

---

上海格西信息科技有限公司

---

数据采集、监视与分析

版本 0.1

# 目录

1. 概述	3
2. 创建项目	3
2.1 第 1 步 新建项目	3
2.2 第 2 步 添加串口设备	4
2.3 第 3 步 添加序列	5
2.4 第 4 步 添加变量	8
2.5 第 5 步 使用脚本关联序列数据和变量	11
2.6 第 6 步 添加界面	13
3. 运行项目	16
3.1 打开并运行项目	16
3.2 使用序列数据面板监视步骤	16
3.3 使用变量数据面板采集和监视数据	17
3.4 使用变量数据面板回放数据	19
3.5 使用设备数据面板监视数据	20
3.6 序列数据的导出和导入	20
3.7 变量数据的导出和导入	20

## 1. 概述

某热电偶采集模块，采用 Modbus RTU 通信协议，其中读温度命令为 03 命令，地址为 0000，温度值为 16 位有符号整数。

本项目演示读温度，然后把采集时间和温度数据保存到文本文件型变量“温度数据”中，最后用曲线图显示温度随时间变化过程。

**本例子文件位于：** <软件安装目录>\Examples\Solutions\SCADA。

文件说明：

- ✓ SCADA. gpj - 数据采集与监控演示项目 - 中文 - 串口版
- ✓ SCADA. en. gpj - 数据采集与监控演示项目 - 英文 - 串口版
- ✓ SCADA. Net. gpj - 数据采集与监控演示项目 - 中文 - 网口版
- ✓ SCADA. Net. en. gpj - 数据采集与监控演示项目 - 英文 - 网口版

所有例子自带仿真器，可以脱离设备仿真运行。

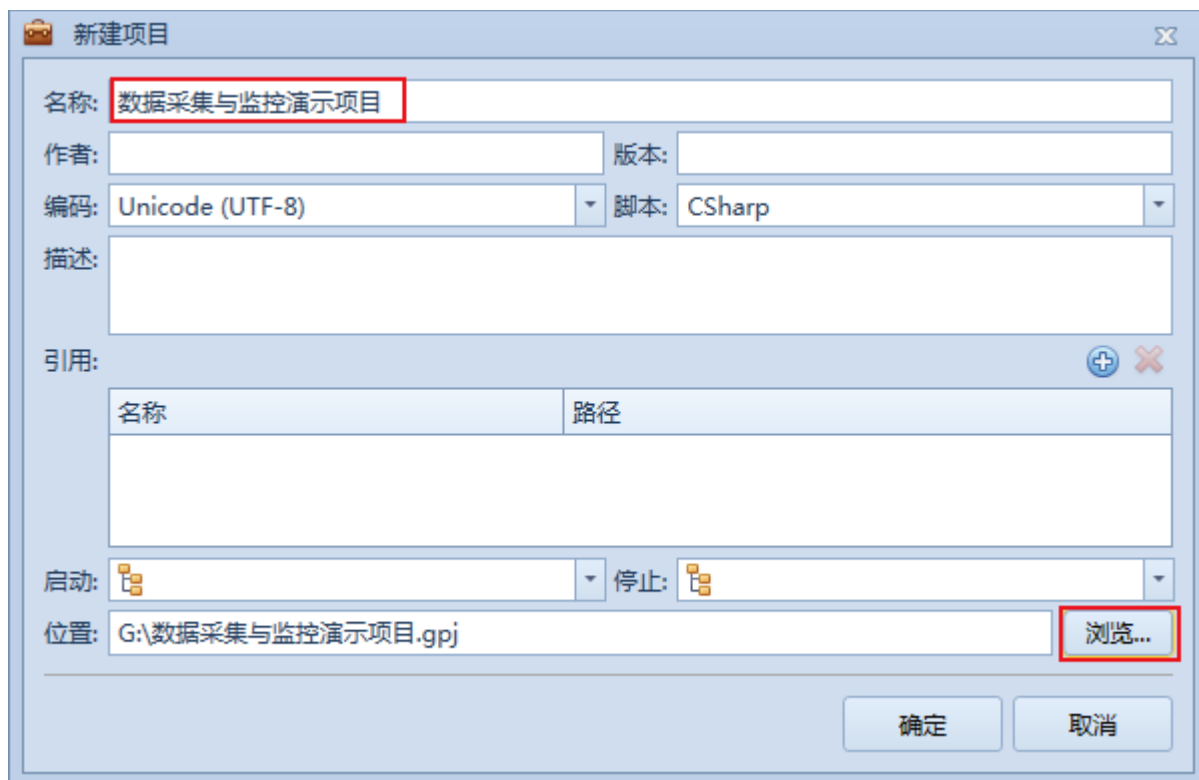
串口版：需要使用串口虚拟软件，如 VSPD 等，虚拟出一对串口进行仿真运行。如果虚拟的串口号和例子预定义的串口号不同，可以修改例子串口号，也可以修改虚拟串口号。

