SeeSharpTools

User Manual

版本 **1.5.0**

目录

1. SeeSharpTools 概要 4
   1.1 SeeSharpTools 介绍 4
   1.2 关于 SeeSharpTools 的 Q&A 4
2. SeeSharpTools 类库详述 6
   2.1 SeeSharpTools.JY.ArrayUtility 类库 6
   2.2 SeeSharpTools.JY.DSP.Fundamental 类库 8
   2.3 SeeSharpTools.JY.DSP.Measurement 类库 10
   2.4 SeeSharpTools.JY.GUI 类库 13
      2.4.1 EasyChart 13
      2.4.2 AquaGauge 15
      2.4.3 PressureGauge 15
      2.4.4 Thermometer 16
      2.4.5 KnobControl 17
      2.4.6 Slide 17
      2.4.7 Led 19
      2.4.8 EasyButton 20
      2.4.9 IndustrySwitch 20
      2.4.10 ButtonSwitch 21
      2.4.11 Tank 21
      2.4.12 ScrollingText 22
      2.4.13 SevenSegment 23
      2.4.14 SegmentBright 23
      2.4.15 LedMatrixControl 24
      2.4.16 SwitchArray、ButtonSwitchArray、LEDArray 25
      2.4.17 JYArray 25
      2.4.18 PathControl 26
      2.4.19 StripChart 26
      2.4.20 GaugeLinear 28
      2.4.21 EasyChartX 29
      2.4.22 ViewController 48
      2.4.23 StripChartX 49
   2.5 SeeSharpTools.JY.Database 类库 53
   2.6 SeeSharpTools.JY.Localization 类库 54

2.7 SeeSharpTools.JY.DSP.FilterMCR 类库 54
2.8 SeeSharpTools.JY.DSP.SoundVibration 类库 55
2.9 SeeSharpTools.JY.DSP.Utility 类库 55
2.10 SeeSharpTools.JY.Graph3D 类库 59
2.10.1 Surface Graph 类 59
2.10.2 Intensity Graph 类 60
2.11 SeeSharpTools.JY.File 类库 62
2.11.1 AnalogWaveformFile 类 62
2.11.2 BinHandler 类 64
2.11.3 CsvHandler 类 64
2.11.4 IniHandler 类 65
2.12 SeeSharpTools.JY.Report 类库 66
2.12.1 WordReport 类 66
2.12.2 ExcelReport 类 66
2.12.3 Logger 类 67
2.13 SeeSharpTools.JY.ThreadSafeQueue 类库 69
2.13.1 ThreadSafeQueue 类 69
2.13.2 CircularQueue 类 69
2.14 SeeSharpTools.JY.Sensors 类库 70
2.14.1 CustomScaling 类 70
2.14.2 DisplacementSensor 类 70
2.14.3 LoadCell 类 70
2.14.4 RTD 类 70
2.14.5 Thermistor 类 71
2.14.6 Thermocouple 类 71
2.15 SeeSharpTools.JY.Statistics 类库 72
2.16 SeeSharpTools.JY.TCP 类库 73
2.17 SeeSharpTools.JY.Mathemetics 类库 75
3. 常见问题解答 81

1. SeeSharpTools 概要

1.1 SeeSharpTools 介绍

SeeSharpTools 软件是简仪科技锐视测控平台提供的免费软件组件，包含一系列类库(Class Library)，提供方便易用的信号生成、分析和显示功能，帮助您在锐视测控平台下快速搭建测试测量解决方案。SeeSharpTools 软件包中所有库的描述信息如下：

类库名	功能概要
JY.ArrayUtility	提供常用数组运算和操作运算功能。
JY.Audio	提供音频测试时常用的波形生成功能和指标算法
JY.DSP.Fundamental	提供常用波形生成和频谱计算功能
JY.GUI	提供测试测量常用控件及相关功能
JY.Database	提供数据库常用功能
JY.Localization	提供对 WinForms 程序实现本地化功能
JY.DSP.FilterMCR	提供基于 Matlab 的滤波器计算相关功能
JY.DSP.SoundVibration	提供声音震动方面的算法功能
JY.DSP.Utility	提供常用的数据处理算法功能
JY.DSP.Utility.Fundmental	提供常用的基本数字信号处理算法
JY.Graph3D	提供 3D 图形功能
JY.File	提供文件读写功能
JY.Report	提供报表功能
JY.ThreadSafeQueue	提供一组线程安全的队列组件
JY.Sensors	提供一组传感器数据转换的组件
JY.Statistics	提供统计计算的算法工具
JY.TCP	提供一组通过网络 Socket 传输数据的功能接口

1.2 关于 SeeSharpTools 的 Q&A

Q：SeeSharpTools 软件是否提供开发环境中的在线帮助文档?

A：是的，在开发环境中调用 SeeSharpTools 各类库提供的方法时，在线帮助会自动显示(如下图所示)。本文档的后续内容也会介绍 SeeSharpTools 各类库的详细功能。

```
Generation.SineWave()
▲ 1 个(共 2 个) ▼ void Generation.SineWave(ref double[] x, [double amplitude = 1], [double phase = 0], [int numberOfCycles = 1])
Generates a sequence containing a sine wave with complete periods.

Chinese Simplified: 生成一个包含整数个周期的正弦波形，可设定正弦波的幅度、初始相位和周期数。
x: contains the sine wave sequence.

Chinese Simplified: 返回的正弦波形。
```

Q：SeeSharpTools 软件中是否包含更丰富的数字信号处理、文件读写等常用功能?

A：在锐视测控平台下简仪科技以例程源代码的方式提供更丰富的测试测量相关功能实现，详情请参看 SeeSharpExamples 例程集。另外，在锐视测控平台下，还有很多功能强大第三方的免费或开源类库，对于这些第三方类库的使用，简仪科技也是以例程和参考设计的方式提供的。

2. SeeSharpTools 类库详述

SeeSharpTools 类库中提供了一组测试测量常用组件，覆盖了界面设计控件、数组操作、信号处理、文件读写、数据库操作等功能。

2.1 SeeSharpTools.JY.ArrayUtility 类库

JY.ArrayUtility 包含 2 个静态类：ArrayCalculation 和 ArrayManipulation，分别提供常用数组运算和操作功能。这两个类提供的方法都是泛型方法，支持的数据类型包括：double、float、decimal、long、ulong、int、uint、short、ushort、char、byte。方法如下表所示：

ArrayCalculation 类	
方法名	**功能概要**
InitializeArray	使用指定元素值初始化数组
Add Subtract Multiply	两个等长一维数值数组之间逐个元素的加、减、乘运算。
AddOffset SubtractOffset MultiplyScale	一维或二维数值数组与单一数值的加、减、乘运算。
Zero	将一维或二维数值数组中所有元素的值置为零。
Sum Average RMS Abs	一维或二维数值数组的和、平均值、均方根值、绝对值的计算。
AreEqual	比较两个数值数组的元素是否完全相同。
ArrayMaxAndMin	寻找一维数组中的最大值和最小值，及其索引； 寻找二维数组中的最大值和最小值，及其行列索引。

ArrayManipulation 类	
方法名	**功能概要**
BuildArray	连接两个二维数组，超出维度部分的元素会被舍弃； 连接多个一维数组，超出维度部分的元素会被舍弃。
Insert_1D_Array	在一维数组的指定位置插入元素。

Insert_2D_Array	向二维数组的指定行或者指定列中插入一维数组。
Connect_1D_Array	将两个一维数组连接为两个数组长度之和的一维数组。
Connected_2D_Array	将两个一维数组或者一个一维数组与二维数组按行或者按列连接为二维数组。
Concatenate	两个或者多个数组(一维或二维)按行或按列连接为二维数组。
Convert2StringArray	将指定类型的数值数组 (一维或者二维)转换为字符串数组(一维或二维)，可以指定格式化的样式。
Convert2NumericArray	将字符串数组(一维或者二维)转换为指定类型的数值数组(一维或二维)。
GetArraySubset	将一维数组 a[]中从指定起始位置起的后续元素拷贝至一维数组 b[]； 将二维数组 a[,]的指定行或列拷贝至一维数组 b[]； 获取二维数组 a[,]的指定行或列，返回至一维数组中，行(或列)索引值由 index 给出，行或者列的数量由 count 给出，按行索引或按列索引由 indexType 给出；
ReplaceArraySubset	将一维数组 a[]的所有元素拷贝至一维数组 b[]中的指定位置，替换原有元素； 将一维数组 a[]的所有元素拷贝至二维数组 b[,]中的指定行或列，替换该行或列的元素； 将二维数组 src[,]的所有元素拷贝至二维数组 dst[]中的指定位置，替换原有元素，起始索引行/列的值由 rowIndex/columnIndex 给出。
InsertArraySubset	在原二维数组 src[,]的指定索引位置插入新的二维数组 insertArray[,]，插入行或者列的位置由 index 给出，插入行列方向有 indexType 给出。
SplitArray	在指定索引处分离一维数组； 在指定索引处分离二维数组。
Interleave_1D_Arrays	交织多个一维输入数组中的相应元素，形成输出数组。当输入数组的大小不同时，交织的数组的元素数等于最小输入数组的元素数乘以输入数组数。
Decimate_1D_Array	使一维数组的元素分成若干输出数组，依次输出元素，舍弃所有使输出数组长度不同的元素。
Rotate_1D_Array	使数组中的元素移动多个位置。
Search_1D_Array	在一维数组中从开始索引处开始搜索元素，返回第一个匹配的索引，如果未找到则返回-1。
Interpolate_1D_Array	在原始一维数组的索引处线性插入一个值。
ThresholdInterpolate_1D_Array	在一维数组中插入点，比较阈值与开始索引位置以后的值，直至找到一对连续的元素，阈值大于等于第一个元素，小于等于第二个元素。
Transpose	二维数组行列转置，即将 N 行 M 列的数组转换为 M 行 N 列。
Reverse	反转一维数组中的所有元素。源数组和目标数组可以是同一个数组。
Sort	对一维数值数组、二维数值数组的所有元素或单个方向(一行或者一列)进行排序，源数组和目标数组可以相同，可以指定排序

	的方向(升序或者降序)。
Remove	移除一维数组(可以指定移除元素的起始位置和长度)或者二维数组的一部分(可以指定移除行或者列以及移除的起始位置和长度)。

2.2 SeeSharpTools.JY.DSP.Fundamental 类库

JY.DSP.Fundamental 包含 4 个静态类：Generation，Generation.Analog，Generation.Digital 和 Spectrum，分别提供常用波形，模拟/数字波形生成和频谱计算功能，方法如下表所示：

方法名	功能概要
Generation 类	
SineWave	生成一个包含整数个周期的正弦波形，可设定正弦波的幅度、初始相位和周期数； 生成一个正弦波形，可设定正弦波的幅度、初始相位、频率和采样率。
SquareWave	生成一个包含整数个周期的方波波形，可设定方波的幅度、占空比和周期数； 生成一个方波波形，可设定方波的幅度、占空比、频率和采样率。
UniformWhiteNoise	生成一个指定幅度的随机白噪声波形。
Ramp	生成一个等差数列。
Generation.Analog 类	
GaussianModulatedSinePattern	生成含有经高斯调制的正弦波
GaussianMonopulse	生成含有高斯单脉冲的波形
SincPattern	生成 sinc 信号的波形
PeriodicSincPattern	生成周期 sinc 信号的波形
PulsePattern	生成脉冲信号的波形
ChirpPattern	生成 chirp 信号的波形
ImpulsePattern	生成冲激信号的波形
Generation.Digital 类	
Gradient	生成数字信号，信号的周期依次减半
Marching	生成数字信号，第一个采样的第一个信号的二进制值将按顺序移至波形后续采样子集的下一个信号
Periodic	生成多周期数字信号，每个周期包含由一段数字信号和空闲状态组成
Single	生成数字信号，所有比特数都设置为 0 或 1
Random	随机生成含 0 和 1 的数字信号，用于生成随机波形的数值产生

	没有任何数学规律
Switch	生成数字信号，值 1 输入用于产生偶数采样中的二进制值，值 2 输入用于产生奇数采样中的二进制值
Spectrum 类	
PowerSpectrum	计算输入时域信号的功率谱，可设定加窗类型和输出功率谱的单位；
AdvancedComplexFFT	计算时域信号的复数频谱数据。返回的结果保存在 Complex 类型的数组中，数组的长度为 N/2+1，其中 N 为输入数据的长度。
PeakSpectrumAnalysis	计算频谱峰值及峰值频率。
DBFullScaleSpectrum	计算输入信号相对于满量程的分贝值。

注意:在使用 Spectrum 类之前，需要安装 JXDSPRuntimeMKL，JXDSPRuntimeMKL 分为 X64 和 X86 两个版本，请根据 windows 系统来选择安装。

2.3 SeeSharpTools.JY.DSP.Measurements 类库

JY.DSP.Measurements 包含 5 个静态类: SquarewaveMeasurements JitterAnalysis. ClockRecovery, JitterAnalysis.Jitter, JitterAnalysis.Level and JitterAnalysis.Timing, 分别提供方波测试和抖动分析的方法，如下表所示：

方法名	功能概要
SquarewaveMeasurements 类	
AmplitudeLevelAnalysis	根据高低电平的设置，分析单个波形的幅度，高低电平等级； 根据高低电平的设置，分析多个波形的幅度，高低电平等级。
PeriodAnalysis	根据高低电平，极性和参考电平设置，分析单个波形的周期，占空比，脉宽； 根据高低电平，极性和参考电平设置，分析多个波形的周期，占空比，脉宽。
PulseMeasurement	根据高低电平，极性和参考电平设置，测量每个波形指定序号脉冲的周期，脉宽和占空比； 根据高低电平，极性和参考电平设置，测量波形指定序号脉冲的周期，脉宽和占空比。
CycleRmsMeanAnalysis	根据高低电平，极性和参考电平设置，测量每个波形所有周期的 RMS 和均值； 根据高低电平，极性和参考电平设置，测量每个波形指定周期的 RMS 和均值； 根据高低电平，极性和参考电平设置，测量波形所有周期的 RMS 和均值； 根据高低电平，极性和参考电平设置，测量波形指定周期的 RMS 和均值。
TransitionMeasurement	根据高低电平，极性和参考电平设置，测量每个波形指定边沿的斜率，上升、下降时间等信息； 根据高低电平，极性和参考电平设置，测量波形指定边沿的斜率，上升、下降时间等信息。
JitterAnalysis.ClockRecovery 类	
MeanClock	从输入信号中恢复参考时钟信号。
ConvertBitCountToBitstream	把信号中每个跃迁的比特数量信息转换为每个单位时间的比特流序列。
CreateReferenceBitWaveform	创建与输入信号同步的理想的参考时钟波形。

JitterAnalysis.Level	
MeasureStateLevel	测量输入信号的高低低电平和幅度。
MeasureReferenceLevel	根据状态电平计算参考电平； 根据输入信号计算参考电平。
FindTransitions	根据设定的参考电平测量信号中每个跃迁的边界和方向。
SelectEdge	根据跃迁方向选择对应的边沿。
OvershootUndershoot	测量过冲和下冲发生的位置和幅度。
MeasureMaxMinVoltage	测量信号中每个跃迁处的最大和最小电压。
MeasurePeakToPeakVoltage	测量信号中每个周期的峰峰值。
MeasureArea	测量信号每个周期的面积和平均值。
MeasureRMSVoltage	测量信号中每个周期的电压有效值。
JitterAnalysis.Timing	
FindLevelCrossings	测量指定电压穿越点的位置。
MeasureRiseTime	测量信号跃迁的上升时间。
MeasureFallTime	测量信号跃迁的下降时间。
MeasurePeriod	对跃迁分布均匀的信号，测量周期。
MeasureDutyCycle	测量信号的占空比。
MeasurePositiveWidth	测量上升跃迁（正脉宽或者高电平）的脉冲持续时间。
MeasureNegativeWidth	测量下降跃迁（负脉宽或者低电平）的脉冲持续时间。
MeasureBurstWidth	测量信号中突发信号的持续时间，突发信号起始于某个边沿穿过某阈值电平，结束于下一个边沿再次穿过此阈值电平，并持续超过最小指定时间。
MeasureSetupHoldTime	测量两个信号之间的延时（偏度）。
ChannelSkew	测量两个信号之间的延时（偏度）。
MeasurePhase	测量两个信号每个周期的相位差。
JitterAnalysis.Jitter	
MeasureTimeIntervalError	测量原始信号的电平穿越位置和对应的参考时钟信号的电平穿越位置的时间差
MeasurePeriodJitter	测量原始信号和对应参考时钟信号的每个周期的差值
MeasuereDutyCycleDistortion	测量信号的占空比失真，表示信号正负脉宽的平均差值
MeasureCycleToCycle	计算序列相邻元素的差值
MeasurePhaseNoise	测量抖动信号的相位噪声，也就是频域中相邻频率分量的 RMS 之和
SignalOperation	
AnalogToDigital	使模拟信号转换为数字信号。
DigitalToAnalog	将数字信号转换为模拟信号。
SignalOperation.Filtering	

MedianFilter	对输入序列进行中值滤波。
FIRFilter	使用 FIR 系数指定的直接型 FIR 滤波器，对输入序列进行滤波。
IIRFilter	通过反向系数和前向系数指定的直接型 IIR 滤波器，对输入序列 X 进行滤波。
ZeroPhaseFilter	使用零相位滤波器对输入序列进行滤波，未进行边界延拓。
SignalOperation.Analog	
Invert	对数组中的每个元素求相反数。
Convolution	计算两个一维输入序列的卷积，使用线性卷积的 direct 方法。
AutoCorrelation	计算一维输入序列的自相关。
CrossCorrelation	计算两个一维输入序列的互相关，使用线性卷积的 direct 方法计算。
ZeroPadder	使输入序列输入数组的大小调整为下一个有效的 2 的整数次幂，序列中新增的尾部元素设为零。
UnwrapPhase	删除绝对值超过 pi 或 180 的不连续部分，展开相位数组。
Scale	删除输入信号 X 的偏移量并缩放结果，使输出序列在[－1:1]的区间。
Normalize	依据统计分布(μ,s)使输入向量归一化（μ 是均值，s 是标准差），得到统计分布为(0,1)的归一化向量。
Upsample	通过升采样因子在输入序列中插入零。
Downsample	通过降采样因子和平均布尔控件，对输入序列进行降采样。
Detrend	对输入序列进行去趋势。
WeightMovingAverage	对输入序列进行权重平均以平滑波动，类似于低通滤波。
ExponentialAverage	对输入序列进行指数平均以平滑波动，类似于低通滤波。
Resampling	对均匀输入序列使用新的采样率进行重采样，起始索引位置不变。
SignalOperation.Digital	
GetDigitalSize	返回数字输入中包含的采样和信号的数量
DigitalToBool	将数字信号转换为二维布尔数组
BoolToDigital	将二维布尔数组转换为数字信号
DigitalToBinary	将数字信号转换为一维 byte 数组； 将数字信号转换为一维 ushort 数组；

	将数字信号转换为一维 uint 数组。
BinaryToDigital	将 byte 数组转换为数字信号； 将 ushot 数组转换为数字信号； 将 uint 数组转换为数字信号。
Shift	对指定数字信号进行移位操作
Invert	对数字输入中的数字数据取反，使 0 变为 1（或 1 变为 0）
GetSubset	返回数字输入的子集，用于在图形上逐个绘制或按组绘制提取的信号
AppendSignals	合并两个数字信号。
AppendSamples	添加数字波形 src2 的所有采样至数字波形 src1 之后

2.4 SeeSharpTools.JY.GUI 类库

SeeSharpTools.JY.GUI 类库目前包含 22 种测试测量领域中常用 GUI 控件。

2.4.1 EasyChart

EasyChart 是基于微软 MSChart 控件实现的自定义控件，封装了最常用的单通道或多通道连续波形、单通道或多通道离散波形显示功能，包含的属性和方法如下表所示，其使用范例参见 SeeSharpExamples/ Graphical User Interface/JYEasyChart Basics。(该控件的功能已完全由 EasyChartX 替代，后续不再更新)

