insert valid parameter values for various Measurement Studio class libraries, such as NI-DAQmx, NI-488.2, and NI-VISA methods. The Parameter Assistant is available only if you have Measurement Studio class libraries installed that are supported by the Parameter Assistant.
- **Add/Remove .NET Class Libraries Wizard**—Launches the Add/Remove .NET Class Libraries wizard, which you can use to add or remove Measurement Studio class libraries or assemblies in existing Visual Basic .NET or Visual C# projects. This menu option is available only when you have a Visual C# or Visual Basic .NET project open.
- **Refresh Project License File**—Updates the `licenses.licx` file in a Measurement Studio project to the currently referenced Measurement Studio assemblies. The Refresh Project process works by going through the `licenses.licx` file line by line for the active project and removing each Measurement Studio licensed type that matches the Measurement Studio `PublicKeyToken`. After all Measurement Studio licensed types are removed from the `licenses.licx` file, the current Measurement Studio licensed types that are referenced by the project are added to the `licenses.licx` file. This ensures all Measurement Studio licensed types used by the project are added to the `licenses.licx` file. This menu option is available only when you have a Visual C# or Visual Basic .NET project open.
- **Refresh Web.Config File**—Updates the `Web.config` file in a Measurement Studio ASP.NET project to the currently referenced Measurement Studio assemblies. This menu option is available only when you have a ASP.NET website project open.
- **Update Measurement Studio Project References**—Updates any outdated Measurement Studio references to the latest version installed on the system. This menu option is available only when you have a Visual C# or Visual Basic .NET Windows Forms project open.
- **NI Tools**—Includes links to the following NI tools:
  - **Measurement & Automation Explorer (MAX)**—Use MAX to configure NI hardware; add new channels, interfaces, and tasks; execute system diagnostics; and view devices and instruments connected to the system. The MAX menu option is available only if you have MAX installed.
  - **NI I/O Trace**—Use NI I/O Trace to monitor, record, and display National Instruments API calls made by instrument connectivity applications. Use NI I/O Trace to quickly locate and analyze any erroneous National Instruments API calls that an application makes and verify that the communication with an instrument is correct. The NI I/O Trace menu item is available only if you have NI I/O Trace installed.

  - **Distributed System Manager**—Use Distributed System Manager to edit, create, and monitor network or shared variables. For more information, refer to the **NI Distributed System Manager Help**. In Distributed System Manager, select **Help»System Manager Overview** to access the help.
  - **NI Measurement Studio Installer Builder**—Launches the Installer Builder application, which helps you to create an installer for your application that includes NI libraries and products. Installer Builder ensures that National Instruments products and dependencies are included in your installer for proper run time behavior. For information on creating an installer using Installer Builder, refer to the *Measurement Studio Installer Builder Help*, accessible from within the Installer Builder application.
- **Measurement Studio Examples**—Launches the Measurement Studio example directory. You can use these example programs to help you learn and understand key concepts and to explore the functionality of Measurement Studio.
- **NI Measurement Studio Help**—Use the *NI Measurement Studio Help* to access detailed Measurement Studio help, including function reference, walkthroughs, and conceptual topic documentation on developing with Measurement Studio.
- **Measurement Studio Getting Started Guide**—Launches the *Getting Started with Measurement Studio* manual.
- **Measurement Studio Online Resources**—Includes links to the following NI websites:
  - **Measurement Studio Home Page**—Launches the Measurement Studio website at *ni.com/mstudio*, which contains Measurement Studio news, support, downloads, and evaluation software.
  - **Discussion Forums**—Launches the NI Discussion Forums website at *forums.ni.com*. With NI Discussion Forums, you can participate in discussion forums and exchange code with measurement and automation developers around the world.
  - **Technical Support**—Launches the NI Technical Support website at *ni.com/support*. These support resources are available for most products at no cost to registered users and include software drivers and updates, a KnowledgeBase, product manuals, step-by-step troubleshooting wizards, hardware schematics and conformity documentation, example code, tutorials and application notes, instrument drivers, discussion forums, and a measurement glossary.
- **Check in NI VLM Licenses**—Checks in the Measurement Studio concurrent license that you have checked out from a VLM (Volume License Manager) server. If your license has already been checked in and you select this menu option, there is no change to your license status. This menu option is available only when you are using a VLM license.
- **Check for Updates**—Launches the NI Update Service, which checks for updates to National Instruments software you have installed.
- **Patents**—Provides information about National Instruments patents.
- **Licenses**—Provides information about National Instruments licenses.
- **About Measurement Studio**—Launches the NI Measurement Studio About box, which provides version information.