网口版：统一采用本地 IP 地址 127. 0. 0. 1，如果端口号被本机其他软件占用，则自行修改例子网口的端口号。**网口版不需要虚拟串口，配置起来比串口版简单，推荐！**

## 2. 创建项目

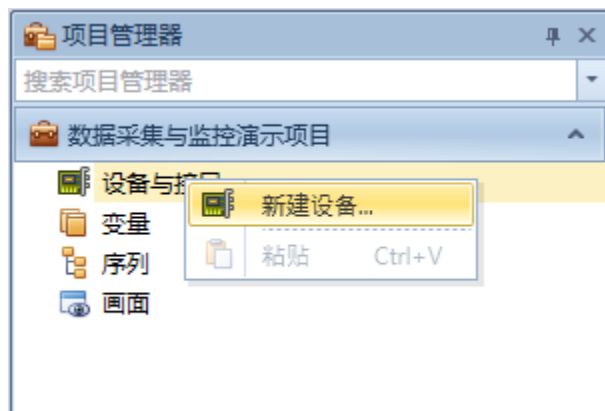
### 2.1 第 1 步 新建项目

启动格西测控大师，在左上角菜单中选择“新建项目”，然后在弹出的“新建项目”对话框中，填写项目名称“数据采集与监控演示项目”，然后点击“浏览...”按钮，选择保存路径和填写项目文件名“数据采集与监控演示项目”，最后点击“确定”按钮。