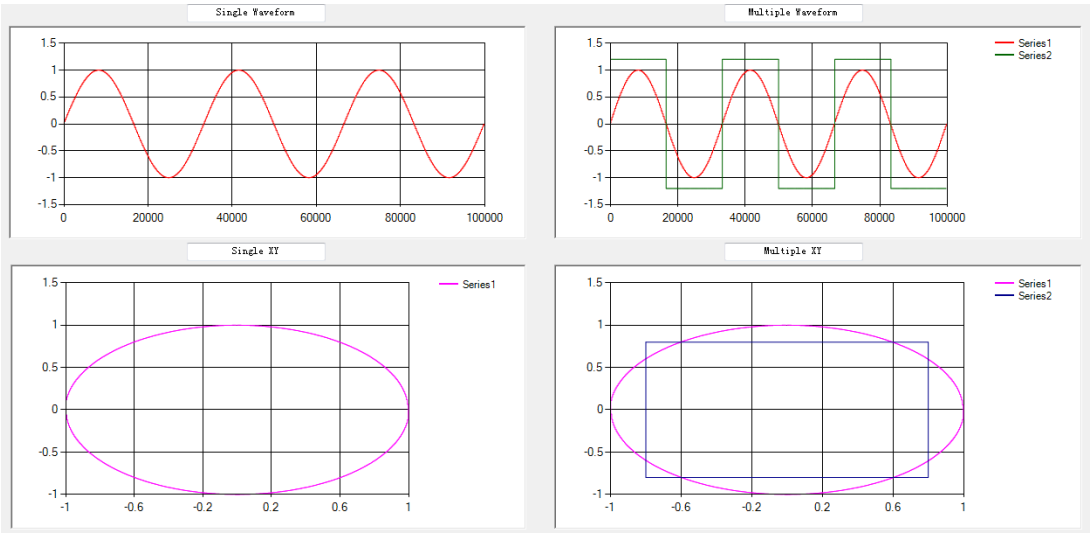
属性	
属性名称	**功能描述**
LegendVisible	设定是否在 Chart 中显示图例，默认为 True。
Palette	设定一系列颜色，依次用于 Chart 中各通道波形的颜色显示。
XAxisLogarithmic[1]	设定 X 轴是否采用对数显示，默认为 False。
YAxisLogarithmic[1]	设定 Y 轴是否采用对数显示，默认为 False。
EasyChartBackColor	设定 EasyChart 外框颜色
ChartAreaBackColor	设定 EasyChart 绘图区颜色
LegendBackColor	设定 Legend 背景色

SeriesNames	设定每条线在 Legend 中显示的名称
YAutoEnable	设定 Y 轴范围是否自动配置
AxisYMax	设定 Y 轴最大值
AxisYMin	设定 Y 轴最小值
方法	
Plot	显示单通道或多通道连续波形、单通道或多通道离散波形。
SaveAsCsv	保存当前图中的数据到 Csv 文件，有路径入参和无参两个重载
SaveAsImage	保存当前图表到 png 文件，有路径入参和无参两个重载

注 1：该属性在运行时不可设置。该属性设为 True 之后，调用 Plot 方法时输入的相应数据必须大于零。

在 Windows 窗体上放置 EasyChart 控件后，可以选中该 EasyChart 控件后在属性表格中对上述属性进行设置并查看其效果，除非特别说明，这些属性也可以在运行时设置。

EasyChart 支持单通道或多通道的连续或离散波形显示。此处的连续波形是指波形上任意相邻的两个点在 X 轴上均有相同的间隔，通常用于时域采样波形或频谱显示，在调用 Plot 方法显示连续波形时，除了输入波形数据，还可设定 X 轴上的起始位置和间隔；而离散波形是指波形上的任意一个点，都由独立的一对 x 和 y 来确定其在 Chart 上的位置，在调用 Plot 方法显示离散波形时，必须输出等长的两个数组，一一对应为各点在 Chart 上的位置。详情可参看 EasyChart 的例程，下图是 EasyChart 例程的运行效果图，4 个图分别对应单通道连续波形、多通道连续波形、单通道离散波形和多通道离散波形。



2.4.2 AquaGauge

AquaGauge 是一个仪表盘控件，用于显示数值，常用的属性如下图所示：

属性	
属性名称	功能描述
BackColor	配置表盘背景颜色。
Glossiness	反光镜效果调试比例（0<=&&<=100 的数值）
Max	表盘最大值
Min	表盘最小值
TextDescription	表盘描述说明
NumberOfDivisions	表盘分隔个数，数值范围（1<=&&<=25 的数值）
NumberOfSubDivisions	表盘子分隔个数，数值范围（1<&&<=10 的数值）
Value	表盘所显示的当前值

控件显示效果如下图所示，使用的方法请参考配套范例：SeeSharpExamples/ Graphical User Interface/AquaGuageControlExample。

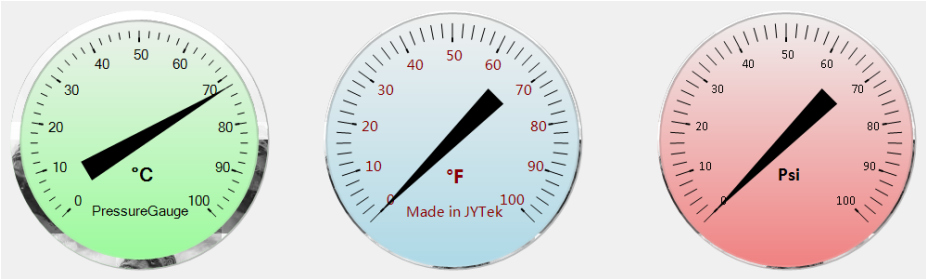
2.4.3 PressureGauge

PressureGauge 是一个压力表盘控件，功能类似于仪表盘，显示风格不同，该控件的常用属性如下表：

属性	
属性名称	功能描述
BackColor	配置表盘背景颜色
BorderWidth	配置表盘边框厚度
Glossiness	反光镜效果调试比例（0<=&&<=100 的数值）
Max	表盘最大值
Min	表盘最小值
DescriptionText	表盘描述说明
UnitText	表盘单位描述
NumberOfDivisions	表盘分隔个数，数值范围（1<=&&<=25 的数值）

NumberOfSubDivisions	表盘子分隔个数，数值范围（1<&&<=10 的数值）
Value	表盘所显示的当前值

控件的显示效果如下图所示，使用方法参见范例：SeeSharpExamples/ Graphical User Interface/PressureGaugeExample。

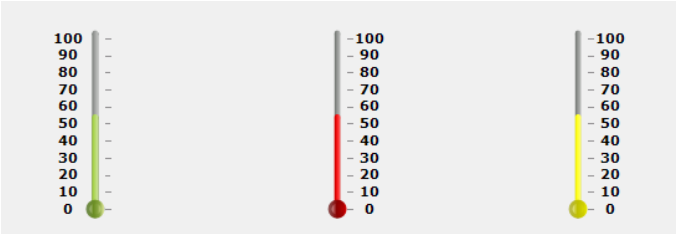


2.4.4 Thermometer

Thermometer 是一个温度计显示控件，可以用于显示温度等物理量。该控件常用属性如下表所示

属性	
属性名称	**功能描述**
LineWidth	配置温度计边界线宽
LineColor	配置温度计显示的颜色
ForeColor	配置温度计边界线的颜色
TickColor	配置温度计刻度线颜色
TickWidth	配置温度计刻度线线宽
BallSize	配置温度计球的大小
TextStyle	配置温度计刻度字体风格
TickStyle	配置温度计刻度风格
Value	配置温度计的显示值
Min	配置温度计显示的最小值
Max	配置温度计显示的最大值
NumberOfDivision	配置温度计刻度的个数
TextDecimals	配置温度计刻度显示的小数点位数

控件的显示效果如下图，详细使用方法请参照配套范例：SeeSharpExamples/ Graphical User Interface/ThermometerExample。

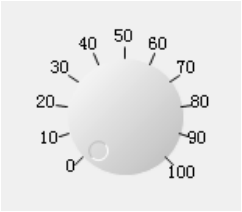


2.4.5 KnobControl

KnobControl 类是一个旋钮控件，该控件可以完成数值的输入操作。该控件的常用的属性方法如下表所示。

属性	
属性名称	功能描述
NumberOfDigits	数值显示时科学表达法的有效位数（0<=&&<=10 的数值）
Max	旋钮最大值
Min	旋钮最小值
NumberOfDivisions	旋钮分隔个数，数值范围（1~25 的数值）
TickVisible	Bool 参数，是否显示旋钮数值
Value	旋钮所表示的当前值
事件	
ValueChanged	默认事件。用户停止操作旋钮时触发。
ValueChanging	用户操作旋钮过程中触发。

控件的显示效果如下图所示，使用方法参见 SeeSharpExamples/ Graphical User Interface/KnobControlExamble。



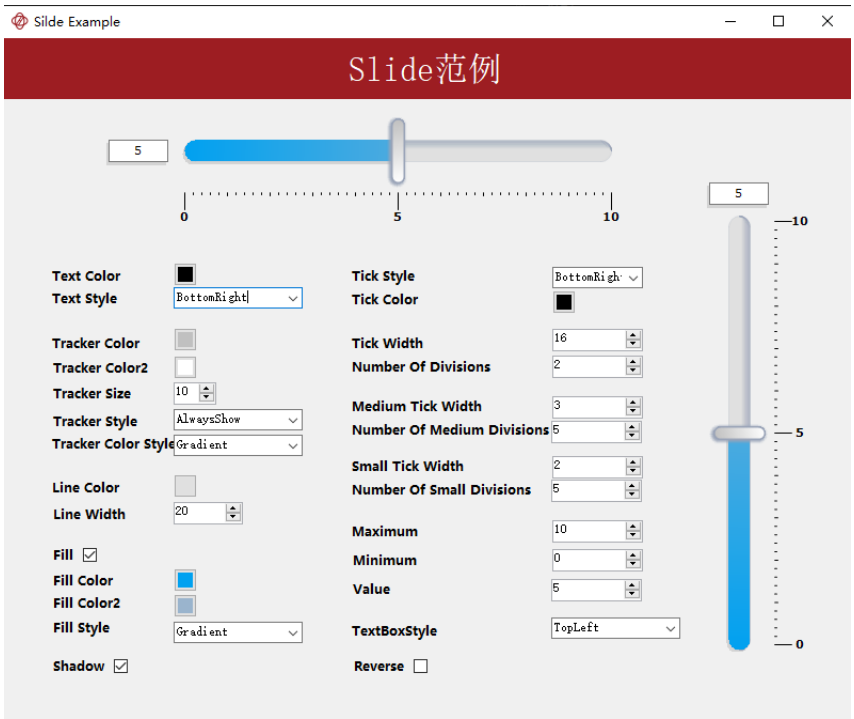
2.4.6 Slide

Slide 类是一个常用的滑杆控件，该控件可以完成数值的输入操作。控件的全部属性方法如下表所示。

属性	
属性名称	功能描述
Direction	滑动杆方向(从左到右或从右到左，从上到下或从下到上)
HeightIndent	滑动杆与控件边框之间的高度间隔
NumberOfDivisions	数值显示时的最大刻度分割数。用 TickWidth 调整最大刻度的长短
Orientation	滑动杆方向（Vertical，Horizontal）两种方向的选择
PaintShadow	是否绘制滑动杆的阴影
WidthIndent	滑动杆与控件边框之间的宽度间隔
NumberOfMediumDivisions	数值显示时的中等刻度分割数。用 TickMediumWidth 调整中等刻度的长短
NumberOfSmallDivisions	数值显示时的最小刻度分割数。用 TickSmallWidth 调整最小刻度的

	长短
TickColor	滑动杆 Tick 颜色选择
TickIndent	刻度与滑动杆之间的间隔
TickMediumWidth	滑动杆 Tick 的中等宽度配置
TickSmallWidth	滑动杆 Tick 的最小宽度配置
TickStyle	滑动杆 Tick 表示风格（None,TopLeft,BottomRight,Both）
TickWidth	滑动杆 Tick 的最大宽度配置
ForeColor	滑动杆 Tex 颜色选择
TextDecimals	数值文本显示的位数
TextStyle	滑动杆数值表示风格（None,TopLeft,BottomRight,Both）
TickTextIndent	刻度与数值文本之间的间隔
Fill	滑动杆主杆线是否填充
FillColor	滑动杆主杆线填充色
FillSecondColor	滑动杆主杆线填充色 2，用于渐变色风格
FillStyle	滑动杆主杆线填充风格，(Gradient 渐变，Aqua 圆润，Solid 实心)三种风格
LineColor	滑动杆主杆线颜色选择
LineWidth	滑动杆主杆线高度选择
TrackerColor	滑动块颜色选择
TrackerColorStyle	滑动杆滑块颜色风格，(Gradient 渐变，Aqua 圆润，Solid 实心)三种风格
TrackerSecondColor	滑动块颜色选择 2，用于渐变色风格
TrackerSize	滑动杆滑块大小，如果 with 与 Height 相等，则会成为圆形
TrackerStyle	滑动杆滑块显示风格，(None 不显示，Hide 悬浮式显示，AlwaysShow 永久显示)三种风格
LargeChange	值大改变；运行时，按 PgUp 或 PgDn 键
Max	滑动杆最大值
Min	滑动杆最小值
SmallChange	值小改变；运行时，按方向键
Value	滑动杆所表示的当前值
ValueDecimals	滑动杆分隔个数，数值范围（1<=&&<=25 的数值）
TextBoxDecimal	值文本框显示的小数点个数
TextBoxFont	值文本框的字体显示
TextBoxIndent	值文本框与滑动杆之间的间隔
TextBoxStyle	值文本框显示风格(None 不显示，TopLeft 上或左，BottomRIght 下或右，Both 都显示)四种风格
事件	
ValueChanged	默认事件。用户操作滑杆后触发
ValueChanging	用户操作滑杆过程中触发

控件的显示效果如下图所示，使用方法参见 SeeSharpExamples/ Graphical User Interface/SildeControlExample。

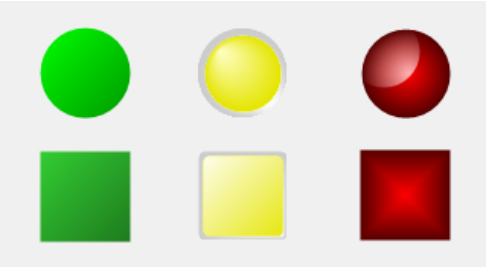


2.4.7 Led

Led 类是一个 LED 灯控件，该控件可完成 bool 数值的显示操作,控件常用的属性方法如下表所示：

属性	
属性名称	功能描述
BlinkColor	灯闪烁的颜色
BlinkInteral	灯光闪烁时长，单位毫秒（100<=&&<=5000 的数值）
BlinkOn	Bool 参数，是否打开闪烁功能
OffColor	灯光关闭时的颜色选择
OnColor	灯光打开时的颜色选择
Style	LED 样式选择，Circular,Rectangular
Value	Bool 参数，灯光开闭选择

控件的显示效果如下图所示。使用范例参见 SeeSharpExamples/ Graphical User Interface/SwitchControlExample。



2.4.8 EasyButton

EasyButton 类是 Button 的一个衍生类，可以预设部分图片在控件上，增强 Button 的显示效果。该控件的相关属性如下表所示：

属性	
属性名称	功能描述
BackgroundStyle	背景风格，(Normal 正常，Gradient 渐变)两种。(仅限 3D 模式)
ButtonColor	按钮颜色(仅限 3D 模式)
ButtonColor	按钮颜色 2，渐变模式下(仅限 3D 模式)
BorderColor	边框颜色(仅限 3D 模式)
BorderRadius	边框弧度(仅限 3D 模式)
BorderThickness	边框厚度 (仅限 3D 模式)
Shadow	开启阴影？(仅限 3D 模式)
ThreeDimensional	开启 3D 模式(仅限 3D 模式)
ButtonImage	图标的图片
ImageSize	图标大小
ImageStretch	图标是否拉伸，与字体同大小
PreSetImage	预选图标图片

控件的显示效果如下图所示，使用范例参见 SeeSharpExamples/ Graphical User Interface/SwitchControlExample

2.4.9 IndustrySwitch

IndustrySwitch 类是一个工业开关的控件类。控件的常用属性如下表所示，

属性	
属性名称	功能描述
OffColor	关状态下显示的颜色
OnColor	开状态下显示的颜色
Style	多种工业 Switch 复选功能
OffText	关状态下显示的文字

OnText	开状态下显示的文字
ShowText	是否显示状态文本？
Value	配置或获取当前控件的值
事件	
ValueChanged	默认事件，值改变后触发

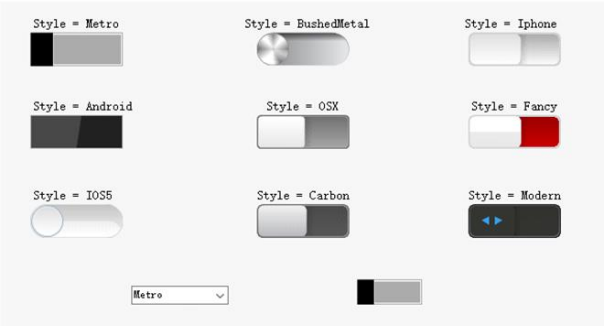
控件的显示效果如下图所示，使用方法参见范例 SeeSharpExamples/ Graphical User Interface/SwitchControlExample。

2.4.10 ButtonSwitch

ButtonSwitch 类是一个按钮开关的控件类，常用如下表所示。

属性	
属性名称	**功能描述**
Style	多种 Switch 复选功能
Value	Bool 参数，Switch 开关选择
事件	
ValueChanged	默认事件，值改变事件

控件的显示效果如下图所示，使用的方法参见范例 SeeSharpExamples/ Graphical User Interface/ButtonSwitchExample。



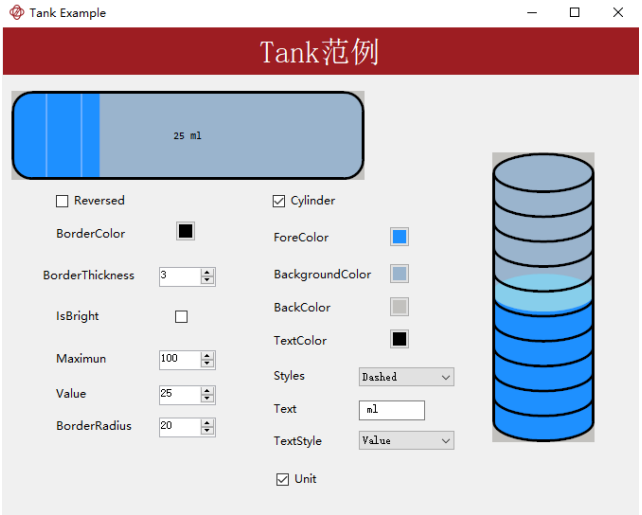
2.4.11 Tank

Tank 类是一个水罐样式的控件类，常用属性如下表所示：

属性	
属性名称	**功能描述**
BackgroundColor(Color)	背景颜色

BackgroundColorOpacity(Double)	背景颜色透明度，范围 0 到 1
BorderColor(Color)	边框颜色选择
BorderRadius(int)	边框弧度
BorderWidth(int)	边框厚度选择
Cylindrical(bool)	是否圆柱体显示？BorderRadius 需要 > 0
DashedInterval(int)	间隔线数量
IsBright(bool)	Bool 参数，是否选用亮色显示 Tank 控件
Orientation(Orientation)	Tank 摆放方向
Maximun(double)	Tank 数值最大值
Reversed(bool)	Tank 的水柱方向是否取反
Style(TankStyles)	Tank 样式选择，Solid,Dashed
Text(String)	Tank 显示文本(TextStyle 为 Text 样式时)
TextColor(Color)	Tank 文本显示颜色选择
TextStyle(TextStyleType)	Tank 文本显示样式选择(None, Text, Value）
Value(double)	Tank 所表示的当前值
Unit(String)	Tank 值显示单位(TextStyle 为 Value 样式时)

控件的使用效果如下图所示，使用方法参见范例：SeeSharpExamples/ Graphical User Interface/TankControlExample



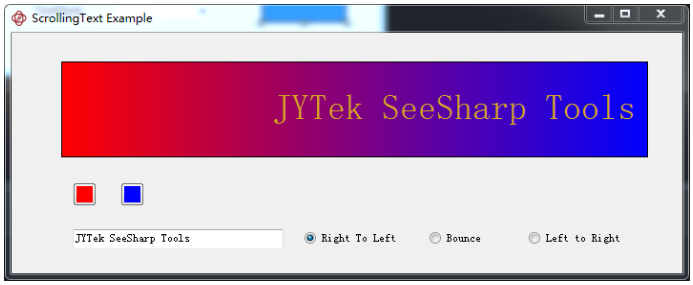
2.4.12 ScrollingText

ScrollingText 类是一个滚动显示字符串的控件，常用属性如下表所示：

属性	
属性名称	**功能描述**
BorderColor	边框颜色选择

BorderThickness	边框厚度选择
ScrollDirection	文本滚动方向选择
BorderVisible	Bool 参数，是否需要显示边框
Text	滚动说明文本
SrollSpeed	滚动文本速度（1<=&&<=1000 的数值）
VerticleAligment	设置滚动条文字位置

控件的显示效果如下图所示，使用方法参见范例：SeeSharpExamples/ Graphical User Interface/ScrollingTextExample。



2.4.13 SevenSegment

SevenSegment 类是一个七段数码管控件类，常用属性如下表所示：

属性	
属性名称	**功能描述**
BackColor	背景颜色
DarkColor	数码管暗时颜色
LightColor	数码管亮时颜色
DecimalShow	是否显示小数点位置
ItalicFactor	数码管倾斜系数
NumberOfChar	数码管总个数
Value	数码管显示内容

控件的显示效果如下图所示，使用方法参见范例：SeeSharpExamples/ Graphical User Interface/SevenSegment。

2.4.14 SegmentBright

SegmentBright 类是一个七段数码管控件类，显示效果和 SevenSegment 不同，有更

多的颜色选择。常用属性如下表所示：

属性	
属性名称	功能描述
BackColor1	背景颜色 1
BackColor2	背景颜色 2 ，由背景颜色 1 与背景颜色 2 共同混合为新的背景色
GradientBackground	是否显示渐变背景色效果
BorderColor	边框颜色
BorderWith	边框宽度
DarkColor	数码管不显示时颜色
ForColor	数码管显示时的颜色选择
HighlightOpaque	显示半透明颜色，默认值为 50
NumberOfChar	数码管总个数