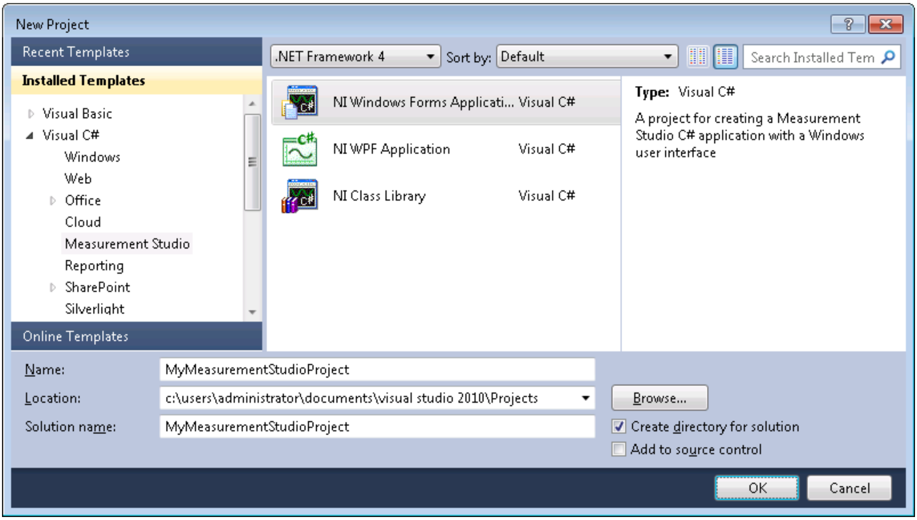
- **Activate Measurement Studio**—Launches the NI Activation Wizard, which guides you through activating Measurement Studio. This menu item appears only if you do not have a Measurement Studio license already activated.
- **Purchase Measurement Studio**—Launches the NI Measurement Studio Web page, where you can purchase a Measurement Studio license. This menu item appears only if you have not purchased a Measurement Studio license.

Creating a Measurement Studio Project

Measurement Studio includes class library and application templates that you can use to quickly create measurement applications with Visual Basic .NET and Visual C#. Use the Visual Studio New Project dialog box to access these templates and to create projects. You can create the following projects in Measurement Studio:

- Measurement Studio Visual Basic .NET project (WPF and Windows Forms)
- Measurement Studio Visual C# project (WPF and Windows Forms)
- Measurement Studio ASP.NET website

Figure 30. New Project Dialog Box in Visual Studio 2010



.NET Framework 4

Sort by: Default

Type: Visual C#

A project for creating a Measurement Studio C# application with a Windows user interface

Name: MyMeasurementStudioProject

Location: c:\users\administrator\documents\visual studio 2010\Projects

Solution name: MyMeasurementStudioProject

☑ Create directory for solution

□ Add to source control

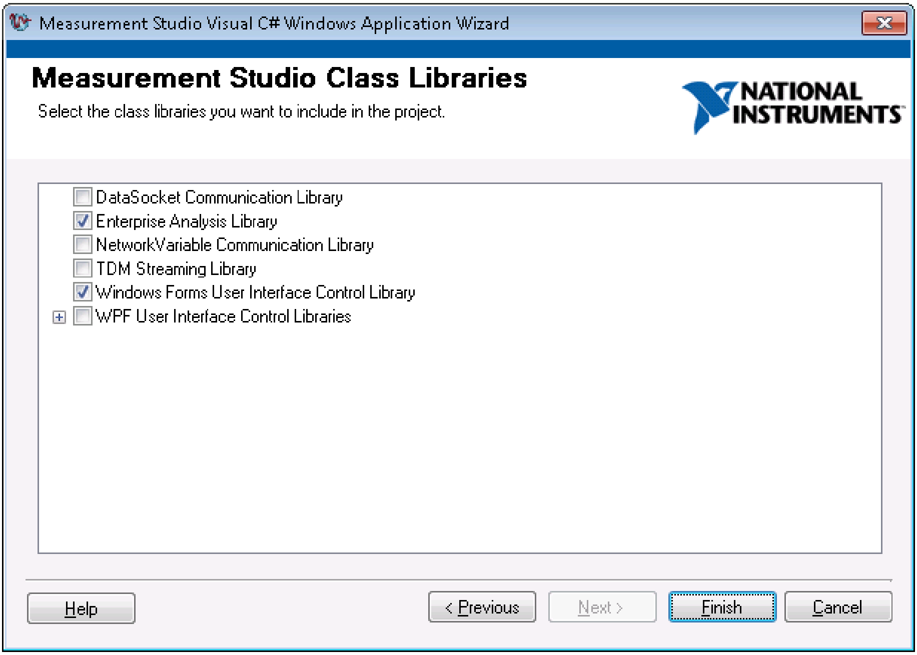
Tip For more information about using project templates to create a new Measurement Studio project, refer to the *Creating a New Measurement Studio Project* section in the *NI Measurement Studio Help*. For information about converting Measurement Studio projects, refer to the *Converting Measurement Studio Projects* section in the *NI Measurement Studio Help*.