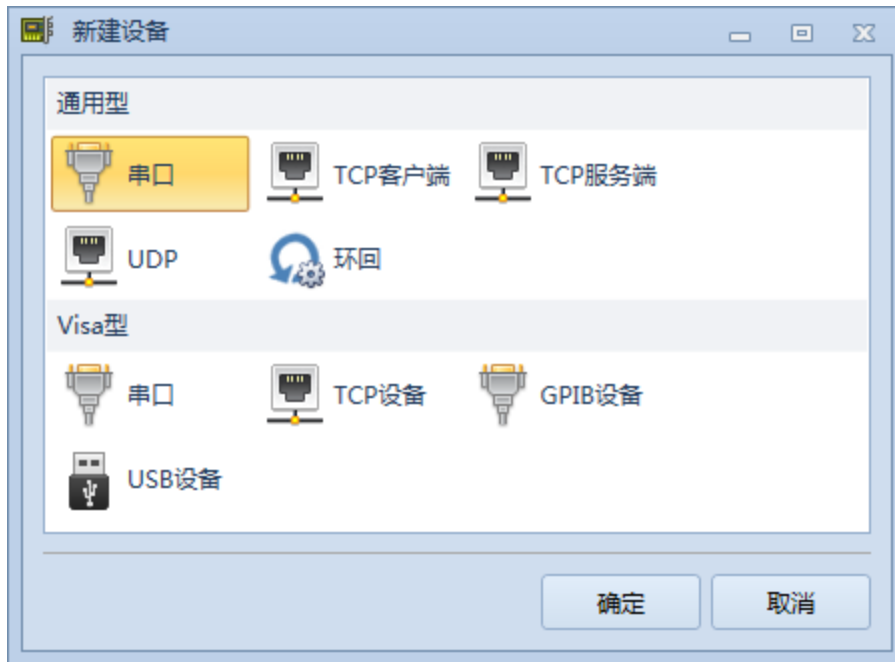


## 2.2 第2步 添加串口设备

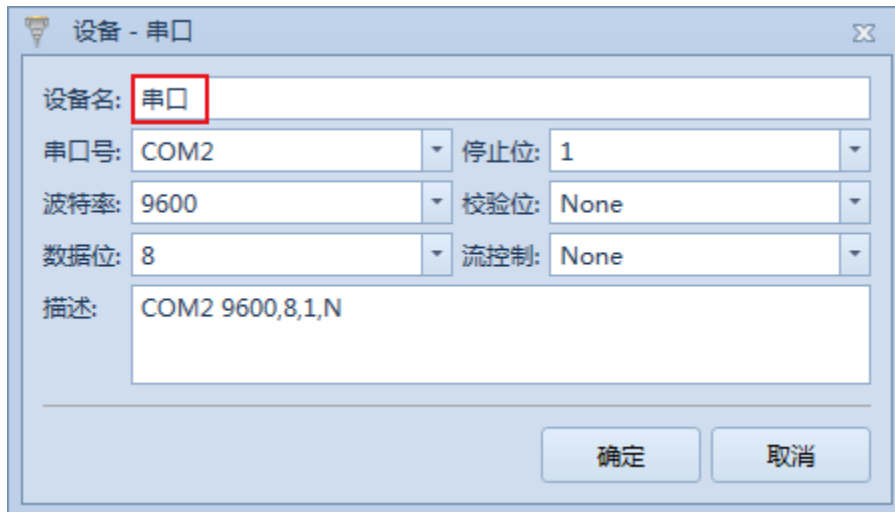
在项目管理器中选择“设备与接口”节点，然后点击鼠标右键，在弹出菜单中选择“新建设备...”。