控件的显示效果如下图所示，使用方法参见范例 SeeSharpExamples/ Graphical User Interface/SegmentBright。

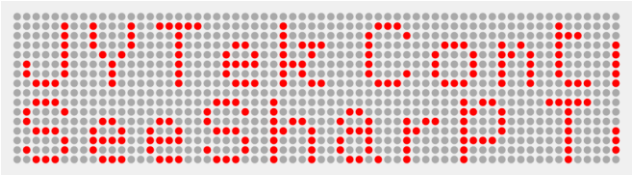
2.4.15 LedMatrixControl

LedMatrixControl 类是一个 Led 矩阵展示控件类，常用属性如下表所示：

属性	
属性名称	功能描述
LedOffColor	矩阵灯关闭颜色选择
LedOnColor	矩阵灯打开颜色选择
LedStyle	矩阵灯样式选择（circle,square）
LedRows	Led 矩阵灯的行数选择
LedColumns	Led 矩阵灯的列数选择
LedSizeCoeff	Led 灯的大小比例系数（0<=&&<=1 的数值）
Items	Location 配置当前显示行 Text 的位置
	Direction 配置当前显示行 Text 滚动方向(Up,Down,Left,Right)
	Speed 配置当前显示行 Text 滚动速度

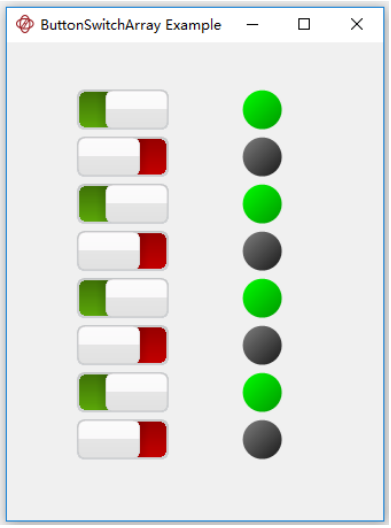
方法	
StartMove	矩阵 Led 灯滚动开始
StopMove	矩阵 Led 灯滚动停止

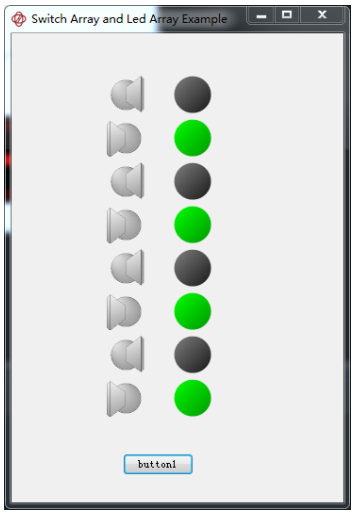
控件显示效果如下图所示，使用方法参见范例：SeeSharpExamples/ Graphical User Interface/LedMatrixControlExample。



2.4.16 SwitchArray、ButtonSwitchArray、LEDArray

SwitchArray、ButtonSwitchArray 和 LEDArray 使用了 SeeSharpTools.GUI 中 IndustrySwith 控件，ButtonSwitch 控件以及 LED 控件作为数组型态呈现。控件支持鼠标控制开关、程序变更数值（单一或数组类型），同时数值变更事件能够让用户界面注册使

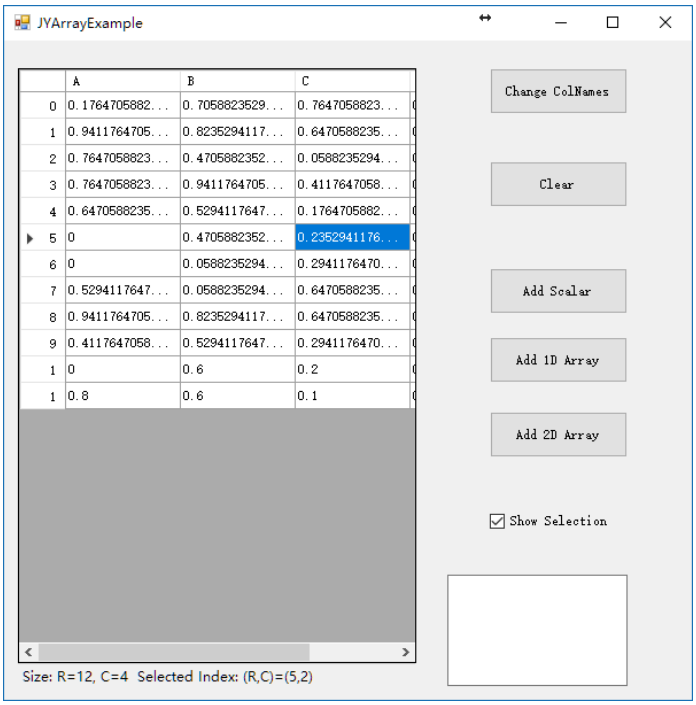




用。控件的显示效果如下图所示，使用方法参见范例：SeeSharpExamples/ Graphical User Interface/JYGUIArrayExample。

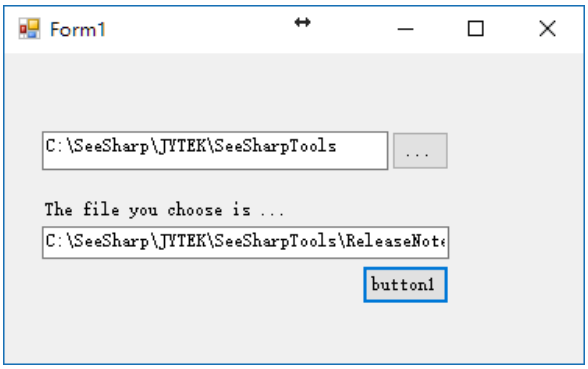
2.4.17 JYArray

JYArray 类提供二维数组显示界面，帮助测量数据呈现。控件支持数组累加呈现、变更列表名称、显示选取资讯、显示栏位号等功能。同时可使用属性提取全部资料。控件的显示效果如下图所示。



2.4.18 PathControl

PathControl 类是一个接收用户配置路径的控件类。PathControl 提供四种输入模式，分别是浏览按钮、文字框输入、鼠标拖放以及程序输入。并可利用属性将路径读出供程序使用。该控件的显示效果如下图所示：



2.4.19 StripChart

StripChart 是基于微软 MSChart 控件实现的自定义控件，实现了单通道和多通道小型记录仪的功能。常用属性和方法如下

属性

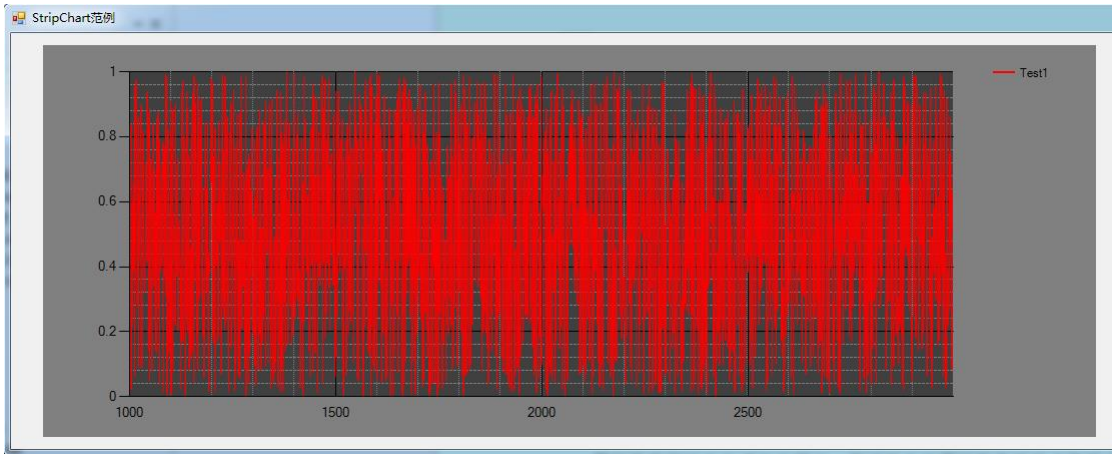
属性名称	功能描述
LegendVisible	设定是否在 Chart 中显示图例，默认为 True。
Palette	设定一系列颜色，依次用于 Chart 中各通道波形的颜色显示。
YAxisLogarithmic	设定 Y 轴是否采用对数显示，默认为 False。
ChartBackColor	设定 StripChart 外框颜色
ChartAreaBackColor	设定 StripChart 绘图区颜色
MajorGridEnabled	设定是否使能主网格线
MinorGridEnabled	设定是否使能辅网格线
LineNum[1]	设定图中待显示的线数(Plot 时会自动更新为真实线数)
LineWidth	设定每条线的线宽
SeriesNames	设定每条线在 Legend 中显示的名称
YAutoEnable	设定 Y 轴范围是否自动配置
AxisYMax	设定 Y 轴最大值
AxisYMin	设定 Y 轴最小值
Displaydirection	设置记录仪滚动的方向，可以从左向右或者从右向左
DisPlayPoints[1]	设置 StripChart 中每条线可以显示的最多点数，超过点数后绘图区开始滚动显示
XAxisTypes	X 轴显示类型，可以是索引、时间戳、用户输入值。默认为索引。
TimeStampFormat	设定时间戳的显示格式
NextTimeStamp	在时间戳模式下，配置或获取下一个时间戳
TimeInterval	时间戳模式下，配置两个样点的时间差
XAxisStartIndex	设定索引的开始数值
XAxisTitle	X 轴的标题
XTitlePosition	X 轴标题位置
XTitleOrientation	X 轴标题方向
YAxisTitle	Y 轴的标题
YTitlePosition	Y 轴标题位置
YTitleOrientation	Y 轴标题方向
ScrollType	StripChart 滚动模式
方法	
Plot	向 StripChart 的每条线写入多个点。
PlotSingle	向 StripChart 的每条线写入一个点
Clear	清空当前绘图

注 1：有 1 标注的属性在运行时不可设置。该属性设为 True 之后，调用 Plot 方法时输入的相应数据必须大于零。

在 Windows 窗体上放置 StripChart 控件后，可以选中该 StripChart 控件后在属性表格中对上述属性进行设置并查看其效果，除非特别说明，这些属性也可以在运行时设置。

StripChart 支持单次写入每条线的多个点(Plot)或者每条线的单个点(PlotSingle)。写入后的点都将在 StripChart 中显示，当每条线写入的点数超过 DisplayPoints 配置的值时绘

图区数值开始向左或向右滚动，保证单个绘图区每条线的点数不会超过 DisplayPoints。DisplayPoints 等于 2000，总绘制点数为 3000 时 StripChart 的显示效果如下。控件的使用方法参见范例：SeeSharpExamples/ Graphical User Interface/StripChartExamples。

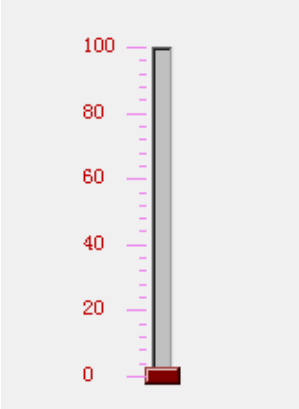


2.4.20 GaugeLinear

GaugeLinear 类是一个滑杆控件，该控件可实现数值的输入操作，同时该控件还支持设置数值滑杆/滑块/刻度的颜色，同时该控件的刻度会根据当前的空间大小自动匹配。控件常用的属性方法如下表所示。

属性	
属性名称	**功能描述**
BackColor	配置控件的背景色
Maximum	滑杆的最大值
Minimum	滑动杆最小值
Orientation	滑动杆方向（Vertical，Horizontal）两种方向的选择
SlideDirection	滑动杆数值递增方向（LeftToRight, RightToLeft）两种方向的选择
Value	配置或获取滑块位置对应的数值
事件名称	**事件描述**
ValueChanged	控件的值被修改时触发

控件的显示效果效果如下图所示，使用方法参见范例：SeeSharpExamples/ Graphical User Interface/GaugeLinearExample。



2.4.21 EasyChartX

EasyChartX 是基于微软 MSChart 控件实现的自定义控件，该控件是 EasyChart 的升级版本。EasyChartX 中加入了测试测量很多常用的特性、尽量简化了对外接口、提高了绘图效率，同时新增分区视图，可以将每条线独立显示到不同的绘图区中。EasyChartX 的属性、事件和方法如下表所示：

属性	
属性名称(类型)	**功能描述**
LegendVisible(bool)	设定是否在 Chart 中显示图例，默认为 True。
BackColor(Color)	设定 EasyChartX 外框颜色
ChartAreaBackColor(Color)	设定 EasyChartX 绘图区颜色，该配置同时生效于两种视图
AutoClear(bool)	设定每次绘图前是否清除已有的数据。
ForeColor(Color)	坐标轴 Label 和名称的字体颜色
Font(Font)	坐标轴 Label 的字体
LegendBackColor(Color)	图例的背景色
LegendFont(Font)	图例的字体
LegendForeColor(Color)	图例的文字颜色
SeriesCount(int)	配置未绘图时显示的线条个数、获取当前绘图中的线条个数。
GradientStyle(ChartGradientStyle)	背景色渐变类型
SplitView(bool)	是否分区视图显示，默认为 false
Miscellaneous (MiscellaneousConfiguration)	杂项配置
MarkerSize(int)	Marker 的大小，取值范围[3-21]

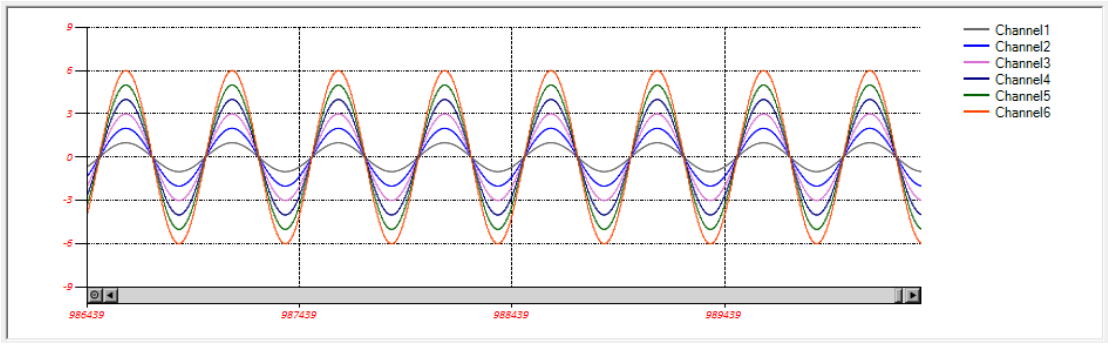
	ShowFunctionMenu(bool)	配置在 Chart 上右键点击，是否显示功能菜单
	MaxSeriesCount(int)	最大线条个数，默认 32，超过该值的数据在 Chart 不显示
	MaxSeriesPointCount(int)	单个线条上最多显示的点数，默认 4000，增大该值会提升数据显示的细腻程度，但是会降低性能。
	CheckNaN(bool)	是否校验数据中的 NaN 数据。如果不检查则 Nan 显示为空点；如果检查则 NaN 替换为 1.5E-200 并显示为空点。
	CheckNegtiveOrZero(bool)	是否校验非正数。如果不检查 Nan，则非正数替换为空点；如果检查 Nan，则非正数替换为 1E200 和 1E-200
	CheckInfinity(bool)	是否校验 Inf 数据。如果不检查 Nan，则 Inf 替换为空点；如果检查 Nan 则 Inf 替换为 1E200 和-1E200/1E-200。
	DataStorage(DataStorageType)	配置绘图数据的缓存类型，包括拷贝、浅拷贝。
	Fitting(EasyChartX.FitType)	筛点拟合算法选择。大数据时使用 None 可改善绘图速度，但是可能会导致数据趋势的显示错误。
	SplitLayoutDirection (LayoutDirection)	分区视图的布局方向：从左到右或者从上到下
	SplitViewAutoLayout(bool)	配置分区视图是否自动布局，默认为 true。配置为 false 时 DirectionChartCount 会生效。
	DirectionChartCount(int)	布局方向的绘图区个数(SplitLayoutDirection 为 LeftToRight 时为一行的绘图区个数，TopToDown 时为一列的绘图区个数)，该属性在 SplitViewAutoLayout 为 False 时生效
	SplitLayoutColumnInterval(int)	分区视图每列绘图区之间的像素间隔，可以为负值
	SplitLayoutRowInterval(int)	分区视图每行绘图区之间的像素间隔，可以为负值
LineSeries(设计时属性名) Series(开发时属性名) (EasyChartXSeriesCollection)		设定每个线条的属性。该属性的配置个数可以超过当前线条数。新增线条按照对应索引进行配置。
	Name(string)	设定线条名称，该名称会被显示到图例中
	Color(Color)	线条颜色
	Visible(bool)	线条是否可见
	Width(LineWidth)	线宽
	XPlotAxis(PlotAxis)	使用哪个 X 轴，当前版本只可以使用主 X 轴。
	YPlotAxis(PlotAxis)	使用哪个 Y 轴，可以选择主 Y 轴和副 Y 轴
	Type(LineType)	线条类型
	Marker(MarkerType)	点标记类型。只有在 Type 为 Line 时生效
SplitPlotArea (EasyChartXPlotAreaCollection)		配置和获取分区视图时各个绘图区的属性
	BackColor(Color)	绘图区背景色
	AxisX(EasyChartXAxis)	主 X 坐标轴
	AxisX2(EasyChartXAxis)	副 X 坐标轴
	AxisY(EasyChartXAxis)	主 Y 坐标轴
	AxisY2(EasyChartXAxis)	副 Y 坐标轴
	XCursor(EasyChartXCursor)	X 轴游标
	YCursor(EasyChartXCursor)	Y 轴游标

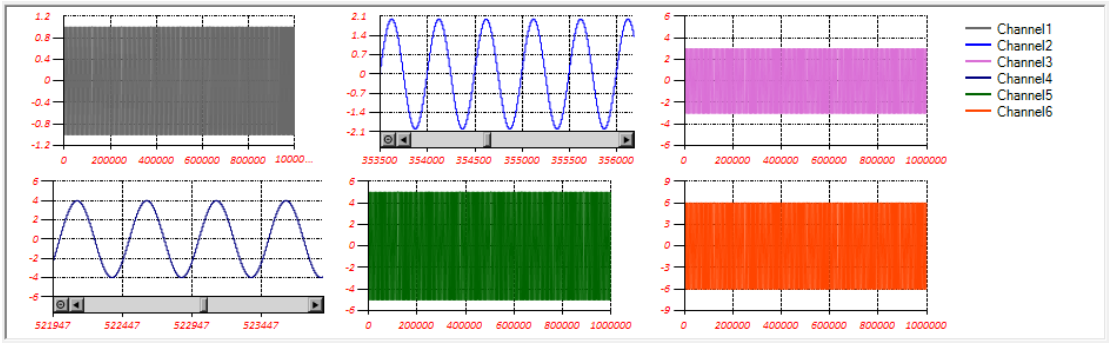
Axes(EasyChartXAxis[]) AxisX(EasyChartXAxis) AxisX2(EasyChartXAxis) AxisY(EasyChartXAxis) AxisY2(EasyChartXAxis)		坐标轴配置，可以配置 X 轴、副 X 轴、Y 轴、副 Y 轴的相关属性。该配置同时生效于两种视图。
	Name(string)	获取坐标轴名称
	Color(Color)	配置坐标轴颜色
	Title(string)	配置坐标轴标题
	TitleOrientation (AxisTextOrientation)	配置坐标轴标题显示方向
	TitlePosition (AxisTextPosition)	配置坐标轴标题显示位置
	LabelEnabled(bool)	配置是否显示坐标轴的值
	LabelFormat(string)	配置坐标轴下方标签的格式
	MajorGridEnabled(bool)	主网格使能
	MajorGridColor(Color)	主网格颜色
	MajorGridType(GridType)	主网格线条类型
	MinorGridEnabled(bool)	副网格使能
	MinorGridColor(Color)	副网格颜色
	MinorGridType(GridType)	副网格线条类型
	TickWidth(double)	配置坐标轴上标记线的宽度和 Chart 高/宽的比值
	AutoScale(bool)	坐标轴范围是否自动配置
	AutoScalingMode	在坐标轴自动配置模式下，选择(ByGridCount 按网格数量自动配置，ByWholeNumbers 按坐标轴值取整自动配置)模式
	Maximum(double)	获取当前坐标轴最大值。未绘图时配置坐标轴最大值；绘图时且 AutoScale 为 false，可以配置坐标轴最大值。
	Minimum(double)	获取当前坐标轴最小值。未绘图时配置坐标轴最小值；绘图时且 AutoScale 为 false，可以配置坐标轴最小值。
	IsZoomed(bool)	获取当前坐标轴是否处于缩放状态
	AutoZoomReset(bool)	配置每次绘图时是否取消缩放。
	InitWithScaleView(bool)	配置绘图开始时是否默认使用缩放视图。
	ViewMaximum(double)	获取当前坐标轴缩放视图最大值，InitWithScaleView 为 true 时配置开始绘图时坐标轴缩放视图最大值
	ViewMinimum(double)	获取当前坐标轴缩放视图最小值，InitWithScaleView 为 true 时配置开始绘图时坐标轴缩放视图最小值
	IsLogarithmic(bool)	当前坐标轴是否使用指数显示
	LogarithmBase(double)	指数显示下的底数
	LogLabelStyle	指数显示下的标签的显示格式(F0, E2)
	ShowLogarithmicLines(bool)	指数显示下是否显示辅助线
	Zoom(double max, double min)	将当前坐标轴缩放到两个数值的范围内
	ZoomReset(int resetTimes)	回退坐标轴缩放操作
Cursors(EasyChartXCursor[])		游标配置，可以配置 X 游标、Y 游标的相关属性。该配置