Adding or Removing Measurement Studio .NET Class Libraries

To add or remove Measurement Studio .NET class libraries from a project, use the Measurement Studio Add/Remove .NET Class Libraries wizard on the Measurement Studio menu. This wizard provides an interface that you can use to select the Measurement Studio .NET class libraries you want to add to or remove from a project.

When you exit the wizard, the wizard adds or removes the appropriate references to or from the project, thus adding or removing the functionality associated with the class library.

Figure 31. Measurement Studio Add/Remove Class Libraries Wizard for Visual Studio 2010



Measurement Studio Visual C# Windows Application Wizard

Measurement Studio Class Libraries

Select the class libraries you want to include in the project.

- ☐ DataSocket Communication Library
- ☑ Enterprise Analysis Library
- ☐ NetworkVariable Communication Library
- ☐ TDM Streaming Library
- ☑ Windows Forms User Interface Control Library
- ⊞ ☐ WPF User Interface Control Libraries

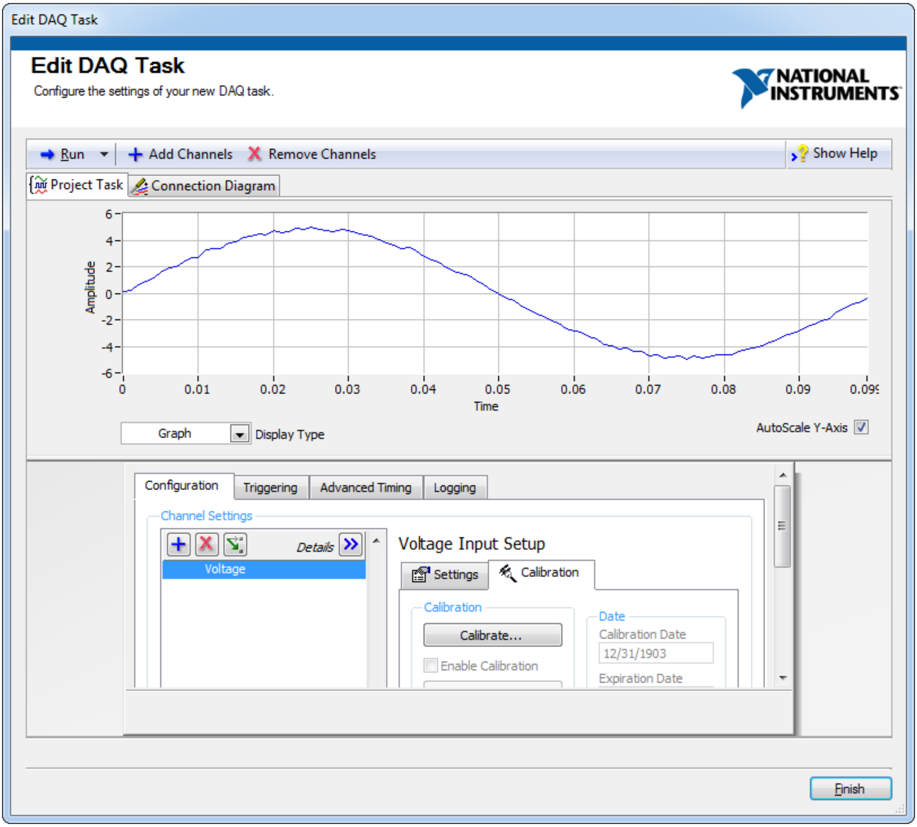
Tip For more information about using the Add/Remove .NET Class Libraries wizard to add or remove Measurement Studio .NET class libraries, refer to the *Adding or Removing Measurement Studio .NET Class Libraries* section in the *NI Measurement Studio Help*.