弹出新建设备对话框中，选择“串口”，点击“确定”。

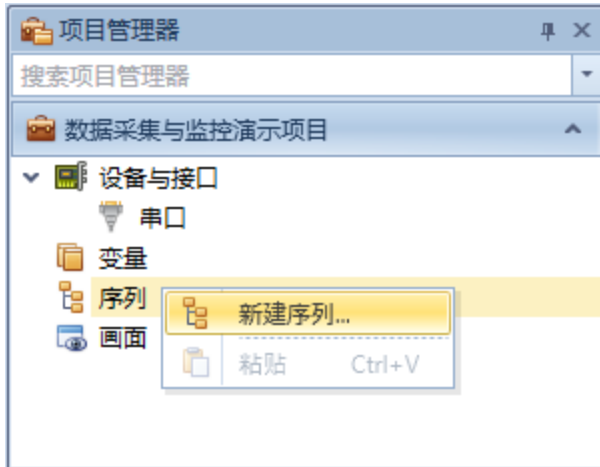


弹出设备属性对话框，填写“设备名”和其他设备参数，最后点击“确定”按钮。其中，“设备名”是设备的标识，可以是任意字符串，引用设备必须使用设备名。

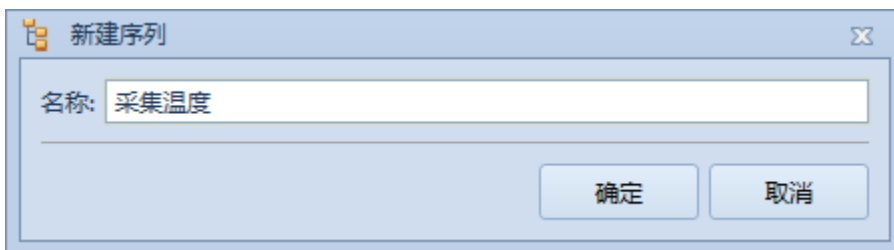


### 2.3 第3步 添加序列

在项目管理器中选择“序列”节点，然后点击鼠标右键，在弹出菜单中选择“新建序列...”。

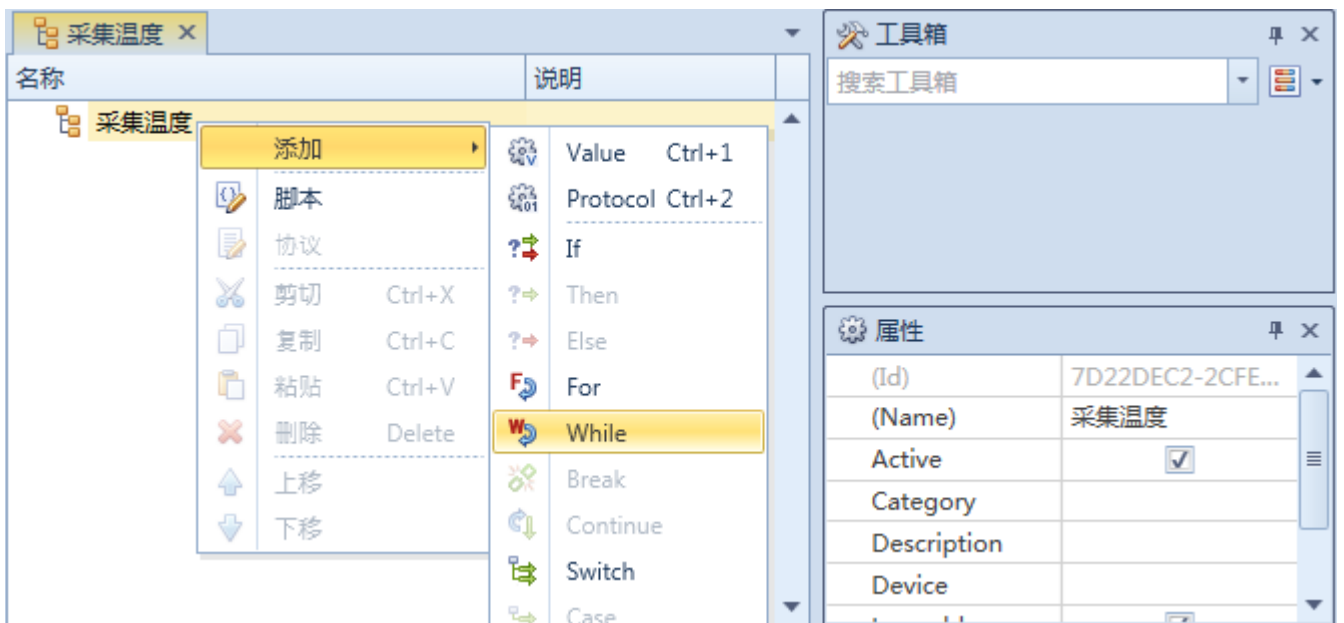


弹出新建序列对话框中，填写“名称”，点击“确定”。

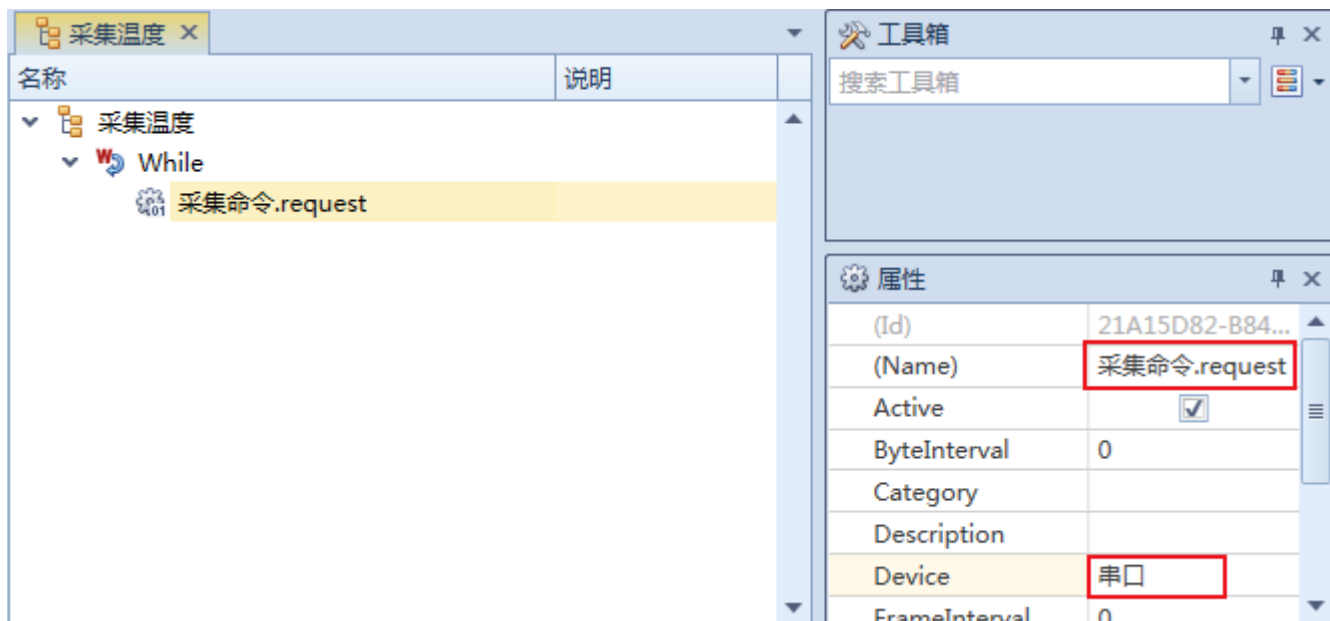


在项目管理器中选择新创建的“采集温度”节点，然后双击鼠标左键，或者点击鼠标右键，在弹出菜单中选择“编辑...”，打开序列编辑页面。