XCursor(EasyChartXCursor) YCursor(EasyChartXCursor)		同时生效于两种视图。
	Name(string)	获取游标名称
	Mode(CursorMode)	游标类型，包括非使能、游标和区域选择
	Color(Color)	游标颜色
	SelectionColor(Color)	游标区域选择时选择区的颜色
	AutoInterval(bool)	是否自动配置选择间隔
	Interval(double)	AutoInterval 时配置游标选择间隔
	Value(double)	游标模式时获取游标当前所在的位置
TabCursors(TabCursorCollection)		标签游标的集合
	CursorValueFormat(string)	鼠标在标签游标上时，显示数值的格式
	void Add(TabCursor item)	添加标签游标
	void Clear()	清空标签游标
	TabCursor(TabCursors 中的元素)	标签游标类
	TabCursor.Color(Color)	游标颜色
	SeriesIndex(int)	TabCursor 绑定到的曲线的索引号，小于 0 为未绑定。
	TabCursor.Value(double)	游标当前对齐到的 X 轴的值
	TabCursor.YValue(double)	当 Direction 为 Vertical 时记录游标对齐的 X 轴对应曲线的 Y 值，只读，只有 SeriesIndex 配置了有效值时可用；Direction 为 Horizental 时可读写。
	TabCursor.Direction (TabCursorDirection)	标签游标的方向，可以为 Horizental 和 Vertical。当该值为 Horizental 时不能跟踪数据点，XValue 和 SeriesIndex 无效。
	TabCursor.Name(string)	游标名称
	TabCursor.Enabled(bool)	是否显示游标
自定义事件		
AxisViewChanged		坐标轴范围变化时触发该事件
	EasyChartXViewEventArgs	事件参数： Axis：触发事件的坐标轴 ParentChart：触发事件的 EasyChartX 实例 IsScaleViewChanged：是否是缩放视图变更触发的事件 IsRaisedByMouseEvent：是否是鼠标操作触发的事件
CursorPositionChanged		用户使用游标选择对齐到某条线后触发该事件
	EasyChartXCursorEventArgs	事件参数： Cursor：触发事件的游标 ParentChart：触发事件的 EasyChartX 实例 SeriesIndex：触发 Cursor 事件的线条索引 IsRaisedByMouseEvent：是否鼠标操作触发
TabCursorChanged		TabCursor 被操作或移动后触发该事件
	TabCursorEventArgs	事件参数： Cursor：触发该事件的 TabCursor ParentChart：触发该事件的 EasyChartX Operation：事件触发操作类型：值被修改、添加 Cursor、

		删除 Cursor ParentChartAre：TabCursor 所在的绘图区对象
BeforePlot		在绘图区绘制曲线前触发该事件
	EasyChartXPlotEventArgs	事件参数： IsClear：该事件是否由 Clear 触发 ParentChart：触发该事件的 EasyChartX 句柄
AfterPlot		在绘图区绘制曲线后触发该事件
	EasyChartXPlotEventArgs	IsClear：该事件是否由 Clear 触发 ParentChart：触发该事件的 EasyChartX 句柄
方法		
Plot<TDataType>(…)		显示单通道或多通道连续波形、单通道或多通道离散波形。该方法二维数组的重载包含 majorOrder(MajorOrder)的入参，用以配置按行绘图还是按列绘图。 Plot 方法的泛型重载支持：double/float/int/uint/short/ushort/byte 数据。
AddDataMarker(…)		在当前绘图中添加数据标记，这些标记在下次 Plot 或者 Clear 时会被清空。单次绘图可以多次添加 Marker。
SaveAsCsv(string filePath)		保存当前图中的数据到 Csv 文件，有路径入参和无参两个重载
SaveAsImage(string filePath)		保存当前图表到 png 文件，有路径入参和无参两个重载
ZoomReset(double min, double max)		停止所有坐标轴的缩放
IsPlotting()		返回当前图表是否处于绘图状态。刚打开未执行 Plot 方法或者刚执行过 Clear 方法时该方法返回 False，其余情况返回 True。
Clear()		清除当前的所有绘图

在 Windows 窗体上放置 EasyChartX 控件后，可以选中该 EasyChartX 控件后在属性表格中对上述属性进行设置。EasyChartX 支持单通道或多通道的连续或离散波形显示，波形的数值范围应该在[-2.5E28, 2.5E28]，超出该范围会导致 Chart 抛出运行时异常。此处的连续波形是指波形上任意相邻的两个点在 X 轴上均有相同的间隔，通常用于时域采样波形或频谱显示，在调用 Plot 方法显示连续波形时，除了输入波形数据，还可设定 X 轴上的起始位置和间隔；而离散波形是指波形上的任意一个点，都由独立的一对 x 和 y 来确定其在 Chart 上的位置，在调用 Plot 方法显示离散波形时，必须输出等长的两个数组，一一对应为各点在 Chart 上的位置。详细使用方法参照 EasyChartX 的例程，下图是一组正弦信号在 EasyChartX 两种模式的绘图效果。使用方法参见范例 SeeSharpExamples/EasyChartX Example。





本节下的子章节将会详细描述各个场景下使用和配置 EasyChartX 的方法。

2.4.21.1 EasyChartX 的 Plot 接口及其使用

EasyChartX 中定义的绘图接口和参数说明如下列表所示：

- `Plot<TDataType>(TDataType [] yData, double xStart = 0, double xIncrement = 1)`
  - 功能说明：绘制单条曲线，Y 轴数据储存在一维数组中，X 轴为递增的数据。TDataType 为泛型，支持的数据类型包括：double、float、int、uint、short、ushort、byte。
  - yData：包含待绘制 Y 轴数据的数组。
  - xStart：X 轴数据的起始值，默认值为 0。
  - xIncrement：X 轴每个样点对应的步进值，默认值为 1。
  - 示例：

```
double[] yData = new double[] { 4, 5, 6 };
easyChartX1.Plot(yData, xStart: 1, xIncrement: 2);
```

- `Plot<TDataType>(TDataType[,] yData, double xStart = 0, double xIncrement = 1, MajorOrder majorOrder = MajorOrder.Row)`
  - 功能说明：根据指定的方向绘制多条曲线，Y 轴的数据储存在二维数组中，X 轴为递增的数据。TDataType 为泛型，支持的数据类型包括：double、float、int、

uint、short、ushort、byte。

- yData：包含待绘制 Y 轴数据的数组，第四个参数 majorOrder 的值为 Row 时每一行对应一条曲线，绘制的曲线个数等于 yData 的行数；majorOrder 的值为 Column 时每一列对应一条曲线，绘制的曲线个数等于 yData 的列数。
- xStart：X 轴数据的起始值，默认值为 0。
- xIncrement：X 轴每个样点对应的步进值，默认值为 1。
- majorOrder：枚举类型，配置绘图的方向，值为 Row 是按行绘图；值为 Column 时按列绘图。默认值为 Row。
- 示例：

```
double[,] yData = new double[,] { { 4, 5 }, { 6, 7 }, { 8, 9 } };
easyChartX1.Plot(yData, xStart: 1, xIncrement: 2, MajorOrder.Column);
```

- Plot<TDataType>(TDataType1 [] xData, TDataType2[] yData)
  - 功能说明：使用数组绘制一条曲线，X 轴的数据储存在 xData 中，Y 轴的数据保存在 yData 中，两个数组的长度必须相等。TDataType1 和 TDataType2 为泛型，支持的数据类型包括：double、float、int、uint、short、ushort、byte。TDataType1 和 TDataType2 的类型可以不同。
  - xData：包含待绘制 X 轴数据的数组。
  - yData：包含待绘制 Y 轴数据的数组。
  - 示例：

```
double[] xData = new double[] { 1, 2, 3 };
double[] yData = new double[] { 4, 5, 6 };
easyChartX1.Plot(xData, yData);
```

- Plot<TDataType>(double[] xData, double[,] yData, MajorOrder majorOrder = MajorOrder.Row)
  - 功能说明：使用数组绘制两条曲线，X 轴的数据储存在 xData 中，Y 轴的数据保存在 yData 中。
  - xData：包含待绘制 X 轴数据的数组。
  - yData：包含待绘制 Y 轴数据的数组，第四个参数 majorOrder 的值为 Row 时每一行对应一条曲线，绘制的曲线个数等于 yData 的行数，此时 yData 的列数必须和 xData 的长度相等；majorOrder 的值为 Column 时每一列对应一条曲线，绘制的曲线个数等于 yData 的列数，此时 yData 的行数必须和 xData 的长度相等。
  - majorOrder：枚举类型，配置绘图的方向，值为 Row 是按行绘图；值为 Column 时按列绘图。默认值为 Row。
  - 示例：

```
double[] xData = new double[] { 1, 2, 3 };
double[,] yData = new double[,] { { 4, 5 }, { 6, 7 }, { 8, 9 } };
easyChartX1.Plot(xData, yData, MajorOrder.Column);
```

- Plot<TDataType>(TDataType1[][] xData, TDataType2[][] yData)
  - 功能说明，使用数组绘制多条独立的曲线，每一条曲线对应 xData 和 yData 的一个元素。xData 的长度必须和 yData 的长度相等，且每个 xData 的元素数组的长度和对应索引的 yData 的元素数组的长度必须相等。TDataType1 和 TDataType2 为泛型，支持的数据类型包括：double、float、int、uint、short、ushort、byte。TDataType1 和 TDataType2 的类型可以不同。
  - xData：包含待绘制 X 轴数据的数组，数组的数组，每个元素数组长度可以不同，元素数组的个数即为绘制的曲线个数。
  - yData：包含待绘制 Y 轴数据的数组，数组的数组，元素数组的个数必须和 xData 的元素数组个数相等，每个元素数组长度必须和 xData 对应索引的数组长度相等，
  - 示例

```
double[][] xData = new double[2][];
xData[0] = new double[] { 1, 2, 3 };
xData[1] = new double[] { 1, 2, 3, 4};
double[][] yData = new double[2][];
yData[0] = new double[] { 7, 8, 9 };
yData[1] = new double[] { 4, 5, 6 };
easyChartX1.Plot(xData, yData);
```

- Plot(IList<double> yData, double xStart = 0, double xIncrement = 1, int xSize = 0)
  - 功能说明，绘制单条或多条曲线，Y 轴数据储存在任意实现了 IList 接口的集合中，X 轴为递增的数据。如果 xSize = 0 则默认 xSize 等于 yData.Count，绘制的曲线个数为 yData.Count/xSize。
  - 包含待绘制 Y 轴数据的集合，yData 可以是任意实现了 IList<double>接口的类，例如 double[]、List<double>或者任意用户自己定义的类。
  - xStart：X 轴数据的起始值，默认值为 0。
  - xIncrement：X 轴每个样点对应的步进值，默认值为 1。
  - xSize：X 轴数据的长度，如果 xSize 的值为 0，则 xSize 的值等于 yData.Count。
  - 示例：

```
IList<double> yData = new List<double>() { 2, 3, 4, 5, 6, 7 };
easyChartX1.Plot(yData, xStart:0, xIncrement:1, xSize:3);
```

- Plot(IList<double> xData, IList<double> yData)
  - 功能说明，绘制单条或多条曲线，X 轴和 Y 轴数据储存在任意实现了 IList 接口的集合中。绘制的曲线个数为 yData.Count/xData.Count。
  - 包含待绘制 X 轴数据的集合，xData 可以是任意实现了 IList<double>接口的类，例如 double[]、List<double>或者任意用户自己定义的类。
  - 包含待绘制 Y 轴数据的集合，yData 可以是任意实现了 IList<double>接口的类，例如 double[]、List<double>或者任意用户自己定义的类。
  - 示例

```
IList<double> xData = new List<double>() { 1, 2, 3 };
IList<double> yData = new List<double>() { 2, 3, 4, 5, 6, 7 };
easyChartX1.Plot(xData, yData);
```

- Plot(IList<IList<double>> xData, IList<IList<double>> yData)
  - 功能说明，绘制多条独立的曲线，X 轴和 Y 轴是任意 IList<double>接口的 IList 集合，xData 的长度和 yData 的长度必须相等，同时所有 xData 的元素集合和相同索引的 yData 元素集合的长度必须相等。
  - 包含所有待绘制 X 轴数据的列表的集合。
  - 含所有待绘制 Y 轴数据的列表的集合。
  - 示例

```
IList<IList<double>> xData = new List<IList<double>>()
{
    new List<double>() { 1, 2, 3, 4 },
    new double[] { 1, 2, 3 }
};
IList<IList<double>> yData = new List<IList<double>>()
{
    new double[] { 4, 5, 6, 7 },
    new List<double>() { 7, 8, 9 },
};
easyChartX1.Plot(xData, yData);
```

2.4.21.2 自定义 EasyChartX 的外观

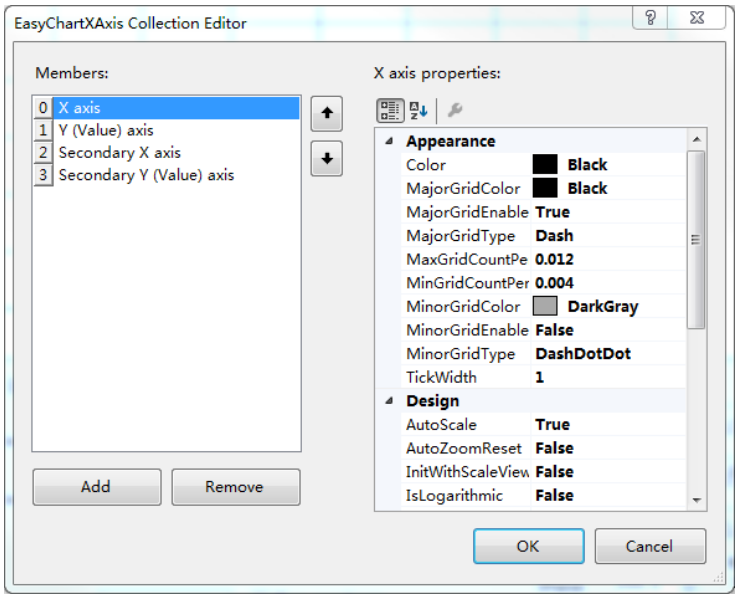
EasyChartX 的外观使用一组属性进行配置，常用的外观配置方法如下说明。

- **修改 EasyChartX 的背景色：**配置 BackColor 属性为指定的颜色。
- **修改绘图区的颜色：**配置 ChartAreaBackColor 为指定的颜色。
- **修改图表上显示的字体和颜色：**修改 Font(字体)和 ForeColor(字体颜色)为指定的值。
- **修改图例是否显示：**配置 LegendVisible 的值为 True(显示)或 False(不显示)。
- **修改图例的字体和颜色：**配置 LegendFont 和 LegendForeColor 属性。
- **修改图例的背景色：**配置 LegendBackColor 属性。

2.4.21.3 配置 EasyChartX 的坐标轴

EasyChartX 一共包括四个坐标轴：X 坐标轴(Axis X)、Y 坐标轴(Axis Y)、副 X 坐标轴(Axis X2)、副 Y 坐标轴(Axis Y2)。这些坐标轴可以实现很多自定义配置，这些配置在设计界面和应用运行时都可以修改。

在界面编辑页面，左键选择 EasyChartX 并右键，在菜单中点击“属性”菜单会弹出属性配置窗体，在属性列表轴找到 Axes 属性，点击右侧的“…”按钮即可弹出坐标轴编辑界面，该界面如下图所示。



在应用运行时，可以使用 easyChartX.AxisX、easyChartX.AxisY、easyChartX.AxisX2、easyChartX.AxisY2 分别配置坐标轴的各项参数。

常用的坐标轴自定义以及配置方式如下描述：

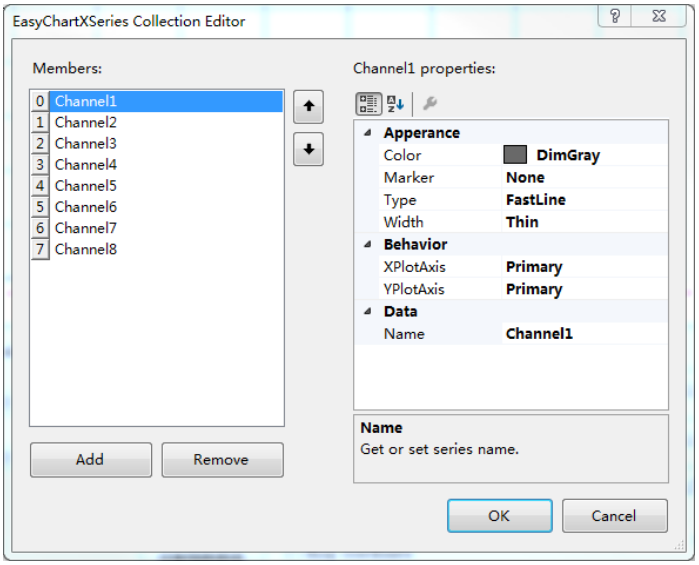
- **配置坐标轴上标签的显示格式：**配置对应坐标轴的 LabelFormat 属性，配置的方式参见 MSDN 链接：Standard numeric format strings | Microsoft Docs。
- **配置坐标轴是否显示标签以及标签的显示方向：**配置对应坐标轴的 LabelEnabled 属性(是否使能)和 LabelAngle 属性(标签显示方向)。
- **配置坐标轴上是否显示主网格：**配置对应坐标轴的 MajorEnabled 属性(是否显示)、MajorGridColor 属性(主网格颜色)、MajorGridType 属性(主网格线条类型)。
- **配置坐标轴上是否显示副网格：**配置对应坐标轴的 MinorEnabled 属性(是否显示)、MinorGridColor 属性(副网格颜色)、MinorGridType 属性(主网格线条类型)。
- **配置坐标轴上主网格的个数和密度：**需要配置主网格个数为固定个数时配置对应坐标轴的 MajorGridCount 为指定的值；如果需要自动配置主网格，则配置 MajorGridCount 为小于等于 0 的值，并修改 MajorGridCountPerPixel 和 MiniorGridCountPerPixel 属性配置主网格密度，前者的值应大于后者值的 2.5 倍。
- **配置坐标轴的标题：**配置对应坐标轴的 Title 属性，修改 TitleOrientation 和 TitlePosition 可以修改坐标轴标题的方向和位置。
- **指定坐标轴起始状态的范围：**配置对应坐标轴的 Maximum 属性为指定的最大值，配置对应坐标轴的 Minimum 属性为指定的最小值。在 EasyChartX 界面上点击鼠标右键，弹出功能菜单，点击 Specify Y Range 即弹出 Y 轴范围配置对话框，可手动配置 Y 轴坐标轴的范围。
- **绘图时指定坐标轴的范围：**配置对应坐标轴的 AutoScale 属性为 False，配置对应坐标轴的 Maximum 属性为指定的最大值，配置对应坐标轴的 Minimum 属性为指定的最小值。

- **指定坐标轴自动配置范围：**配置对应坐标轴的 AutoScale 属性为 True。
- **指定坐标轴在绘图时自动缩放到指定范围：**配置对应坐标轴的 InitWithScaleView 属性为 True，并配置 ViewMaximum 和 ViewMinimum 为需要缩放的范围终点和起点。
- **运行时缩放坐标轴到指定范围：**调用指定坐标轴的 Zoom(double start, double end)方法，start 和 end 分别为需要缩放的起始和终点。
- **取消坐标轴的缩放：**调用指定坐标轴的 ZoomReset(int resetTime)方法，resetTime 为回退缩放的次数，如果彻底取消缩放则使用该参数默认值即可。
- **每次绘图前自动取消上次绘图中使用的缩放：**配置对应坐标轴的 AutoZoomReset 属性为 True。
- **配置坐标轴网格标记的宽度：**修改对应坐标轴的 TickWidth 属性即可修改主网格标记的长度(X 轴)或宽度(Y 轴)，该字段为浮点类型，表示标记的长度或宽度占绘图区域长度或宽度的百分比。如果不显示主网格标记可以将该属性配置为 0。

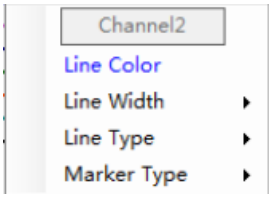
2.4.21.4 配置曲线的外观

EasyChartX 可以在设计时和运行时配置曲线的各项外观参数。

- 在编辑界面时可以在属性配置页面中选择 LineSeries 属性配置曲线参数。该属性都是集合类型，可以使用下标访问指定的曲线参数，集合中元素的个数不需要等于当前的曲线个数，待绘图时曲线个数增加时会自动适配到指定的配置。LineSeries 属性在编辑界面时的配置界面如下图所示。