Creating a Measurement Studio NI-DAQmx Application

To create a Measurement Studio NI-DAQmx application, use the NI DAQ Windows Application template, which launches the DAQ Assistant. The DAQ Assistant integrates into Visual Studio as a code designer. Use the DAQ Assistant user interface to interactively create and configure the NI-DAQmx task. The DAQ Assistant automatically generates a Visual Basic .NET or Visual C# class that includes the functionality you configure in the user interface.

Note The DAQ Assistant is available only if you have installed NI-DAQmx and either the Measurement Studio Professional or Measurement Studio Enterprise package.

Figure 32. DAQ Assistant in Visual Studio 2012



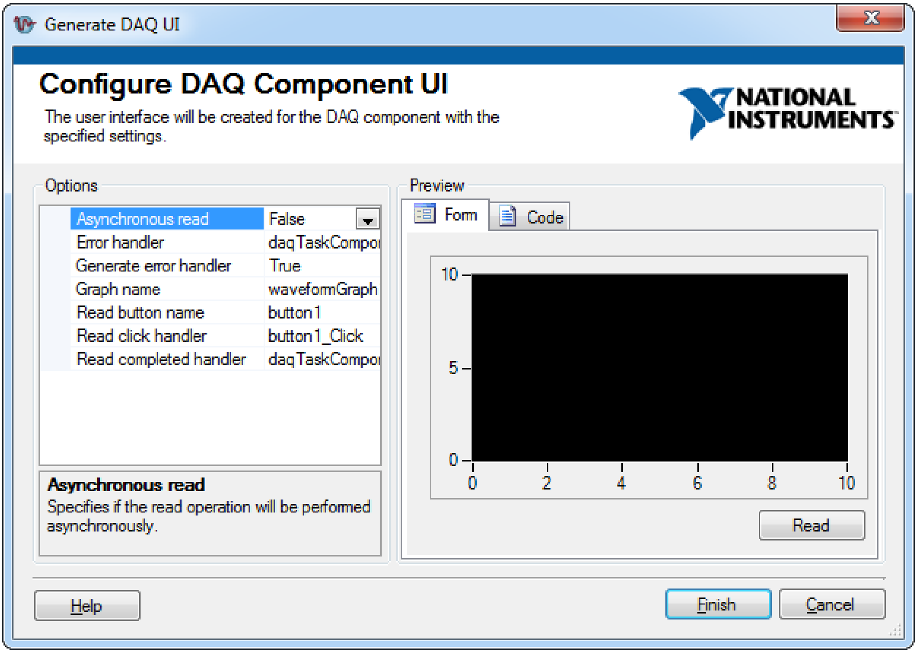
The DAQ Assistant interactively assists you in performing the following operations:

- Creating an NI-DAQmx task class
- Configuring an NI-DAQmx task class
- Generating a Visual Basic .NET or Visual C# class that includes the functionality you configure in the user interface
- Generating code that uses an NI-DAQmx task class
- Using an NI-DAQmx task class in a project
- Generating a DAQ component that uses the task to provide appropriate operations for your measurement type.

Creating an NI-DAQmx User Interface

Using the Configure DAQ Component UI wizard you can customize and preview a user interface and code for your task. The wizard also generates event handlers and code to acquire data and present it on your generated user interface.

Figure 33. Configure DAQ Component UI Wizard in Visual Studio 2012



Generate DAQ UI

Configure DAQ Component UI

The user interface will be created for the DAQ component with the specified settings.

Options

Asynchronous read	False
Error handler	daqTaskCompo
Generate error handler	True
Graph name	waveformGraph
Read button name	button1
Read click handler	button1_Click
Read completed handler	daqTaskCompo

Asynchronous read
Specifies if the read operation will be performed asynchronously.