在“采集温度”编辑页面，选中“采集温度”节点，单击鼠标右键，在弹出菜单中选择“添加 -> While”，添加 While 类型步骤，条件参数 ConditionExpression 设置为 True，无限循环执行采集任务。



接下来，添加发送采集命令，选中“While”节点，单击鼠标右键，在弹出菜单中选择“添加 -> Protocol”，添加协议类型步骤，然后在属性面板，修改步骤名称“(Name)”为“采集命令.request”，TransceiveMode 设置为 Send，表示主动发送，Device 属性选择名称为“串口”的设备。

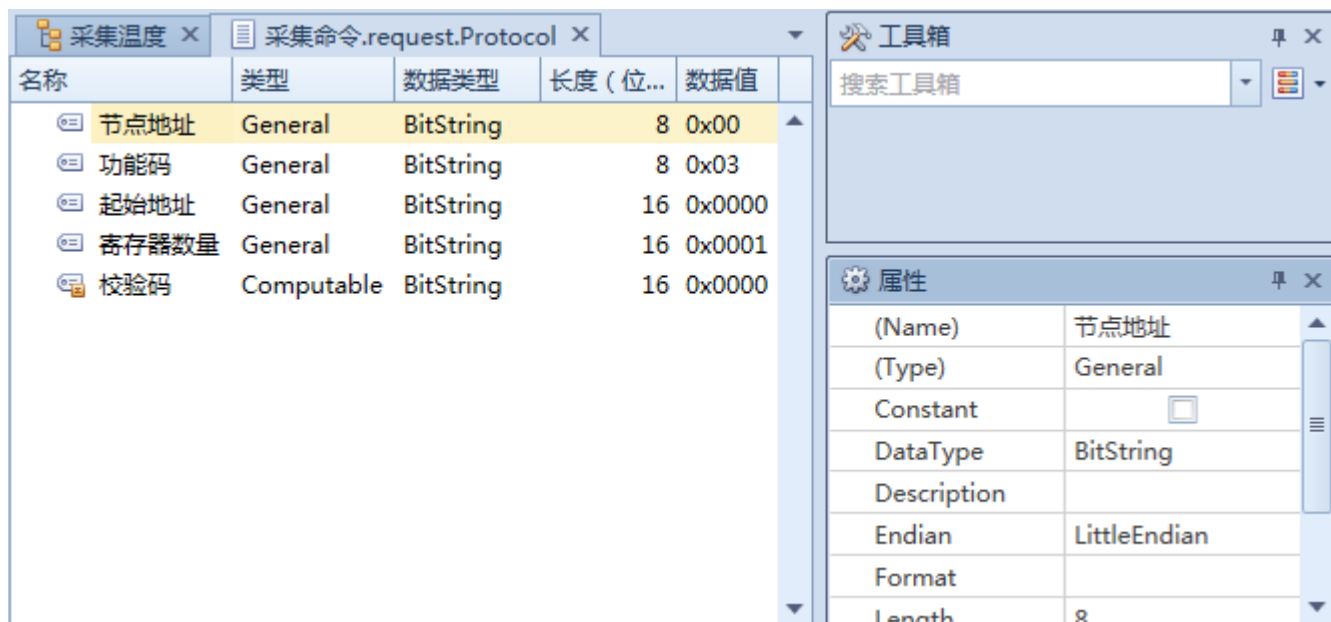


接下来，编辑发送采集命令的协议数据，选中“采集命令.request”节点，单击工具栏的“协议”按钮，打开协议编辑界面，通过单击鼠标右键，在弹出菜单中选择“添加 -> 协议字段”，依次按以下属性表添加协议字段。