- 在运行时可以左键点击图例中对应的曲线配置曲线参数。在图例中鼠标左键点击待修改的曲线将弹出曲线的外观配置菜单，该菜单的界面如下图所示，可以分别用来配置曲线的颜色、宽度、线型、标记类型。



- 在代码中可以使用 Series 属性配置曲线参数，该属性都是集合类型，可以使用下标访问指定的曲线参数，集合中元素的个数不需要等于当前的曲线个数，待绘图时曲线个数增加时会自动适配到指定的配置。例如在代码中修改某个曲线的颜色可以使用 easyChartX.Series[index].Color = xxx 实现，其中 index 为曲线的索引号。

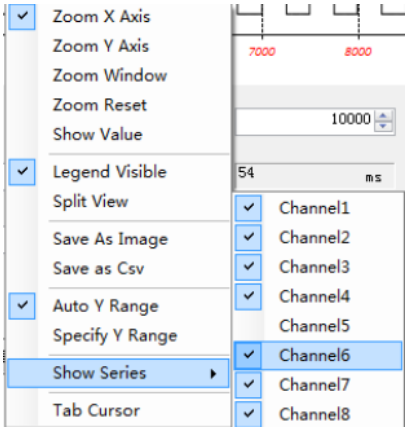
曲线的外观配置项目包括：

- Color：曲线的颜色。
- Marker：曲线上每个点的标记类型，在点数较少时可以用来凸出显示数据点，该值不是 None 时需要联带配置 Type 属性为 Line 才能生效。
- Type：曲线的类型，默认为 FastLine。
- Width：曲线的宽度，有 Thin、Middle、Thick 三个可选项。
- Name：曲线的名称，该名称会被显示到图例中。

2.4.21.5 在绘图区显示或隐藏某条曲线

在绘图区中显示或隐藏曲线的实现方式有两种：

- 运行时，在 EasyChartX 上点击鼠标右键弹出功能对话框，选择点击其中的“Show Series”子菜单即可看到当前显示的曲线，子菜单如下图所示，点击其中某个曲线即可修改该曲线的显示或隐藏。



- 在代码中通过配置 Series 属性中某个元素的 Visible 属性修改曲线的显示或隐藏某个线条，例如通过调用 easyChartX.Series[index].Visible = false 即可隐藏 index 索引的曲线。

2.4.21.6 配置界面未绘图时图例中显示的线条个数

EasyChartX 使用 SeriesCount 属性配置当前绘图曲线的个数。当曲线处于未绘图的状态是配置 SeriesCount 即可配置图例中显示的线条个数，在绘图状态时配置该属性值无效。

2.4.21.7 使用副坐标轴显示数据

副坐标轴无需手动设置是否显示，EasyChartX 会自动监测曲线的坐标轴配置，如果存在某个显示的曲线使用了副坐标轴，该坐标轴会自动显示；如果不存在任何显示的曲线使用副坐标轴，则副坐标轴会自动隐藏。曲线使用副坐标轴在界面编辑时和代码运行时都可以配置。目前的版本中暂不支持 X 轴配置使用副坐标轴。

- 界面编辑时：界面编辑页面，选择 EasyChartX，在属性窗体中边界 LineSeries 属性，在弹出的窗体中算则待修改的曲线，修改其 YPlotAxis 属性为 Secondary。
- 在代码中执行 easyChartX.Series[index].YPlotAxis = EasyChartXAxis.PlotAxis.Secondary 即可将索引号为 index 的曲线配置 Y 轴使用副坐标轴显示。

2.4.21.8 使用对数坐标轴

EasyChartX 支持在 X 轴和 Y 轴使用对数方式进行显示。该操作在界面编辑时和代码运行时都可以进行配置。

- 界面编辑页面，选择 EasyChartX，在属性配置窗体中配置 Axes 属性。在弹出的 Axes 属性配置对话框中选择待修改的坐标轴，修改 IsLogarithmic 属性为 True。
- 在代码中执行 easyChartX.AxisX.IsLogarithmic 属性为 true，修改 Y 轴时将 AxisX 替换为 AxisY。代码中执行时需保证当前绘图区不存在数据，否则切换后的曲线显示效果可能会出现扭曲的情况。

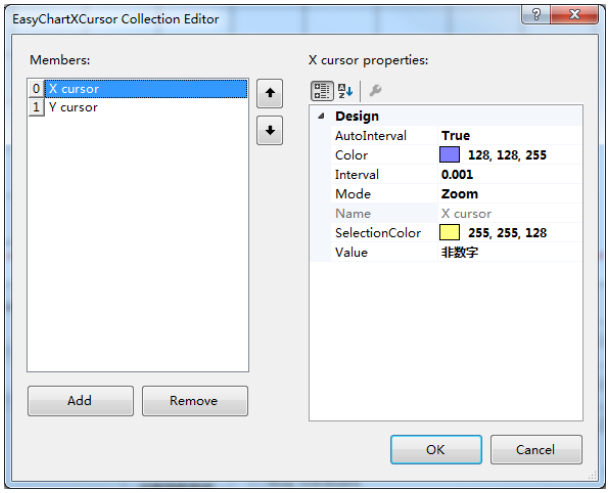
在 X 轴或 Y 轴使用对数坐标时对应的数据不能出现小于等于 0、double.NaN、double.Infinity 等非法值，否则在绘图时会导致绘图区出现红色的 X 错误。对于 Y 轴使用对数坐标轴且不确保数据全部合法的情况下可以使用 EasyChartX 的非法数据监测功能，使用方式参考 2.3.21.13 节。

2.4.21.9 配置游标的功能

EasyChartX 在 X 轴和 Y 轴分别有一个游标，坐标轴上的游标有两个功能，视图缩放和数据选择。两个游标的功能可以独立配置互不干扰。游标的功能和属性在界面编辑时、代

码控制、运行时都可以修改。

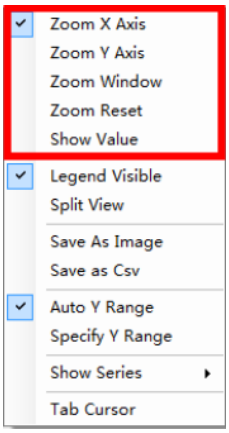
- 界面编辑界面选择 EasyChartX，在属性窗体中选择 Cursors 属性，弹出如下的对话框即可分别编辑每个游标的功能。



- 使用代码可以控制游标的功能和配置，调用 easyChartX.XCursor.Mode 即可修改游标的功能，两个游标之间互相组合的功能如下所示。在游标模式和数据标记模式下，使用 XCursor.Value 和 YCursor.Value 可以分别获取和配置游标的值。

		X 轴游标功能		
		Disable	Cursor	Zoom
Y 轴游标功能	Disable	游标不显示	X 轴游标	X 轴缩放
	Cursor	Y 轴游标	数据标记	X 轴缩放
	Zoom	Y 轴缩放	Y 轴缩放	窗体缩放

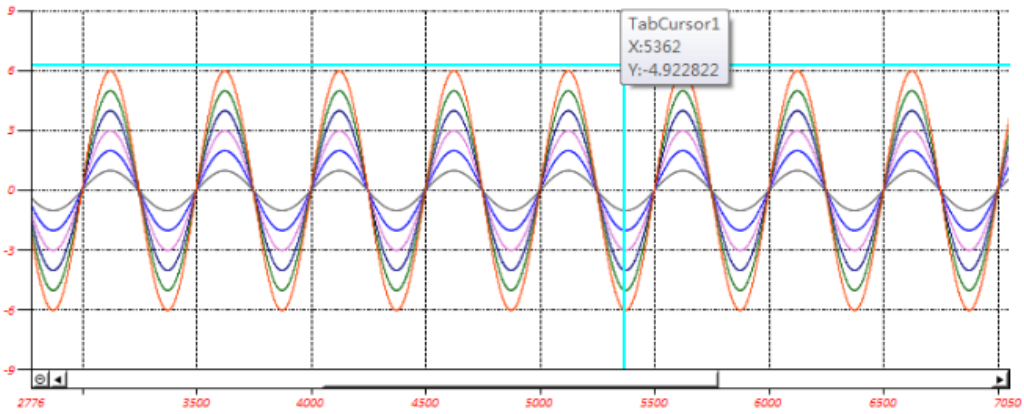
- 在运行时可以使用右键菜单选择游标的功能，右键菜单如下图所示，用于修改游标功能的选项如下图。某个菜单项前被选中即选中对应游标功能，任何子菜单项未选中则游标不使能。



2.4.21.10 使用 EasyChartX 的标签游标

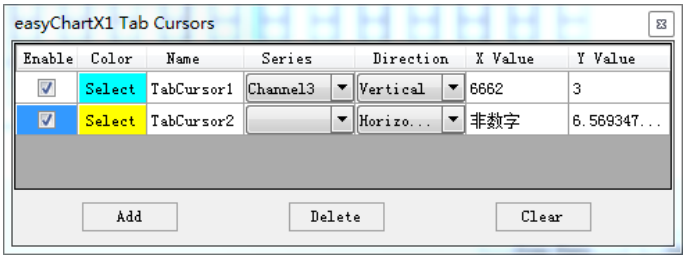
EasyChartX 提供标签游标用于跟踪指定点的数值变化或者在绘图区绘制一个直线。一

个竖向标签游标和横向标签游标在绘图区的显示效果如下图所示(淡蓝色的游标线)。



添加标签游标有两种方式。

- 运行时通过界面添加。在绘图区存在数据时点击鼠标右键，在弹出的功能菜单中点击最下方的 Tab Cursor 按钮，即可弹出标签游标信息对话框，该对话框如下图所示。通过 Add、Delete、Clear 按钮对标签游标执行新增、删除、清空等操作。同时 Enable、Color、Name、Series、Direction、XValue(Direction 为 Vertical)、YValue(Direction 为 Horizental)都可以在运行时进行编辑。



- 使用代码添加。标签游标在 EasyChartX 的 TabCursors 属性中维护，该属性为集合属性，可以包含多个 TabCursor。通过调用以下方法即可添加一个 TabCursor 到绘图区。

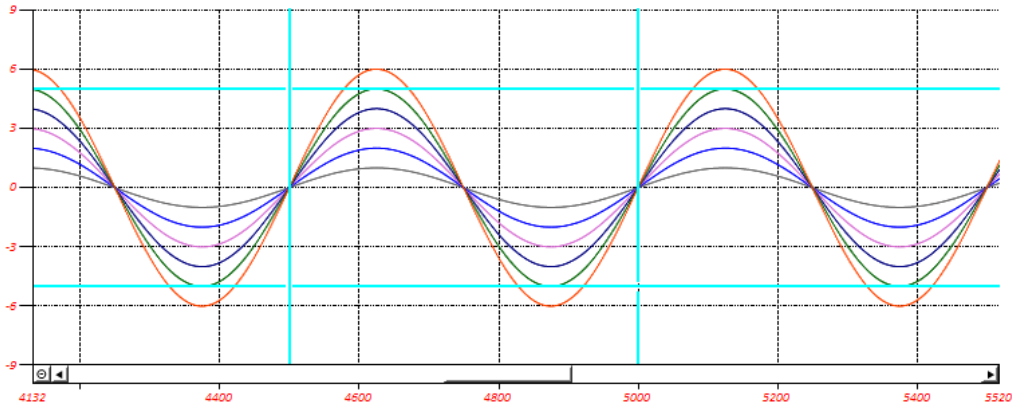
```
this.easyChartX1.TabCursors.Add(new TabCursor())
```

TabCursor 包含的属性定义如下。

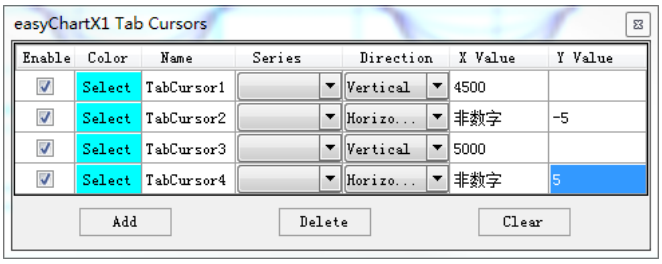
属性	
TabCursor.Color(Color)	游标颜色
SeriesIndex(int)	TabCursor 绑定到的曲线的索引号，小于 0 为未绑定。
Value(double)	游标当前对齐到的 X 轴的值
YValue(double)	当 Direction 为 Vertical 时记录游标对齐的 X 轴对应曲线的 Y 值，只读，只有 SeriesIndex 配置了有效值时可用；Direction 为 Horizental 时可读写。
Direction(TabCursorDirection)	标签游标的方向，可以为 Horizental 和 Vertical。当该值为 Horizental 时不能跟踪数据点，XValue 和 SeriesIndex 无效。
Name(string)	游标名称

Enabled(bool)	是否显示游标

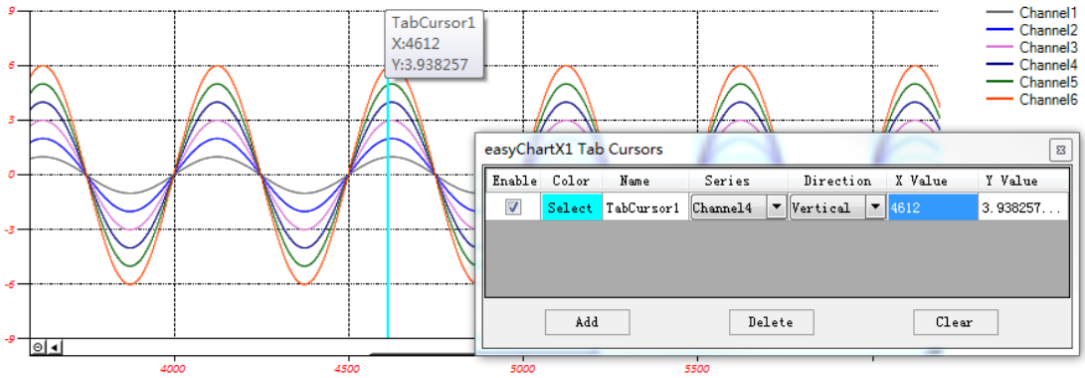
使用标签游标作为区域标记时需要修改 SeriesIndex 无效(在属性中配置属性值为-1，在 TabCursor 管理对话框中选择空白)。然后修改 XValue(Direction 为 Vertical 时)或者 YValue (Direction 为 Horizental 时)为指定的值，此时标签游标即可成为一个标记范围的工具，例如以下的标签游标即选中了 X 轴从 4500 到 500、Y 轴从-5 到 5 的范围。



此时标签游标的配置如下图。



使用标签游标跟踪指定点的值时需要配置 TabCursor 的 Direction 为 Vertical，配置 Series 属性为需要跟踪的点所在的曲线，此时修改 XValue 的值或者使用鼠标拖动 TabCursor 即可在 TabCursor 管理对话框上实时看到选择曲线在对应的 X 值时 Y 值的变化。同时将鼠标放到该 TabCursor 上也可以看到当前跟踪的 X 值和 Y 值，如下图所示。即使 EasyChartX 重新绘图该游标也会自动跟踪到最近的 X 值并获取其 Y 值。



2.4.21.11 为指定的点添加标记

如果需要对曲线上所有点做标记，可以将对应曲线的 Type 修改为 Line，然后配置 MarkerType 为预期的值即可。

如果需要对曲线上某个特殊的点进行标记可以调用 AddDataMarker 方法实现。该方法

```
AddDataMarker(IList<double> xValue, IList<double> yValue,
              Color markerColor,
              DataMarkerType markerType = DataMarkerType.Square,
              EasyChartXAxis.PlotAxis xAxis = EasyChartXAxis.PlotAxis.Primary,
              EasyChartXAxis.PlotAxis yAxis = EasyChartXAxis.PlotAxis.Primary)
```

的签名如下。

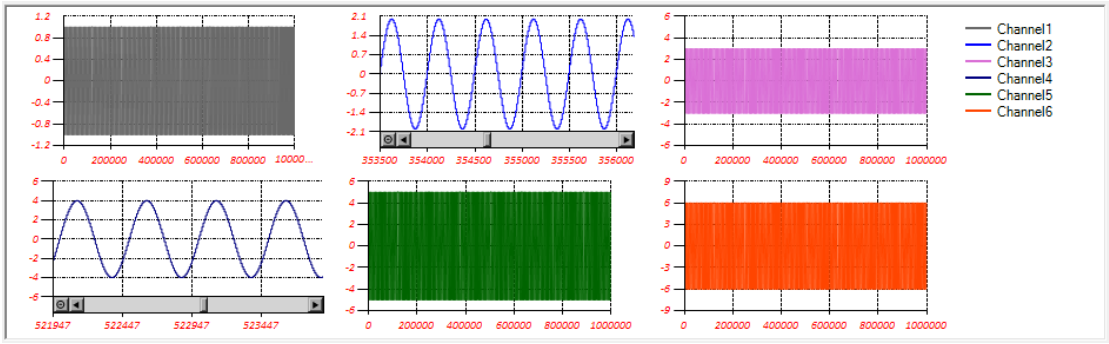
- xValue，yValue：需要标记的所有点的 X 坐标值和 Y 坐标值，集合类型，可以是任意实现了 IList<double>接口的对象，例如 double[]、List<double>等。
- markerColor：标记的颜色。
- markerType：标记的类型，可选值有 Square、Circle、Diamond、Triangle、Cross。
- xAxis、yAxis：标记的点绘制在哪个坐标轴上，默认值为主 X 坐标轴和主 Y 坐标轴。

Marker 的大小统一管理，需要在 Miscellaneous 的 MarkerSize 中进行配置，默认值为 7 个像素。

EasyChartX 的 Marker 在执行 Clear 方法或者重新调用 Plot 方法时会被清空。如果需要在未调用 Clear 方法或者重新绘图的情况下清空已标记的 Marker，可以调用 EasyChartX 的 ClearMarker 方法。

2.4.21.12 自定义分区视图的布局

配置 EasyChartX 的 IsSplitView 属性为 True 即可切换到分区视图，每个曲线占用独立的绘图区域，如下图所示。



默认情况下这些绘图区域会自动配置布局，如果自动配置的布局不符合预期，用户可以自行配置布局方式。分区视图的布局配置在 Miscellaneous 属性中，涉及到分区视图布局的参数包括：

- SplitViewAutoLayout：是否使用默认分区布局。配置为 True 时后面的所有属性配置无效；配置为 False 时使用下面的参数进行分区布局。
- DirectionChartCount：绘图区按照布局方向的最大个数。
- SplitLayoutDirection：分区视图的绘图区域按照曲线顺序的排布方向，可选的值有：
  - LeftToRight：按照曲线的顺序，绘图区域依次从左到右分布，一行的绘图区个数达到 DirectionChartCount 后自动换行。
  - TopToBottom：按照曲线的顺序，绘图区域依次从上到下分布，一列的绘图区个数达到 DirectionChartCount 后自动换列。
- SplitLayoutColumnInterval：分区视图相邻列之间的间隔，该字段可以配置为负值，需要注意该参数如果过小可能会导致绘图区域出现覆盖情况。
- SplitLayoutColumnInterval：分区视图相邻行之间的间隔，该字段可以配置为负值，需要注意该参数如果过小可能会导致绘图区域出现覆盖情况。

2.4.21.13 在绘图前检查 Y 轴的非法数据

EasyChartX 在 Miscellaneous 属性中提供了部分接口用于配置 EasyChartX 的 Y 轴非法数据检查，这些字段分别包括：

- CheckInfinity：检查数据中是否存在正无穷大或负无穷大的值。配置为 True 后 Y 轴数据中所有 PositiveInfinity 会被修改为 double.NaN(CheckNaN 为 False)或者 1E200(CheckNaN 为 True)；所有 NegtivePositiveInfinity 会被修改为 double.NaN(CheckNaN 为 False)或者-1E200(CheckNaN 为 True 且 CheckNegtiveOrZero 为 False)或者 1E-200(CheckNaN 为 True 且 CheckNegtiveOrZero 为 True)。同时非法的点在 Chart 上会被自动隐藏。
- CheckNaN：检查数据中是否存在 double.NaN 的点。配置为 True 后所有的 NaN 点会被修改为 1.5E-200，同时非法的点在 Chart 上会被自动隐藏。

- CheckNegtiveOrZero：检查数据中是否存在小于等于 0 的点，用于 Y 轴使用对数坐标的场景。配置为 True 后所有小于等 0 的点会被配置为 double.NaN(CheckNaN 为 False)或者 1E-200(CheckNaN 为 True)。同时非法的点在 Chart 上会被自动隐藏。

2.4.21.14 绘图时保留原来的曲线

EasyChartX 在默认情况下每次绘图前会清空已有的曲线，如果需要累积绘图(即每次绘图不清空原来的曲线)，可以配置 AutoClear 属性的值为 False。

2.4.21.15 绘图后取消上次绘图时的缩放

EasyChartX 默认情况下如果执行了缩放操作，下次绘图时会继续保持该缩放状态。如果需要每次绘图后取消缩放状态，可以配置对应坐标轴的 AutoZoomReset 属性为 True。