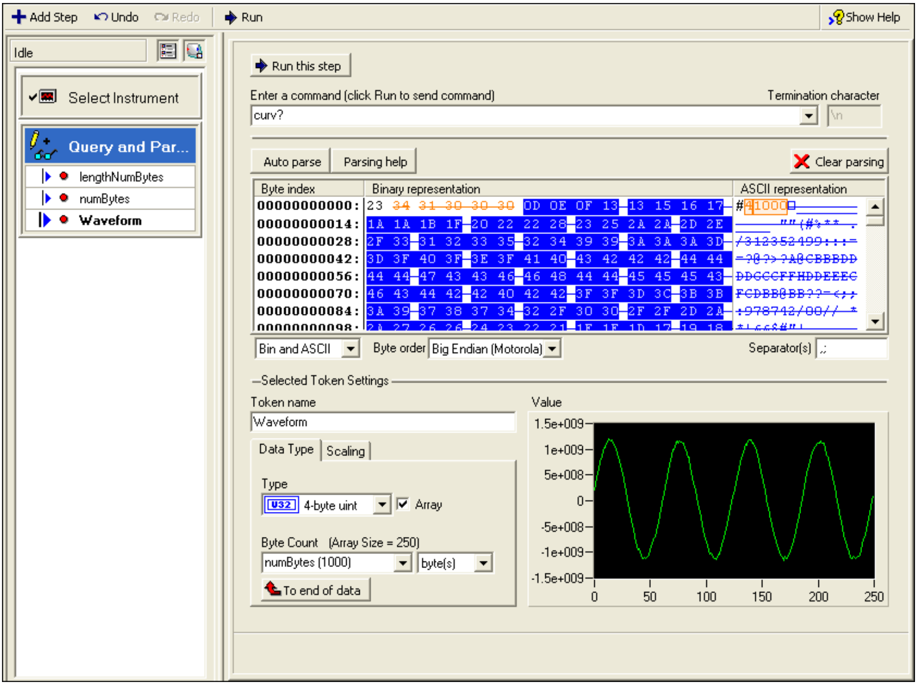
Tip For more information on how to create an NI-DAQmx user interface, refer to the *Using a .NET DAQ Component in a Project* topic in the *NI Measurement Studio Help*.

Creating an Instrument Control Application

To create a Measurement Studio instrument control application, use the NI VISA Windows Application template, which launches the Instrument I/O Assistant. The Instrument I/O Assistant integrates into Visual Studio as a code designer. Use the Instrument I/O Assistant user interface to create and configure the instrumentation task. The Instrument I/O Assistant generates a Visual Basic .NET or Visual C# class that includes the functionality you configure in the user interface. Use this assistant to help you write code that communicates with devices such as serial, Ethernet, or GPIB instruments.

Note The Instrument I/O Assistant is available only if you have installed either the Measurement Studio Professional or Measurement Studio Enterprise package, and if you are using Measurement Studio support for Visual Studio 2010.

Figure 34. Instrument I/O Assistant



Enter a command (click Run to send command): curv?

Termination character: \n

```
Byte index     Binary representation                          ASCII representation
00000000000:   23 34 31 30 30 30 0D 0E 0F 13 13 15 16 17     #41000
00000000014:   1A 1A 1B 1F 20 22 22 28 23 25 2A 2A 2D 2E     ""(#%**.
00000000028:   2F 33 31 32 33 35 32 34 39 39 3A 3A 3A 3D     /312352499:::=
00000000042:   3D 3F 40 3F 3E 3F 41 40 43 42 42 42 44 44     =?@?>?A@CBBBDD
00000000056:   44 44 47 43 43 46 46 48 44 44 45 45 45 43     DDGCCFFHDDEEEC
00000000070:   46 43 44 42 42 40 42 42 3F 3F 3D 3C 3B 3B     FCDBB@BB??=<;;
00000000084:   3A 39 37 38 37 34 32 2F 30 30 2F 2F 2D 2A     :978742/00//-*
00000000098:   2A 27 26 26 24 23 22 21 1F 1F 1D 17 19 18     *'&&$#"!
```

Bin and ASCII — Byte order: Big Endian (Motorola) — Separator(s): ,;

Selected Token Settings

Token name: Waveform

Type: U32 4-byte uint ☑ Array

Byte Count (Array Size = 250): numBytes (1000) byte(s)

The Instrument I/O Assistant aids you in performing the following operations:

- Creating an instrumentation task class.
- Configuring an instrumentation task class to communicate with an instrument and parse data you receive from the instrument.

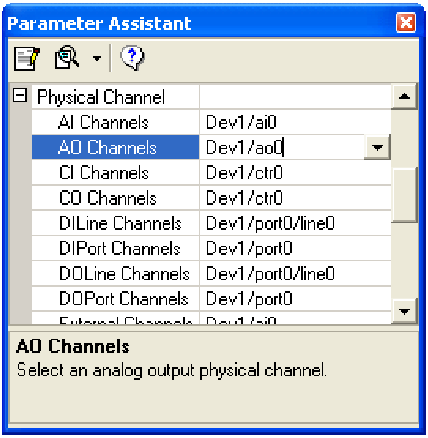
Tip For more information about using the Instrument I/O Assistant to create a Measurement Studio instrument control application, refer to the *Creating a Measurement Studio Instrument Control Application* section of the *NI Measurement Studio Help*.