(Name)	(Type)	Constant	DataType	Endian	Length	Value
节点地址	General		BitString	LittleEndian	8	0x00
功能码	General	√	BitString	LittleEndian	8	0x03
起始地址	General	√	BitString	BigEndian	16	0x0000
寄存器数量	General		BitString	BigEndian	16	0x0001
校验码	Computable	√	BitString	LittleEndian	16	0x0000

其中，校验码的参数配置如下表。

属性	值	描述
Algorithm	CRC16MODBUS	计算的算法
Priority	1	计算优先级，在有多个计算型字段时有用。
Location	Back	表示计算型字段在需要计算的数据后面。
StartPosition	0	起始计算字节序号
EndPosition	-1	结束计算字节序号，-1 表示计算到该字段前面。



接下来，添加接收采集数据命令，选中“While”节点，单击鼠标右键，在弹出菜单中选择“添加 -> Protocol”，然后在属性面板，修改步骤名称“(Name)”为“采集命令.response”，TransceiveMode 设置为 Receive，表示接收，Device 属性选择名称为“串口”的设备。

接下来，编辑接收采集数据命令的协议数据，选中“采集命令.response”节点，单击工具栏的“协议”按钮，打开协议编辑界面，通过单击鼠标右键，在弹出菜单中选择“添加 -> 协议字段”，依次按以下属性表添加协议字段。

(Name)	(Type)	Constant	DataType	Endian	Length	Value
节点地址	General		BitString	LittleEndian	8	0x00
功能码	General	√	BitString	LittleEndian	8	0x03
字节数	General	√	BitString	LittleEndian	8	0x02
温度	General		Int16	BigEndian	16	0
校验码	Computable	√	BitString	LittleEndian	16	0x0000