2.4.21.16 提升 EasyChartX 的绘图性能

EasyChartX 在 Miscellaneous 属性中提供了部分接口用于调整 EasyChartX 的绘图性能，这些字段分别包括：

- DataStorage：指定 EasyChartX 绘图时的数据保存选项。可选的值有
  - Clone：EasyChartX 每次绘图时都会拷贝数据到缓存中，会降低性能，但是可以保证数据被其他功能修改时不会影响当前数据更新时的失效问题。
  - NoClone：EasyChartX 每次绘图时不拷贝数据，性能较高，用于大数据量的数据绘制，但是需要用户保证该数据在有效的时间周期内不会被额外的线程更改。
- MaxSeriesPointCount：单条曲线在界面上最多绘制的 GDI 点数。该数值越高则绘制的图像越精细，性能更低；数值越小则绘制的图像越粗糙，性能越高。对于宽度较小的 Chart 可以适当调低该参数用于提升绘图性能。
- Fitting：曲线绘图拟合算法，可选的值有：
  - Range：区间内范围趋势拟合，绘图效果最好，性能较低。
  - None：抽点显示，绘图效果较差，性能最高，适用于曲线趋势变化较小的场景，除非对性能要求极高，否则不推荐使用。

2.4.21.17 隐藏右键功能菜单

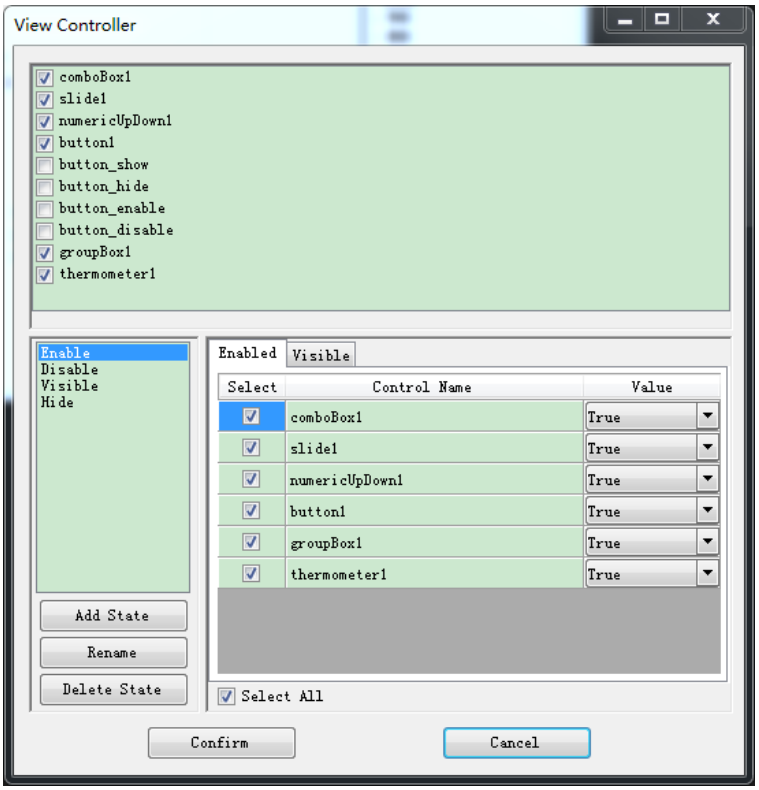
在部分场景下需要禁用 EasyChartX 的右键功能菜单，可以在 Miscellaneous 属性中配置 ShowFunctionMenu 字段为 False。

2.4.22 ViewController

ViewController 是一个用来操作界面联动的类，使用切换状态的方式根据当前状态调整界面控件的状态。目前支持操作控件的 Enabled(是否使能)和 Visible 属性(是否可见)。该控件的属性如下所示，其中除了 State 以外其他属性都是在设计时配置完成的。该控件仅在设计时可见，运行时会自动隐藏。

属性	
属性名称	**功能描述**
StateNames(string[])	所有状态名
State(string)	当前状态，不区分大小写
ControlInfos(string[])	各个状态的控制信息，无需手动修改

ViewController 的 StateNames 和 ControlInfos 通过 ViewController 控件的小三角配置。点击后的配置界面如下图所示，详细的使用方法参见范例：SeeSharpExamples/ Graphical User Interface/ViewControllerExample。



配置状态和联动信息的流程如下：

- 先在最上面的列表中选择需要联动的控件，这些控件会自动同步到下面的表格中
- 左下方添加、重命名或者删除状态。如需配置某个状态的控件属性，点击该状态
- 在右侧对需要操作的控件打勾，然后选择对应的Enabled和Visible的值。Enabled和Visible通过上面的tab页切换
- 配置结束后点击Confirm即可。

使用时可以用类似 viewController1.State = “idle”的方法，将 idle 中配置为不使能或不显示的控件灰化或隐藏。该属性是跨线程安全的，在非 UI 线程执行时无需调用额外的 Invoke。

2.4.23 StripChartX

StripChartX 是基于微软 MSChart 控件实现的记录仪控件，该控件是 StripChart 的升级版本。StripChartX 中加入了测试测量很多常用的特性、尽量简化了对外接口、提高了绘图效率，同时新增分区视图，可以将每条线独立显示到不同的绘图区中。StripChartX 的属性、事件和方法如下表所示：

属性	
属性名称(类型)	**功能描述**
LegendVisible(bool)	设定是否在 Chart 中显示图例，默认为 True。
BackColor(Color)	设定 StripChartX 外框颜色
ChartAreaBackColor(Color)	设定 StripChartX 绘图区颜色，该配置同时生效于两种视图
ForeColor(Color)	坐标轴 Label 和名称的字体颜色
Font(Font)	坐标轴 Label 的字体
LegendBackColor(Color)	图例的背景色
LegendFont(Font)	图例的字体
LegendForeColor(Color)	图例的字体颜色
SeriesCount(int)	配置未绘图时显示的线条个数、获取当前绘图中的线条个数。
GradientStyle(ChartGradientStyle)	背景色渐变类型
ScrollType(StripScrollType)	滚动类型。累积类型(积攒到最大点数再滚动)/滚动类型(初始即滚动)
DisplayPoints(int)	记录仪最大显示点数，超过该点数的数据被丢弃
XDataType(XAxisDataType)	X 轴数据类型：索引/时间戳/字符串
TimeStampFormat	时间戳格式，XDataType 为时间戳时有效
NextTimeStamp	下次绘图的首个时间戳，XDataType 为时间戳时有效
TimeInterval(TimsSpan)	时间间隔，XDataType 为时间戳时有效

StartIndex(int)		起始索引号，XDataType 为索引时有效
Direction(ScrollDirection)		滚动方向，从右到左/从左到右
Miscellaneous (MiscellaneousConfiguration)		杂项配置
	MarkerSize(int)	Marker 的大小，取值范围[3-21]
	ShowFunctionMenu(bool)	配置在 Chart 上右键点击，是否显示功能菜单
	MaxSeriesCount(int)	最大线条个数，默认 32，超过该值的数据在 Chart 不显示
	MaxSeriesPointCount(int)	单个线条上最多显示的点数，默认 4000，增大该值会提升数据显示的细腻程度，但是会降低性能。
	CheckNaN(bool)	是否校验数据中的 NaN 数据。如果不检查则 Nan 显示为空点；如果检查则 NaN 替换为 1.5E-200 并显示为空点。
	CheckNegtiveOrZero	是否校验非正数。如果不检查 Nan，则非正数替换为空点；如果检查 Nan，则非正数替换为 1E200 和 1E-200
	CheckInfinity	是否校验 Inf 数据。如果不检查 Nan，则 Inf 替换为空点；如果检查 Nan 则 Inf 替换为 1E200 和-1E200/1E-200。
	Fitting(StripChartX.FitType)	筛点拟合算法选择。大数据时使用 None 可改善绘图速度，但是可能会导致数据趋势的显示错误。
	SplitLayoutDirection (LayoutDirection)	分区视图的布局方向：从左到右或者从上到下
	SplitViewAutoLayout(bool)	配置分区视图是否自动布局，默认为 true。配置为 false 时 DirectionChartCount 会生效。
	DirectionChartCount(int)	布局方向的绘图区个数(SplitLayoutDirection 为 LeftToRight 时为一行的绘图区个数，TopToDown 时为一列的绘图区个数)，该属性在 SplitViewAutoLayout 为 False 时生效
	SplitLayoutColumnInterval(int)	分区视图每列绘图区之间的像素间隔，可以为负值
	SplitLayoutRowInterval(int)	分区视图每行绘图区之间的像素间隔，可以为负值
LineSeries(设计时属性名) Series(开发时属性名) (EasyChartXSeriesCollection)		设定每个线条的属性。该属性的配置个数可以超过当前线条数。新增线条按照对应索引进行配置。
	Name(string)	设定线条名称，该名称会被显示到图例中
	Color(Color)	线条颜色
	Visible(bool)	线条是否可见
	Width(LineWidth)	线宽
	XPlotAxis(PlotAxis)	使用哪个 X 轴，当前版本只可以使用主 X 轴。
	YPlotAxis(PlotAxis)	使用哪个 Y 轴，可以选择主 Y 轴和副 Y 轴
	Type(LineType)	线条类型
	Marker(MarkerType)	点标记类型。只有在 Type 为 Line 时生效
SplitPlotArea (EasyChartXPlotAreaCollection)		配置和获取分区视图时各个绘图区的属性
	BackColor(Color)	绘图区背景色
	AxisX(EasyChartXAxis)	主 X 坐标轴
	AxisX2(EasyChartXAxis)	副 X 坐标轴

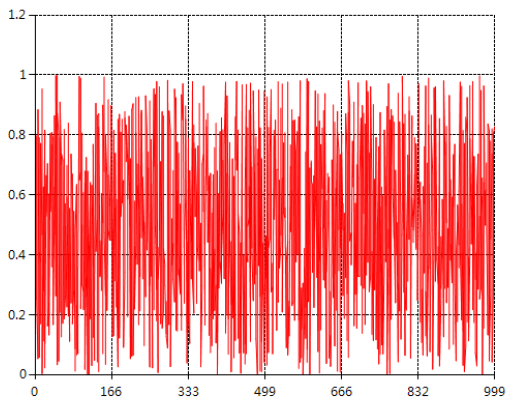
	AxisY(EasyChartXAxis)	主 Y 坐标轴
	AxisY2(EasyChartXAxis)	副 Y 坐标轴
	XCursor(EasyChartXCursor)	X 轴游标
	YCursor(EasyChartXCursor)	Y 轴游标
Axes(EasyChartXAxis[]) AxisX(EasyChartXAxis) AxisX2(EasyChartXAxis) AxisY(EasyChartXAxis) AxisY2(EasyChartXAxis)		坐标轴配置，可以配置 X 轴、副 X 轴、Y 轴、副 Y 轴的相关属性。该配置同时生效于两种视图。
	Name(string)	获取坐标轴名称
	Color(Color)	配置坐标轴颜色
	Title(string)	配置坐标轴标题
	TitleOrientation (AxisTextOrientation)	配置坐标轴标题显示方向
	TitlePosition (AxisTextPosition)	配置坐标轴标题显示位置
	LabelEnabled(bool)	配置是否显示坐标轴的值
	LabelFormat(string)	配置坐标轴下方标签的格式
	MajorGridEnabled(bool)	主网格使能
	MajorGridColor(Color)	主网格颜色
	MajorGridType(GridType)	主网格线条类型
	MinorGridEnabled(bool)	副网格使能
	MinorGridColor(Color)	副网格颜色
	MinorGridType(GridType)	副网格线条类型
	TickWidth(double)	配置坐标轴上标记线的宽度和 Chart 高/宽的比值
	AutoScale(bool)	坐标轴范围是否自动配置
	Maximum(double)	获取当前坐标轴最大值。未绘图时配置坐标轴最大值；绘图时且 AutoScale 为 false，可以配置坐标轴最大值。
	Minimum(double)	获取当前坐标轴最小值。未绘图时配置坐标轴最小值；绘图时且 AutoScale 为 false，可以配置坐标轴最小值。
	IsZoomed(bool)	获取当前坐标轴是否处于缩放状态
	AutoZoomReset(bool)	配置每次绘图时是否取消缩放。
	InitWithScaleView(bool)	配置绘图开始时是否默认使用缩放视图。
	ViewMaximum(double)	获取当前坐标轴缩放视图最大值，InitWithScaleView 为 true 时配置开始绘图时坐标轴缩放视图最大值
	ViewMinimum(double)	获取当前坐标轴缩放视图最小值，InitWithScaleView 为 true 时配置开始绘图时坐标轴缩放视图最小值
	IsLogarithmic(bool)	当前坐标轴是否使用指数显示，只支持 Y 轴。
	Zoom(double max, double min)	将当前坐标轴缩放到两个数值的范围内
	ZoomReset(int resetTimes)	回退坐标轴缩放操作
Cursors(EasyChartXCursor[]) XCursor(EasyChartXCursor) YCursor(EasyChartXCursor)		游标配置，可以配置 X 游标、Y 游标的相关属性。该配置同时生效于两种视图。

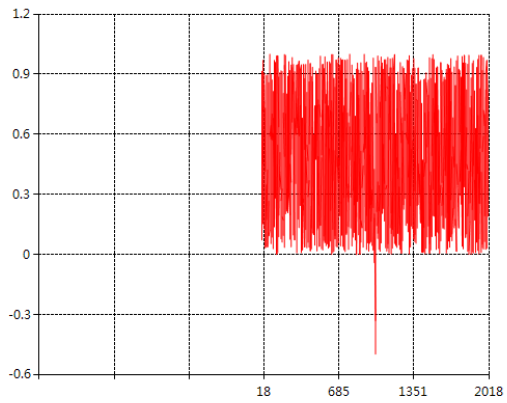
	Name(string)	获取游标名称
	Mode(CursorMode)	游标类型，包括非使能、游标和区域选择
	Color(Color)	游标颜色
	SelectionColor(Color)	游标区域选择时选择区的颜色
	AutoInterval(bool)	是否自动配置选择间隔
	Interval(double)	AutoInterval 时配置游标选择间隔
	Value(double)	游标模式时获取游标当前所在的位置
TabCursors(TabCursorCollection)		标签游标的集合
	CursorValueFormat(string)	鼠标在标签游标上时，显示数值的格式
	void Add(TabCursor item)	添加标签游标
	void Clear()	清空标签游标
	TabCursor(TabCursors 中的元素)	标签游标类
	TabCursor.Color(Color)	游标颜色
	SeriesIndex(int)	TabCursor 绑定到的曲线的索引号，小于 0 为未绑定。
	TabCursor.Value(double)	游标当前对齐到的 X 轴的值
	TabCursor.YValue(double)	游标对齐的 X 轴对应曲线的 Y 值。只有 SeriesIndex 有效时可用
	TabCursor.Name(string)	游标名称
	TabCursor.Enabled(bool)	是否显示游标
自定义事件		
AxisViewChanged		坐标轴范围变化时触发该事件
	EasyChartXViewEventArgs	事件参数： Axis：触发事件的坐标轴 ParentChart：触发事件的 EasyChartX 实例 IsScaleViewChanged：是否是缩放视图变更触发的事件 IsRaisedByMouseEvent：是否是鼠标操作触发的事件
CursorPositionChanged		用户使用游标选择对齐到某条线后触发该事件
	EasyChartXCursorEventArgs	事件参数： Cursor：触发事件的游标 ParentChart：触发事件的 EasyChartX 实例 SeriesIndex：触发 Cursor 事件的线条索引 IsRaisedByMouseEvent：是否鼠标操作触发
BeforePlot		在绘图区绘制曲线前触发该事件
	EasyChartXPlotEventArgs	事件参数： IsClear：该事件是否由 Clear 触发 ParentChart：触发该事件的 EasyChartX 句柄
AfterPlot		在绘图区绘制曲线后触发该事件
	EasyChartXPlotEventArgs	IsClear：该事件是否由 Clear 触发 ParentChart：触发该事件的 EasyChartX 句柄
方法		
Plot<TDataType>(…)		记录单通道或多通道的多个样点。当记录仪中的样点个数超过 DisplayPoints 属性值时，前面的点会被丢弃。 Plot 方法的泛型重载支持：double/float/int/uint/short/

	ushort/byte 数据。
PlotSingle<TDataType>(…)	记录单通道或多通道的一个样点。当记录仪中的样点个数超过 DisplayPoints 属性值时，前面的点会被丢弃。PlotSingle 方法的泛型重载支持：double/float/int/uint/short/ushort/byte 数据。
SaveAsCsv(string filePath)	保存当前图中的数据到 Csv 文件，有路径入参和无参两个重载
SaveAsImage(string filePath)	保存当前图表到 png 文件，有路径入参和无参两个重载
ZoomReset(double min, double max)	停止所有坐标轴的缩放
Clear()	清除当前的所有绘图

在 Windows 窗体上放置 StripChartX 控件后，可以选中该 StripChartX 控件后在属性表格中对上述属性进行设置。StripChartX 支持单通道或多通道的连续或离散波形记录显示，波形的数值范围应该在[-2.5E28, 2.5E28]，超出该范围会导致 Chart 抛出运行时异常。

StripChartX 有两种滚动模式：累积模式和滚动模式。累积模式的点数一直累积，直到超过 DisplayPoints 的值时开始滚动；滚动模式从开始的一侧开始滚动，点数少于 DisplayPoints 的配置时，部分绘图区显示空白。两种模式的显示效果分别如下图所示：





2.5 SeeSharpTools.JY.Database 类库

JY.Database 包含 1 个 DbOperation 类，其中封装了对常见数据库的通用操作，支持的数据库类型 Microsoft SQL Server，Microsoft Access，MySql，Oracle，OleDb，ODBC 等，用户无需关注数据库的实际差异，只需要提供对应数据库的连接字符串以及数据库类型，然后使用通用的 SQL 语句就可以对数据库进行访问。类中的方法如下表所示：

方法名	功能概要
DbOperation 类	

ExecuteNonQuery	一般用于 UPDATE、INSERT 和 DELETE 语句是否操作成功。唯一的返回值是受到命令影响的行数，如果影响行数为 0 的话，则证明操作是不成功的，大于 0 则证明操作成功
ExecuteScalar	执行查询并返回查询所返回一个结果或者一列，如给定表中的记录个数或者当前服务器时间。返回值为一个对象（object），可以根据需要把对象强制转换为合适的类型。
ExecuteDataReader	DataReader（数据读取器）是从数据源中选择某些数据最简单高效的办法，返回的 DatReader 对象提供了“游标”形式的读取方法，当从结果中读取一行完毕后“游标”会继续读取到下一行，每次调用都会返回一行数据。
ExecuteDataTable	执行一个查询命令，返回的 DataTable（数据表）对象非常类似于实际物理的数据库表，它由一组特定属性的列组成，可能包含 0 行或者多行数据。数据表也可以定义主键（可以是一列或者多列），列上也可以包含约束。

2.6 SeeSharpTools.JY.Localization 类库

JY.Localization 包含一个 Localization 类，主要用于对 WinForms 程序实现本地化，在项目中需要结合资源文件来进行使用。类中包含的方法如下：

方法名	功能概要
Localization 类	
SetLang	读取对应语言的资源文件中信息，对指定窗体的界面实现本地化操作。

2.7 SeeSharpTools.JY.DSP.FilterMCR 类库

JY.DSP.Fundamental 包含 3 个静态类：IIRFilter, FIRFilter 和 JYSpectrum，分别提供 IIR 滤波器，FIR 滤波器，频谱计算功能，方法如下表所示：

方法名	功能概要
IIRFilter 类/FIR 类	
Initialize	预加载类需要调用的函数，加载完成后，首次调用时间会大大缩短。
ProcessLowpass	低通滤波器。
ProcessHighpass	高通滤波器。
ProcessBandpass	带通滤波器。
ProcessBandstop	带阻滤波器。
JYSpectrum 类	
FFTSpectrum	计算输入时域信号的频谱，可设定输出频谱的单位；

注意事项：

1. 项目属性->调试->启用调试器请取消勾选《启用 Visual Studio 承载进程》。
2. 该组件执行时依赖 MCR 2017a 组件，否则会抛出引擎加载失败的异常。如果当前计算机未安装 MCR 可以在以下路径中安装 MCR 2017a 和 2017a 版本的更新包： MCR installer、MCR Update Package。
3. 当前组件运行时依赖 DSPMatlab.dll 和 MWArray.dll，一般情况下编译时 DSPMatlab.dll 和 MWArray.dll 会被自动编译到执行目录，执行目录在 Debug(Release)模式时默认为工程的根目录/bin/Debug(Release)文件夹。在某些情况下可能不会拷贝 DSPMatlab.dll 和 MWArray.dll 到执行目录，这会导致应用抛出组件加载失败的异常，出现该异常时手动拷贝 SeeSharpTools 安装目录/bin 文件夹下的这两个文件到执行目录即可解决该异常。

2.8 SeeSharpTools.JY.DSP.SoundVibration 类库

JY.DSP.SoundVibration 包含一个 HarmonicAnalyzer 静态类，主要用于对声音和振动信号的分析和处理。类中包含的方法如下：

方法名	功能概要
HarmonicAnalyzer 类	
ToneAnalysis	计算输入信号的基波和所有谐波频率和幅度，以及总谐波失真（THD）。该方法有两个重载，分别用于计算全局的谐波参数和各阶谐波强度。