Selecting a Measurement Studio Parameter Value

To access I/O devices or resources, you must specify string constants or scalar values for many method parameters and property values. Use the Measurement Studio Parameter Assistant, available from the Measurement Studio menu, to discover and insert into your code valid parameter values for methods and various Measurement Studio class libraries, such as NI-DAQmx, NI-488.2, and NI-VISA.

With the Parameter Assistant, you can select the correct parameter value for a device or resource based on your current system configuration. Click the **Insert Selected Item** button on the Parameter Assistant to insert the value into the current location in the active source file.

Figure 35. Measurement Studio Parameter Assistant



Physical Channel	
AI Channels	Dev1/ai0
AO Channels	Dev1/ao0
CI Channels	Dev1/ctr0
CO Channels	Dev1/ctr0
DILine Channels	Dev1/port0/line0
DIPort Channels	Dev1/port0
DOLine Channels	Dev1/port0/line0
DOPort Channels	Dev1/port0

AO Channels

Select an analog output physical channel.

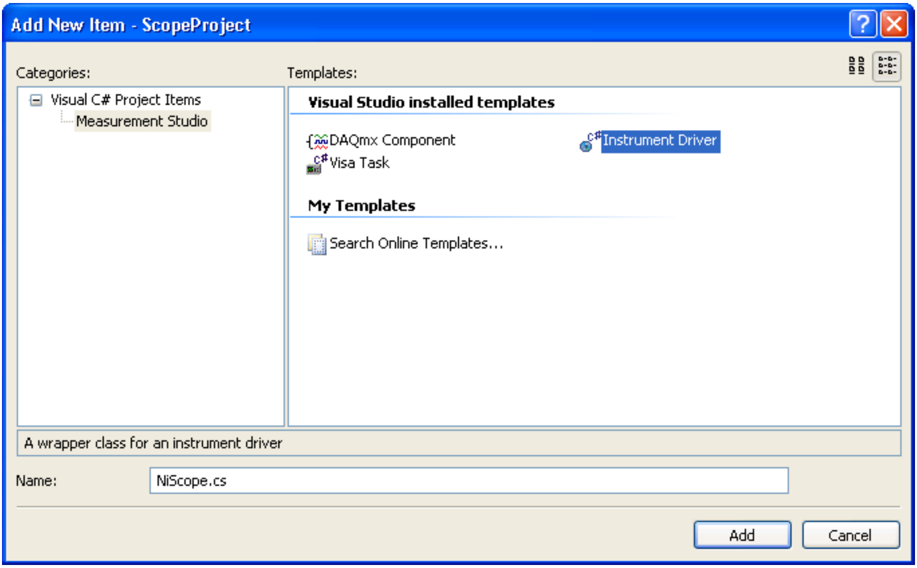
Tip For information about using the Measurement Studio Parameter Assistant to select a parameter value, refer to the *Selecting a Measurement Studio Parameter Value* section in the *NI Measurement Studio Help*.

Using the Instrument Driver Wizard

To use an IVI or VXI *plug&play* instrument driver with a C DLL in a Measurement Studio .NET application, use the Measurement Studio .NET Instrument Driver wizard to create .NET entry points to the C DLL functions you need to call from your application. Use the Add New Item wizard to select the .NET Instrument Driver Wizard.

The Measurement Studio .NET Instrument Driver wizard generates a .NET wrapper class for calling into IVI, VXI plug&play, and legacy instrument drivers based on the instrument driver function panel, header file, and an optional `.sub` file for IVI drivers. The wizard can generate both Visual C# and Visual Basic .NET source code. After completing the wizard, a new instrument driver wrapper class is added to your project and opened in the source code editor.

Figure 36. Launching the Measurement Studio .NET Instrument Driver Wizard from the Add New Item Wizard



A wrapper class for an instrument driver

Name: NiScope.cs

Tip For information about the .NET instrument driver wizard, refer to the *Calling Instrument Drivers from .NET Languages* topic in the *NI Measurement Studio Help*.

373225D-01 May15