ToneAnalysisResult 类是 ToneAnalysis 方法中一个重载的返回值类型，该类型用于保存波形指标计算结果，各属性的描述如下所示。

属性名	功能概要
ToneAnalysisResult 类	
THD	谐波失真
THDPlusN	总谐波失真加噪声
SINAD	交流信号信噪比
SNR	信噪比
NoiseFloor	噪声基底
ENOB	有效位数

2.9 SeeSharpTools.JY.DSP.Utility 类库

JY.DSP.Utility 包含 8 个静态类： Phase、Synchronizer、ToneAnalysis、

SystemNoiseCalculation、PeakValleyAnalysis、SignalProcess、MedianFilter 和 ToneAnalyzer 类。其中 PeakSprectrum 类用于检测信号的峰值功率和所在频率点；Phase 类用于检测信号的相位信息；Synchronizer 类用于对多通道数据的相位进行补偿；ToneAnalysis 计算波形的谐波/信噪比等指标；PeakValleyAnalysis 用于检测波形的波峰和波谷；SignalProcess 类用于波形的峰值侦测、过零点侦测以及上下限侦测；MedianFilter 是一个中值滤波算法，可以用来过滤波形中的噪声；ToneAnalyzer 类用于计算单频信号的频率、幅值、相位信息。各个类中包含的方法如下：

方法名	功能概要
Phase 类	
CalPhaseShift	计算波形之间的相位差
Synchronizer 类	
Sync	对多通道数据的相位进行补偿
SystemNoiseCalculation 类	
CalculateSystemNoise	计算波形在指定带宽内噪声的 RMS
PeakValleyAnalysis 类	
FindPeaks	检测信号的波峰
FindValleys	检测信号的波谷

SignalProcessing 类用于波形的阈值和 AC/DC 部分的 RMS 检测。

SignalProcessing 类	
CheckThreshold	搜寻波形中超过阈值(threshold)的波峰值/波谷值。
CheckCrossZeroPoints	搜寻波形中爬升超过/下降低于零点的值。返回索引和值的字典
CheckInRange	检测波形中超过范围的值。返回值为长度等于波形长度的 List，依照索引保存结果：1 为超过上限，-1 为低于下限，0 为在 range 范围内。
EstimateACDC	计算出信号的 AC 成分和 DC 成分

SignalOperation 类用于操作信号。

SignalOperation 类		
Filtering		信号滤波类
	FIRFilter	使用 FIR 系数指定的直接型 FIR 滤波器，对输入序列进行滤波。
	IIRFilter	通过反向系数和前向系数指定的直接型 IIR 滤波器，对输入序列 X 进行滤波。
	MedianFilter	对输入序列进行中值滤波。
	ZeroPhaseFilter	使用零相位滤波器对输入序列进行滤波，未进行边界延拓。
Analog		模拟信号操作类
	AutoCorrelation	计算一维输入序列的自相关。
	Convolution	计算两个一维输入序列的卷积，使用线性卷积的 direct 方法。
	Invert	对数组中的每个元素求相反数。
	CrossCorrelation	计算两个一维输入序列的互相关，使用线性卷积的 direct 方法计算。

	ZeroPadder	使输入序列输入数组的大小调整为下一个有效的 2 的整数次幂，序列中新增的尾部元素设为零。
	UnwrapPhase	删除绝对值超过pi或180的不连续部分，展开相位数组。
	Scale	删除输入信号X的偏移量并缩放结果，使输出序列在[－1:1]的区间。
	Normalize	依据统计分布(μ,s)使输入向量归一化（μ是均值，s是标准差），得到统计分布为(0,1)的归一化向量。
	UpSample	通过升采样因子在输入序列中插入零。
	Downsample	通过降采样因子和平均布尔控件，对输入序列进行降采样。
	Detrend	对输入序列进行去趋势。
	WeightMovingAverage	对输入序列进行权重平均以平滑波动，类似于低通滤波。
	ExponentialAverage	对输入序列进行指数平均以平滑波动，类似于低通滤波。
	Resampling	对均匀输入序列使用新的采样率进行重采样，起始索引位置不变。
Digital		数字信号操作类
	GetDigitalSize	返回数字输入中包含的采样和信号的数量。
	DigitalToBool	将数字信号转换为二维布尔数组。
	BoolToDigital	将二维布尔数组转换为数字信号。
	DigitalToBinary	将数字信号转换为一维无符号整型数组。
	BinaryToDigital	将一维无符号整型数组转换为数字信号。
	Shift	对指定数字信号进行移位操作。
	Invert	对数字输入中的数字数据取反，使 0 变为 1（或 1 变为 0）。
	GetSubset	返回数字输入的子集，用于在图形上逐个绘制或按组绘制提取的信号。
	AppendSignals	合并两个数字信号。
	AppendSamples	添加数字波形 src2 的所有采样至数字波形 src1 之后。
AnalogToDigital		将模拟信号转换为数字信号。
DigitalToAnalog		将数字信号转换为模拟信号。

MedianFilter 类提供一种非线性数字滤波，可以用于滤除信号中的噪声。

MedianFilter 类	
Process	通过移动窗体来计算每个窗体块的中值。

ToneAnalyzer 类用于计算单频信号的频率、幅值、相位信息。

ToneAnalyzer 类	
SingleToneAnalysis	计算单频信号的参数，返回 ToneInfo 类型的波形信息。
ToneInfo 类	
Frequency(double)	波形频率
Amplitude(double)	波形幅值
Phase(double)	波形相位

注意事项：

1. 该组件执行时依赖 MCR 2017a 组件，否则会抛出引擎加载失败的异常。如果当前计算机未安装 MCR 可以在以下路径中安装 MCR 2017a 和 2017a 版本的更新包： MCR

installer、MCR Update Package。

2. 当前组件运行时依赖 FindPeaksAndValleysMatlab.dll 和 MWArray.dll，一般情况下编译时 FindPeaksAndValleysMatlab.dll 和 MWArray.dll 会被自动编译到执行目录，执行目录在 Debug(Release)模式时默认为工程的根目录/bin/Debug(Release)文件夹。在某些情况下可能不会拷贝 FindPeaksAndValleysMatlab.dll 和 MWArray.dll 到执行目录，这会导致应用抛出组件加载失败的异常，出现该异常时手动拷贝 SeeSharpTools 安装目录/bin 文件夹下的这两个文件到执行目录即可解决该异常。

2.10 SeeSharpTools.JY.Graph3D 类库

JY.Graph3D 为基于 ILNumerics 类库(1.4 版本)以及 ILNumerics.Drawing(0.9 版本)实现的自定义控件,开源授权符合 LGPL 定义

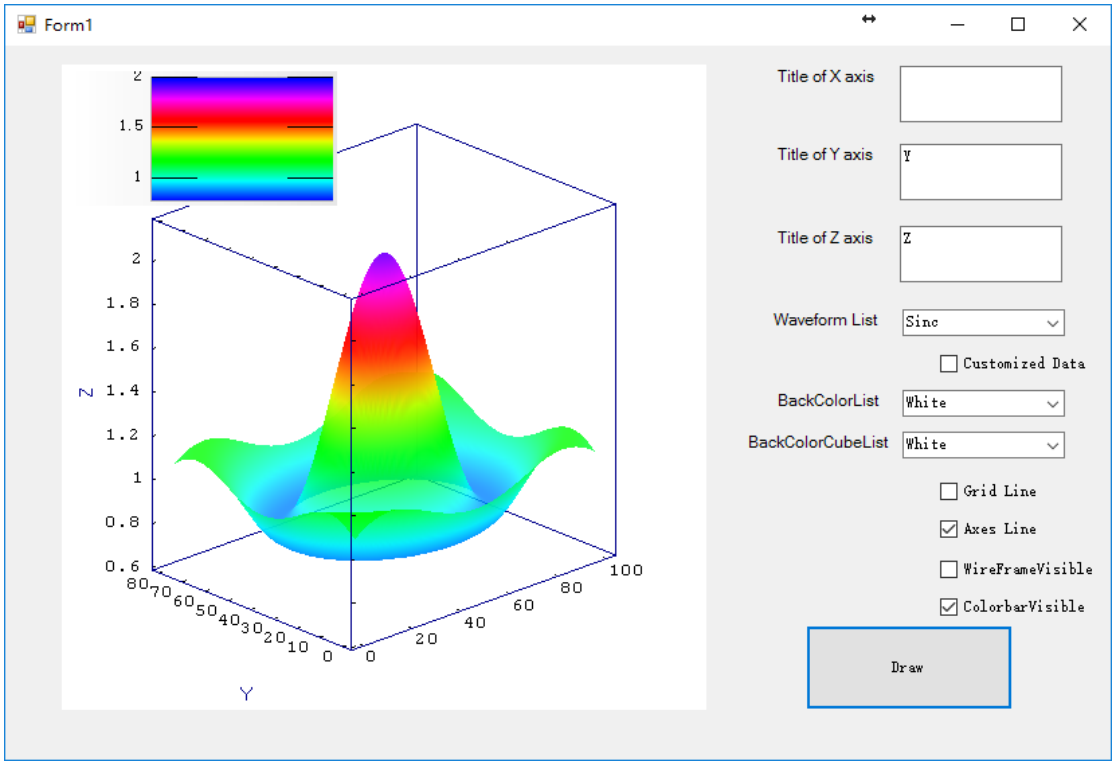
类库中引用了量测常用的 Intensity Graph 以及 Surface Graph 等两种图表控件

2.10.1 Surface Graph 类

Surface Graph 包含的属性以及方法如下表所示

属性	
属性名称	功能描述
AxesLineVisible	设定是否在 SurfaceGraph 中显示轴线标注,默认为 True。
BackgroundColor	设定 SurfaceGraph 中绘图区外的背景颜色,默认为 White
CubeColor	设定 SurfaceGraph 中绘图区颜色，默认为 White
CubeGridVisible	设定是否在 SurfaceGraph 中显示方网格线，默认为 false。
WireframeVisible	设定是否在 SurfaceGraph 上显示网格，默认为 false。
XAxisTitle	设定 SurfaceGraph 中 X 轴的标注文字，默认为 X。
YAxisTItle	设定 SurfaceGraph 中 Y 轴的标注文字，默认为 Y。
ZAxisTitle	设定 SurfaceGraph 中 Z 轴的标注文字，默认为 Z。
BackColorOfColorBar	设定 SurfaceGraph 中 Colorbar 的背景颜色,默认为 White
BorderStyleOfColorBar	设定 SurfaceGraph 中 Colorbar 的线条类型,默认为 FixedSingle
HeightofColorBar	设定 SurfaceGraph 中 Colorbar 的高度,默认为 120
Position	设定 SurfaceGraph 中 Colorbar 距离左上方的位置点,默认为 (5,5)
VisibleColorBar	设定是否在 SurfaceGraph 中显示 ColorBar，默认为 True。
WidthofColorBar	设定 SurfaceGraph 中 Colorbar 的宽度,默认为 120
方法	
方法名称	功能描述
Plot (double[,])	绘制图表(给予 2Ddouble 数组)
Plot (stdWfm)	绘制图表(使用内嵌的标准数组,Sync 以及 Waterfall)
Refresh()	重新绘制图表
ColorBarUpdate()	更新 ColorBar

图形绘制完成后,鼠标左键单击可放大检视,右键单击可缩小检视,双击恢复默认大小,鼠标拖曳可改变视角

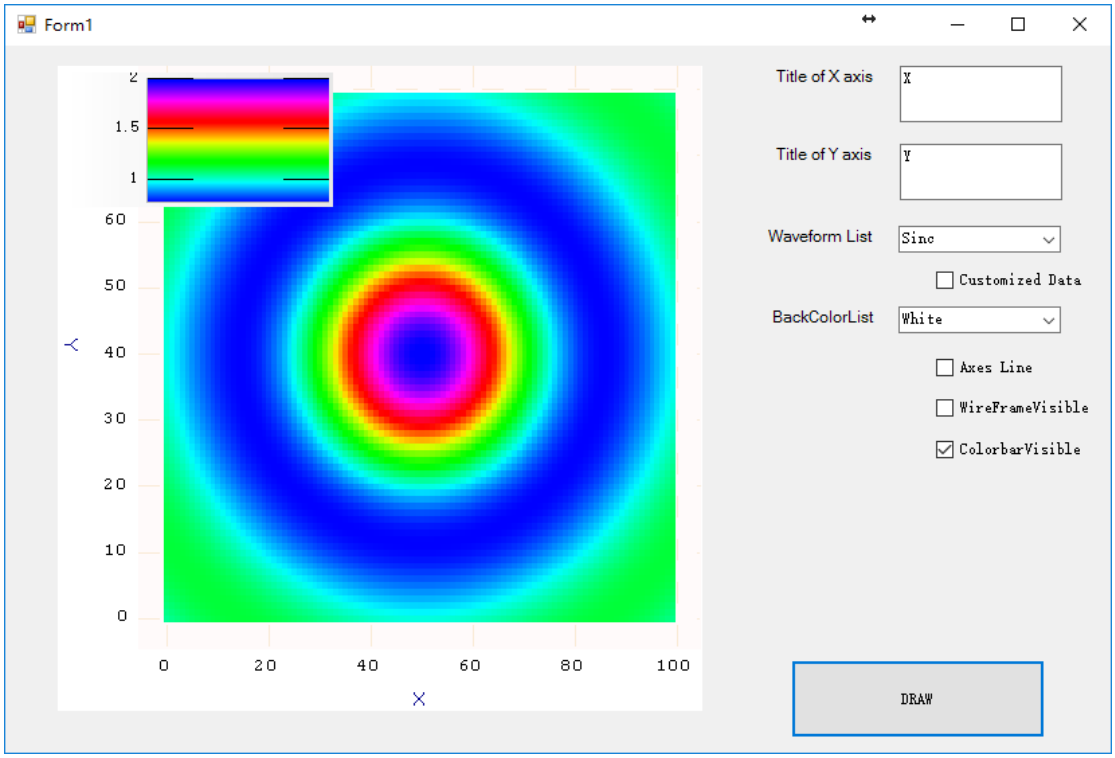


2.10.2 Intensity Graph 类

Intensity Graph 包含的属性以及方法如下表所示

属性	
属性名称	**功能描述**
AxesLineVisible	设定是否在 IntensityGraph 中显示轴线标注,默认为 True。
BackgroundColor	设定 IntensityGraph 中绘图区外的背景颜色，默认为 White
WireframeVisible	设定是否在 IntensityGraph 上显示网格，默认为 false。
XAxisTitle	设定 IntensityGraph 中 X 轴的标注文字，默认为 X。
YAxisTItle	设定 IntensityGraph 中 Y 轴的标注文字，默认为 Y。
BackColorOfColorBar	设定 IntensityGraph 中 Colorbar 的背景颜色,默认为 White
BorderStyleOfColorBar	设定 IntensityGraph 中 Colorbar 的线条类型,默认为 FixedSingle
HeightofColorBar	设定 IntensityGraph 中 Colorbar 的高度,默认为 120
Position	设定 IntensityGraph 中 Colorbar 距离左上方的位置点,默认为(5,5)
VisibleColorBar	设定是否在 IntensityGraph 中显示 ColorBar，默认为 True。
WidthofColorBar	设定 IntensityGraph 中 Colorbar 的宽度,默认为 120
方法	
方法名称	**功能描述**
Plot(double[,])	绘制图表(给予 2Ddouble 数组)
Plot(stdWfm)	绘制图表(使用内嵌的标准数组,Sync 以及 Waterfall)
Refresh()	重新绘制图表
ColorBarUpdate()	更新 ColorBar

图形绘制完成后,鼠标左键单击可放大检视,右键单击可缩小检视,双击恢复默认大小,鼠标拖曳可改变视角



2.11 SeeSharpTools.JY.File 类库

2.11.1 AnalogWaveformFile 类

JY.File 类库中包含了一个 AnalogWaveformFile 类，提供波形的存储和读取功能，公共属性和方法如下图所示。

属性	
属性名称	**功能描述**
FileFormat	文件格式，枚举，Stream/Multi-Record。
DataType	数据格式，枚举，Int16/Int32/Float/Double。
ByteOrder	字节序，枚举，Little Endian/Big Endian。
ArchiveInformation	归档信息，见”ArchiveInformation”类说明。
NumberOfChannels	通道数。
SampleRate	采样率，S/s
DataLength	数据长度，以 Sample 为单位
Channels	各通道采样信息，包括通道名称、采集量程、换算因子、物理单位名称和备注，List<ChannelInfo>列表，见” ChannelInfo”类说明。
DataStartTime	数据起始点时间。
TimeLabels	时间戳标签序列，可定义一组标签来标识指定数据采样点所在的时间，当文件格式为 Stream 时可对于长时间连续数据的时间此证，当文件格式为 Multi-Record 时用于标识个 Record 的起始点时间信息，见” Timelabel”类说明。
CustomLabels	自定义标签序列，可定义一组标签来保存备注信息，见” CustomLabel”类说明。
EnableWriteBuffering	是否开启写操作缓存。
EnableFastStream	是否开启高速读写。
方法	
Open	创建新文件或打开文件。
Close	关闭文件。
Write	写入数据，多态方法，支持多种数据类型(short,int,float,double)。
Read	读取数据，多态方法，支持单通道、多通道，以及多种数据类型(short,int,float,double)。
SetFilePosition	读取文件位置，以 Sample 为单位。
GetFilePositon	获取当前文件读取位置，或已写入的数据长度，以 Sample 为单位。
AddTimeLabel	添加时间标签，多态方法。
GetTimeLabel	获取指定时间标签信息。
AddCustomLabel	添加自定义备注，支持 Int/Double/String。
GetCustomLabel	获取指定自定义标签信息。

ArchiveInformation 类定义文件归档信息。

属性	
属性名称	功能描述
FileVersion	只读，WaveformFile 的版本
Author	可用于描述数据来源
DataLabel	可用于数据的归档标识
DataGroupID	可用于文件的关联标识
FileCreatedDate	只读，文件建立的时间
GeographicLocation	可用于描述地理位置信息
Description	可用于描述数据备注

ChannelInfo 类定义通道信息。

属性	
属性名称	功能描述
Name	通道名称
RangeHigh	通道量程上限
RangeLow	通道量程下限
Offset	a of "a + b * x"
Scale	b of "a + b * x"
Unit	单位
Description	备注

TimeLabel 类定义时间标签信息。

属性	
属性名称	功能描述
Name	标签名。
Postion	样点位置，表示相对于起始样点的样点个数。
Time	样点时间。
Description	备注。

CustomLabel 类定义自定义标签信息，并提供获取和修改自定义标签值的方法。

属性	
属性名称	功能描述
Name	标签名。
DataType	标签数据格式，枚举，Int16/Int32/Double/String。
方法	
SetValue	设置标签值。
GetValue	获取标签值。

2.11.2 BinHandler 类

BinHandler 类实现从 bin 文件读取和写入数据的功能，目前支持读写 double 或 string 类型的一维或二维数据。

方法	
方法名称	功能描述
ReadData	从 bin 文件中读取数据到二维 string 数组
ReadDoubleData	从 bin 文件中读取数据到二维 double 数组
WriteData	将一维或二维的 string 或 double 数组写入 bin 文件。可以配置当文件存在时追加还是覆盖。

2.11.3 CsvHandler 类

CsvHandler 是静态类，实现从 csv 文件读取和写入数据的功能，目前支持读写 string、double、int、uint、short、ushort 类型的一维或二维数据。

方法	
方法名称	功能描述
Read<TDataType>	从 csv 文件中读取数据到一维或二维数组，支持的数据类型有：double/float/int/uint/short/ushort/byte。
Write<TDataType>	向 csv 文件中写入一维或二维数组，支持所有支持 ToString 方法的类型，确定支持的数据类型有 double/float/int/uint/short/ushort/byte。

CsvHandler 的 Read 和 Write 方法为泛型方法。

Read 方法支持 double/float/ int/uint/short/ushort/byte 类型的数据，需要手动指定泛型的类型，例如读取 double 类型时的代码:`Read<double>(filepath, …)`。Read 方法有多个重载，分别支持读取指定行/从某行起始的多行/从某行起始的所有行、指定列/多个指定的列，支持包含路径的入参和不包含文件的入参(不包含路径入参时会弹出 winform 的文件选择窗口)，同时支持配置读取的编码格式(如果不配置则默认使用系统的默认编码格式，大多系统为 unicode)。

Write 方法理论上支持所有实现 ToString 方法的类型，确定支持 double/float/ int/uint/ short/ushort/byte 类型的数据。Write 方法支持写入一维和二位数组，可以配置写入的编码格式(如果不配置则默认使用系统的默认编码格式，大多系统为 unicode)，支持包含路径的入参和不包含文件的入参(不包含路径入参时会弹出 winform 的文件选择窗口)。

注：CsvHandler 在 1.4.3 及以前版本的接口都被标记为废弃，后期不再维护，不推荐使用。基于原有接口的应用仍可以正常运行。

2.11.4 IniHandler 类

IniHandler 类是静态类，实现对 ini 文件的操作。有 6 个调用接口，前四个主要用于整个 ini 文件的读写改操作，后两个方法适用于单个键值对的读写操作。每个方法的说明如下所示：

方法	
方法名称	**功能描述**
ReadIniFile	从 ini 文件中读取数据，返回 IniData 数据结构
WriteIniFile	将 IniData 数据写入文件
MergeIniData	将两个 IniData 数据合并，写入第一 IniData
MergeToFile	将 IniData 中的数据合并到某个文件的 IniData，并写入原文件
Read	读取 Ini 文件中所有的 Section 名称；某个 Section 下所有的 Key；某个 Section 下某个 Key 对应的 Value
Write	向 Ini 文件中的某个 Section 添加或者键值对或者修改某个 Key 对应的值

IniHandler 中涉及到三个数据类型，使用说明如下：

数据类型		
类		**功能描述**
IniData		读取完整文件后保存 ini 文件数据结构的类
	Sections	所有字段数据的集合，可以使用 foreach 遍历
	Global	所有全局键值对的集合
	Sections.AddSection(string sectionName)	添加空白字段
	Sections.AddSection(SectionData sectionData)	添加已有字段
	Sections.ContainsSection(string sectionName)	是否包含某个名称的字段
	Sections.RemoveSection(string sectionName)	删除某个字段
	Sections[string sectionName]	获取某个名称的 SectionData
Section		保存单个字段数据的类
	SectionName	获取或配置字段的名称
	Keys	保存键值对数据的集合，可使用 foreach 遍历
	Keys.AddKey(string key, string value)	添加一个键值对
	Keys.RemoveKey(string key)	删除一个键值对
	Keys.ContainsKey(string key)	是否包含某个键值对
	Keys[string keyName]	获取该键对应的值，可以使用 Key[key]=value 修改某个键值对的名称
KeyData		保存单个键值对数据的类
	KeyName	键值对的键
	Value	键值对的值

2.12 SeeSharpTools.JY.Report 类库

JY.DSP.Report 包含 2 个类：WordReport，ExcelReport 和 Logger，分别提供 Word 报表创建，Excel 报表生成和日志功能：

2.12.1 WordReport 类

WordReport 类	
属性名	功能概要
DefaultFont	默认配置的 Word 文字格式，包含字型、颜色、粗体、斜体、底线
方法名	功能概要
Close	关闭 Word 引擎
SaveAs	另存档案
Show	显示 Word 程序
Hide	隐藏 Word 程序在背景执行
WriteTextToDoc	写入文字
WriteTableToDoc	写入 Table 表格
InsertGraph	写入图表
InsertPicture	写入图片
ExecuteMacroFromScript	运行宏

2.12.2 ExcelReport 类

ExcelReport 类	
属性名	功能概要
DefaultFont	默认配置的 Excel 文字格式， 包含字型、颜色、粗体、斜体、底线
RowCount	获取 Excel 报表的行数
ColumnCount	获取 Excel 报表的列数
方法名	功能概要
Hide	隐藏 Excel 程序在背景执行
Show	显示 Excel 程序
SaveAs	另存档案
Close	关闭 Excel 引擎
RCToString	将 Excel 中行列数字转换成行列字符串

StringToRC	将 Excel 中行列字符串转换成行列数字
LetterToInt	字符串转换成 Excel 数字
IntToLetter	数字转换成 Excel 字符串
WriteTextToReport	写入文字
WriteTableToReport	写入表格
WriteListToReport	写入清单
InsertGraph	写入图表
WriteArrayToReport	写入数组
AppendRow	接续行写入
AppendColumn	接续列写入
ReadSingleCell	读取单一 cell 单元
ReadConsecutiveCells	读取连续 cell 单元
ReadRegionCells	读取指定区间的 cell 单元
ReadCurrentSheet	读取当前工作分页
ReadSheet	读取指定工作分页
ReadAllSheets	读取所有工作分页
ExecuteMacroFromScript	运行宏

2.12.3 Logger 类

Logger 类是一个线程安全的静态日志类，当前版本支持记录日志到单个文件或者到某个目录下。Logger 类的方法和属性如下：

Logger 类	
静态属性名	**功能说明**
LogLevel (LogLevel 枚举)	配置和获取当前日志的记录级别，包括 Trace/ Debug/ Info/ Warn/ Error/ Fatal，日志级别依次升高。记录时低于 LogLevel 级别的日志信息不会被记录到日志中。
LogConfig (LogConfig 类)	LogConfig 类类型，保存 Logger 的配置数据。
Enabled(bool)	配置是否启用日志记录功能。
静态方法名	**功能概要**
Initialize	初始化日志，可以入参日志文件路径(单文件日志)和配置类 LogConfig
Print	以指定的日志级别打印日志，包含多个重载
Trace	以 Trace 级别写入日志
Debug	以 Debug 级别写入日志
Info	以 Info 级别写入日志
Warn	以 Warn 级别写入日志
Error	以 Error 级别写入日志

Fatal	以 Fatal 级别写入日志
StackTrace	以指定级别写入当前的堆栈信息，默认为 Debug
ThreadInfo	以指定级别写入当前的线程信息，默认为 Debug
Close	关闭当前日志记录会话。
LogConfig 类	
属性名	**功能说明**
Type(LogType 枚举)	日志类型，目前只支持 FileLog，即文件日志
FileLog (FileLogConfig 类)	文件日志类型的配置类
LogFormat(string)	日志打印的 format，{0}为日志级别、{1}为时间戳、{2}为日志信息，默认为"[{0}] [{1}] {2}"，打印效果如： [Warn] [2018-06-06 11:55:02:685] This is a test.
ExceptionFormat(string)	日志打印的异常信息的 format，{0}为日志级别、{1}为时间戳、{2}为异常类型、{3}为异常信息。
StackTraceFormat(string)	日志打印时堆栈信息的 format，{0}是打印信息
ThreadInfoFormat(string)	日志打印时线程信息的 format，{0}是线程 ID、{1}是线程名称
TimeStampFormat(string)	时间戳格式，默认为：yyyy-MM-dd hh:mm:ss:fff
Header(string)	日志头
FileLogConfig 类	
属性名	**功能说明**
Extension(string)	文件扩展名，默认为 log
LogMode(FileLogMode 枚举)	日志模式，包括单文件日志和目录日志
Path(string)	日志文件路径或文件目录
LogNameFormat(string)	目录日志模式时日志文件名时间的 format，默认为： yyyy-MM-dd hh-mm-ss
MaxLogSize(long)	文件的最长字节数，默认为 100M，超过该值后：单文件日志清空文件写入；目录日志新建日志文件写入
Encode(Encoding 类)	编码格式，默认为 Unicode
Flush(FlushType 枚举)	文件刷新模式：SyncFlush(同步刷新)、AsyncFlush(异步刷新)

2.13 SeeSharpTools.JY.ThreadSafeQueue 类库

该类库包含线程安全的队列组件。

2.13.1 ThreadSafeQueue 类

ThreadSafeQueue 是基于 Queue 开发的线程安全的队列，可以保存数据的引用(非值拷贝)，所有数据都以 object 类型保存在队列中，在出队后仍为 object 类型，如果读写非 object 数据时需要类型强转。该队列在容量不够时会自动扩展。

ThreadSafeQueue 类	
属性名	**功能描述**
Destroyed(bool)	当前队列是否被销毁的标志
Count(int)	当前队列中数据的个数
方法名	**功能描述**
Clear	清空队列
Enqueue	数据入列(对象的引用)
Dequeue	数据出列

2.13.2 CircularQueue 类

CircularQueue 是基于数组开发的线程安全队列，使用泛型实现，可以支持保存除了 string 以外的对象，保存数据时会进行值拷贝。该队列的大小在构造时已经确定，为了保证不溢出，初始化时可以根据单次写入数据的多少，配置为一个较大的数值。同时该对象实现了 IList 接口，可以对保存在队列中的数据进行随机读写。

CircularQueue 类	
属性名	**功能描述**
Count(int)	获取队列中数据的个数
Capacity(int)	获取队列能容纳的最多元素个数，该参数在构造方法里赋值
AutoLock(bool)	是否使用悲观锁(即自动获取操作的锁)，默认为 true
BlockWait(bool)	读取数据时，如果数据不够是否阻塞读取线程，默认为 true
Disposed(bool)	获取当前队列是否不可用的状态。
方法名	**功能描述**
Clear	清空队列。
Enqueue	数据入列。如果获取锁超时，抛出异常
TryEnqueue	数据入列。如果获取锁超时，返回 false，否则返回 true
Dequeue	数据出列，如果获取锁超时，抛出异常
TryDequeue	数据出列，如果获取锁超时，返回 false，否则返回 true

Enter	乐观锁模式下获取操作队列的锁，超时抛出异常
TryEnter	乐观锁模式下获取操作队列的锁，超时返回 false
Leave	乐观锁模式下释放操作队列的锁。
Reset	重置序列可用状态，清空内部数据。
Dispose	释放阻塞的线程，标记序列未不可用。

2.14 SeeSharpTools.JY.Sensors 类库

SeeSharpTools.JY.Sensors 类库包含 6 种传感器计算类型，能够将采集到的值转换成传感器的物理量。

2.14.1 CustomScaling 类

CustomScaling 类	
方法名	功能描述
Convert	自定义 Func<double,double>函数

2.14.2 DisplacementSensor 类

DisplacementSensor 类	
方法名	功能描述
Convert	电阻式位移传感器

2.14.3 LoadCell 类

LoadCell 类	
方法名	功能描述
Convert	荷重元传感器

2.14.4 RTD 类

RTD 类	
方法名	功能描述
Convert	RTD 温度传感器

2.14.5 Thermistor 类

Thermistor 类	
方法名	功能描述
Convert	热敏电阻传感器

2.14.6 Thermocouple 类

Thermocouple 类	
方法名	功能描述
Convert	热电偶温度传感器（支持 B,E,J,K,N,R,S,T 型）

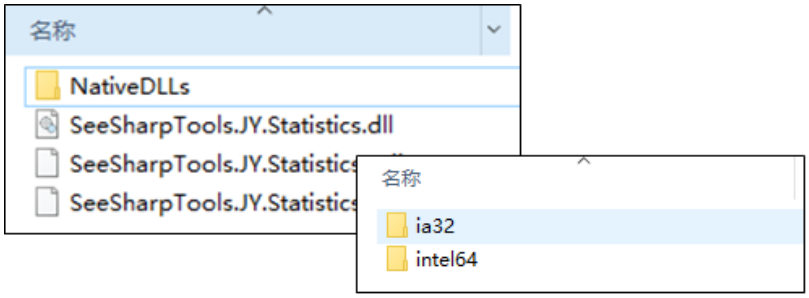
2.15 SeeSharpTools.JY.Statistics 类库

JY. Statistics 类库包含常用的统计功能共 9 中功能，整合 Math.NET 类库以及针对 Intel CPU 优化的高效能类库 Intel IPP 库（透过简单属性修改计算核心类别来切换）。

如果需要使用 IntelIPP 库，须上网下载、安装 Intel IPP 类库。将 dll 按照运行平台不同分成 ia32 和 intel64 的文件夹， 放在与 SeeSharpTools.JY.Statistics.dll 同个目录下，如下图所示

..\NativeDLLs\ia32\ →32 位 dll

..\NativeDLLs\intel64 → 64 位 dll



Engine 类	
属性	功能描述
Provider	选择运行引擎使用 Math.NET 或是 IntelIPP

Statistics 类	
方法名	功能描述
Histogram	画出数组统计直方图
Kurtosis	计算出 Kurtosis 值
Maximum	计算出 Maximum 值
Mean	计算出 Mean 值
Minimum	计算出 Minimum 值
RMS	计算出 RMS 值
Skewness	计算出 Skewness 值
StandardDeviation	计算出 StandardDeviation 值
Variance	计算出 Variance 值
CalculateCapability	计算一组样点的统计过程控制(SPC)指标，返回 CapabilityResult 数据结构。
CapabilityResult 类	

属性名	功能描述
Mu	样点均值
Sigma	样点标准差
P	样点出现在上下限的统计概率
Pl	样点低于下限的统计概率
Pu	样点高于上限的统计概率
Cp	(High - Low)/(6*Sigma)
Cpl	(Mu - L)./(3*sigma)
Cpu	(U - Mu)/(3*Sigma)
Cpk	min(Cpl, Cpu)
Ca	(Sigma-C)/(Et*0.5)

注意事项：

1. 该组件执行时依赖 MCR 2017a 组件，否则会抛出引擎加载失败的异常。如果当前计算机未安装 MCR 可以在以下路径中安装 MCR 2017a 和 2017a 版本的更新包：MCR installer、MCR Update Package。
2. 当前组件运行时依赖 StatisticsMatlab.dll 和 MWArray.dll，一般情况下编译时 StatisticsMatlab.dll 和 MWArray.dll 会被自动编译到执行目录，执行目录在 Debug(Release)模式时默认为工程的根目录/bin/Debug(Release)文件夹。在某些情况下可能不会拷贝 StatisticsMatlab.dll 和 MWArray.dll 到执行目录，这会导致应用抛出组件加载失败的异常，出现该异常时手动拷贝 SeeSharpTools 安装目录/bin 文件夹下的这两个文件到执行目录即可解决该异常。

2.16 SeeSharpTools.JY.TCP 类库

JY.TCP 类库包含两个类：JYTCPServer 以及 JYTCPClient 类库。

JY.TCP 类库提供了两种资料传输模式，第一种是字符串模式，第二种是数据模式，需要在一开始创建对象的时候选择。Server 支持多个 Client 连线。

JYTCPServer 类	
属性名	功能概要
ConnectedClients	当前连接到该服务端的客户端信息
BufferSize	缓存大小
LocalIP	当前服务器绑定的 IP 地址
方法名	功能概要
Start	开始任务

Stop	停止任务
ReadDataStream	从指定客户端读取数据，当前支持 double/float/类型的一维数组和二维数组
ReadString	从指定客户端读取字符串
SendDataStream	向指定客户端发送数据，当前支持 double/float 类型的一维数组和二维数组
SendString	向指定客户端发送字符串
事件	**功能概要**
ClientDisconnected	客户端断线事件
ClientConnected	客户端建立连线事件

JYTCPClient 类	
属性名	**功能概要**
AvailableSamples	返回对象缓存中的数目
BufferSize	缓存大小
Connected	判定是否连接成功
方法名	**功能概要**
Connect	开始连接
DisConnect	终止连接
ReadDataStream	从服务端读取数据，当前支持 double/float/类型的一维数组和二维数组
ReadString	从服务端读取字符串
SendDataStream	向服务端发送数据，当前支持 double/float 类型的一维数组和二维数组
SendString	向服务端发送字符串
事件	**功能概要**
ServerDisconnected	服务器端断开事件

2.17 SeeSharpTools.JY.Mathemetics 类库

JY.Mathematics 类库提供各种数学计算类库，包含数组操作、数组计算、统计、微积分、曲线拟合、内插算法、线性代数等静态类库。

ArrayArithmetic 类	
方法名	功能概要
Absolute	绝对值
Acos	反余弦
Add	数组相加
Asin	反正弦
Atan	反正切
Cos	余弦
Divide	数组相除
Exp	自然指数
FindMaxMin	找最大最小值
Initialize	初始化
Ln	自然对数
Log	对数
Multiply	数组相乘
Pow	指数
Product	数组相乘后相加
Sin	正弦
Sqrt	平方根
Substract	数组相减
Sum	求和
Tan	正切

ArrayOperation 类	
方法名	功能概要
Concatenate	拼接
ConvertTo	转换
Copy	拷贝
Delete	删除
GetSubset	获取部分
Insert	插入
Inverse	反转
ReplaceSubset	置换部分
Transpose	转置

Statistics 类	
方法名	功能概要
Histogram	直方图
Kurtosis	峰度
Mean	平均值
Median	中位数
Percentile	百分位数
RMS	均方根值
Skewness	偏度
StandardDeviation	标准差
Variance	方差

LinearAlgebra 类	
方法名	功能概要
Dot	点积
Norm	范数

Interpolation 类	
方法名	功能概要
Interpolate_CubicSpline	三次样条插值
Interpolate_Linear	线性内插
Interpolate_LogLinear	对数线性内插
Interpolate_Polynomial	多项式插值
Interpolate_Step	步进插值

Fitting 类	
方法名	功能概要
ExponentialFitting	指数拟合
LinearFitting	线性拟合
TransposePolynomialFitting	多项式拟合

Calculus 类	
方法名	功能概要
Derivative_2ndOrderCentral	二次微分
Derivative_4thOrderCentral	四次微分
Derivative_Backward	反向微分
Derivative_Forward	顺向微分
Integral_Bode	Bode 积分
Integral_Simpsons	Simpson 积分
Integral_Simpsons38	Simpson3/8 积分
Integral_Trapezodial	梯形积分

Vector 类	
方法名	功能概要
ArrayAbs	数组取模
ArrayAdd	数组加法
ArrayAscend	数组升序
ArrayAscendIndex	数组升序，并返回索引位置
ArrayConvert	数组转换类型
ArrayCopy	数组拷贝模板
ArrayDescend	数组降序
ArrayDescendIndex	数组降序，并返回索引位置
ArrayDivision	数组除法
ArrayDotProduct	数组求点积
ArrayDotProductConj	数组求点积
ArrayEqual	比较 Array 是否相同
ArrayExp	数组取 Exp
ArrayExp10	数组取 Exp10
ArrayExp2	数组取 Exp2
ArrayFindMax	数组取最大值
ArrayFindMin	数组取最小值
ArrayFindMaxMin	数组取最大和最小值
ArrayLn	数组取 Ln
ArrayLog10	数组取 Log10
ArrayLog2	数组取 Log2
ArrayMax	取两个数组对应位置上较大的
ArrayMean	数组取平均
ArrayMin	取两个数组对应位置上较小的
ArrayMulti	两数组相乘
ArrayPhaseMean	相位数组求平均
ArrayPhaseMeanWeight	相位数组求平均，向量和
ArrayReserve	数组反转
ArrayRoot	数组平方根
ArrayRotate	数组旋转
ArrayScale	数组乘以常数
ArraySquare	数组取平方
ArraySub	数组减法
ArraySum	数组求和
ArrayTranspose	数组转置

JY.Mathematics 里头特定含式支持基于 intel 技术的加速运算能力，操作方法以及支

持列表如下:

【使用 Intel IPP 加速】

1. 安装 intel IPP 软件包（从 intel 网站上下载）
2. 将生成目录新增一个”IntelIPP”文件夹
3. 将 intel 软件包中的 dll 拷贝到”IntelIPP”目录底下。如果需要 x86 的程序就拷贝 x86 的 dll, x64 的程序就拷贝 x64 的 dll。

   x86 路径

   C:\Program Files (x86)\IntelSWTools\compilers_and_libraries_2017.4.210\windows\redist\ia32\ipp

   x64 路径

   C:\Program Files (x86)\IntelSWTools\compilers_and_libraries_2017.4.210\windows\redist\intel64\ipp
4. 在代码中加入一行

   Engine.Provider = ProviderEngine.IntelIPP
5. 支持含式列表如下

ArrayArithmetics	ArrayOperation	Statistics	LinearAlgebra
Absolute	Inverse	Mean	Dot
Acos		StandardDeviation	
Add			
Asin			
Atan			
Cos			
Divide			
Exp			
FindMaxMin			
Initialize			
Ln			
Log			
Multiply			
Pow			
Product			
Sin			
Sqrt			
Substract			
Sum			
Tan			

【使用 Intel MKL 加速】

1. 安装 MKL runtime 软件包（从简仪科技网站上下载）

2. 在代码中加入一行

Engine.Provider = ProviderEngine.IntelMKL

ArrayArithmetics	Statistics	LinearAlgebra
Absolute	Kurtosis	Dot
Acos	Skewness	
Add		
Asin		
Atan		
Cos		
Exp		
FindMaxMin		
Ln		
Log		
Multiply		
Pow		
Sin		
Sqrt		
Substract		
Sum		
Tan		

2.18 SeeSharpTools.JY.DSP.Measurements 类库

JY.DSP.Measurements 类库提供测量的类库，目前包含方波测量的静态类库。

SquarewaveMeasurements 类	
方法名	功能概要
AmplitudeLevelAnalysis	方波幅度分析：根据高低电平状态配置（峰值或直方图），分析输入波形的幅值信息，输出幅值和高低电平值。
PeriodAnalysis	方波周期分析：根据输入波形和波形采样间隔，百分比电平配置，脉冲极性，参考电平配置（百分比或绝对值，高电平，中间电平和低电平），分析方波的周期特性。输出：周期（周期个数，最大值，最小值和平均值），脉冲（脉冲个数，最大值，最小值和平均值），占空比（占空比周期个数，最大值，最小值和平均值）
PulseMeasurement	方波脉冲测量：根据输入波形和波形采样间隔，百分比电平配置，脉冲极性，参考电平配置（百分比或绝对值，高电平，中间电平和低电平），脉冲序号，分析方波的脉冲参数。输出：指定序号脉冲的周期，脉冲宽度，占空比，和测量信息（包括脉冲中心的时刻，测量所使用的参考电平设置）
TransitionMeasurement	方波边沿测量：根据输入波形和波形采样间隔，百分比电平配置，边沿方向，参考电平配置（百分比或绝对值，高电平，中间电平和低电平），边沿序号，分析方波的边沿参数。输出：指定边沿的斜率，上升/下降时间，边沿过冲和欠冲信息和测量信息（包括，边沿开始和结束时间，测量边沿所采用的参考电平设置）

简仪科技
JYTEK

3. 常见问题解答

如在使用中遇到其它问题，请联系上海简仪科技有限公司。我们的联系邮箱：mailto:support@jytek.com