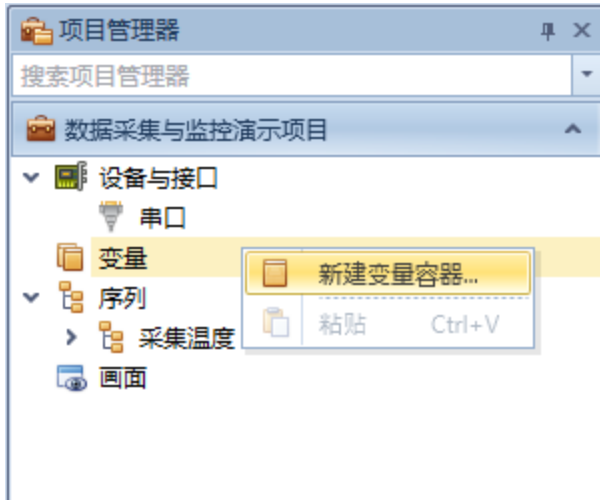
其中，校验码的参数配置如下表。

Algorithm	Priority	Location	StartPosition	EndPosition
CRC16MODBUS	1	Back	0	-1

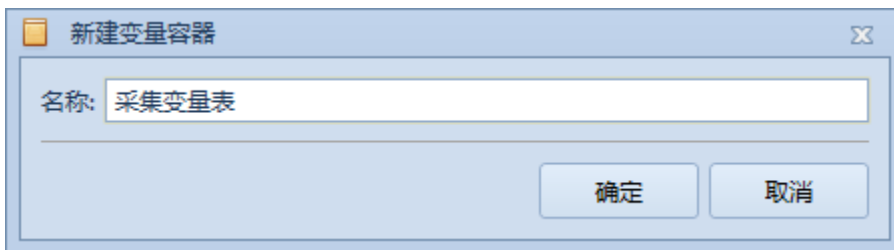
## 2.4 第4步 添加变量

在项目管理器中选择“变量”节点，然后单击鼠标右键，在弹出菜单中选择“新建变量容器...”。



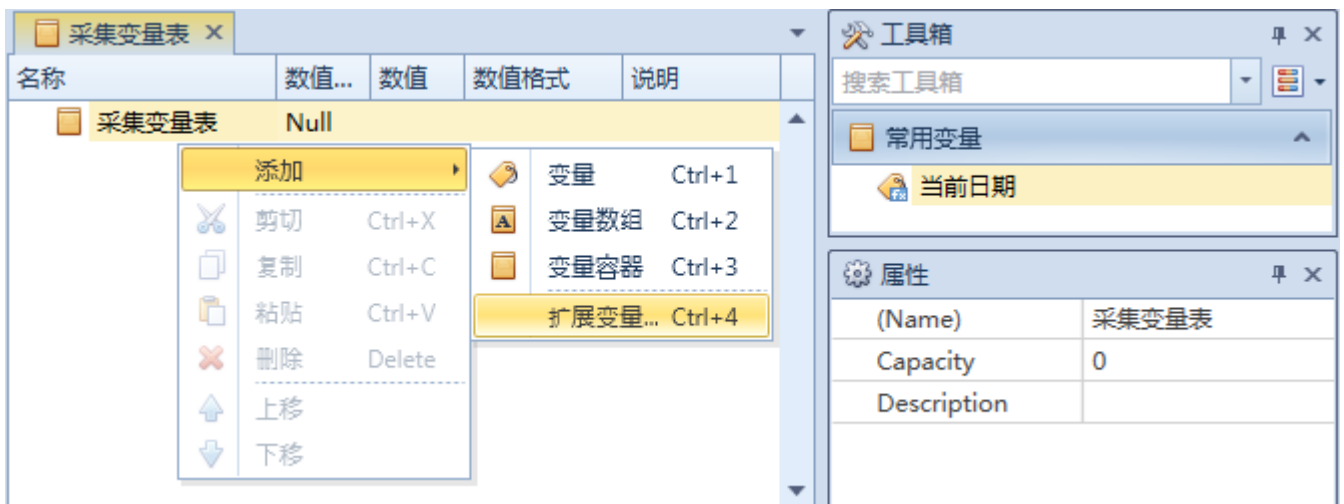


弹出新建变量容器对话框中，填写“名称”，点击“确定”。

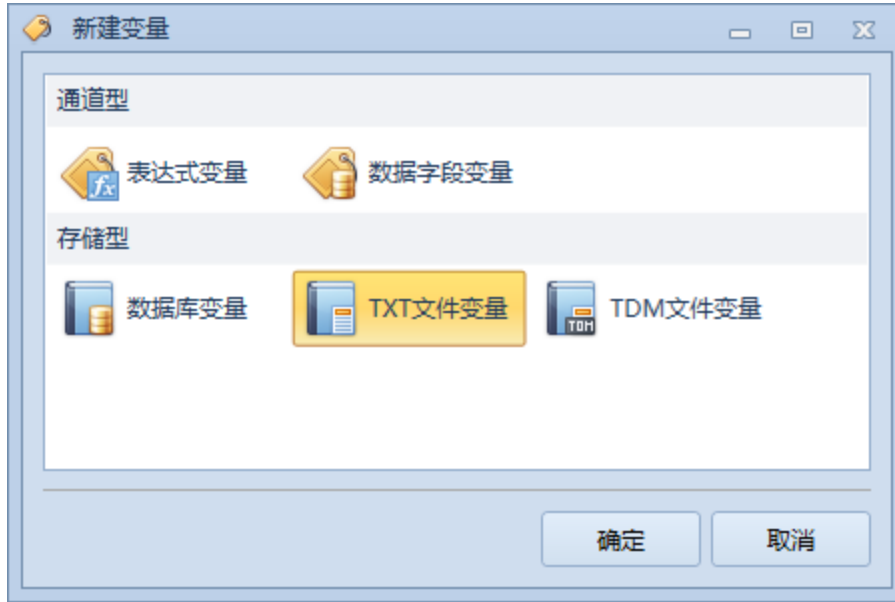


在项目管理器中选择新创建的“采集变量表”节点，然后双击鼠标左键，或者点击鼠标右键，在弹出菜单中选择“编辑...”，打开变量编辑页面。

在“采集变量表”编辑页面，选中“采集变量表”节点，单击鼠标右键，在弹出菜单中选择“添加 -> 扩展变量...”。



弹出新建变量对话框中，选择“TXT 文件变量”，点击“确定”。



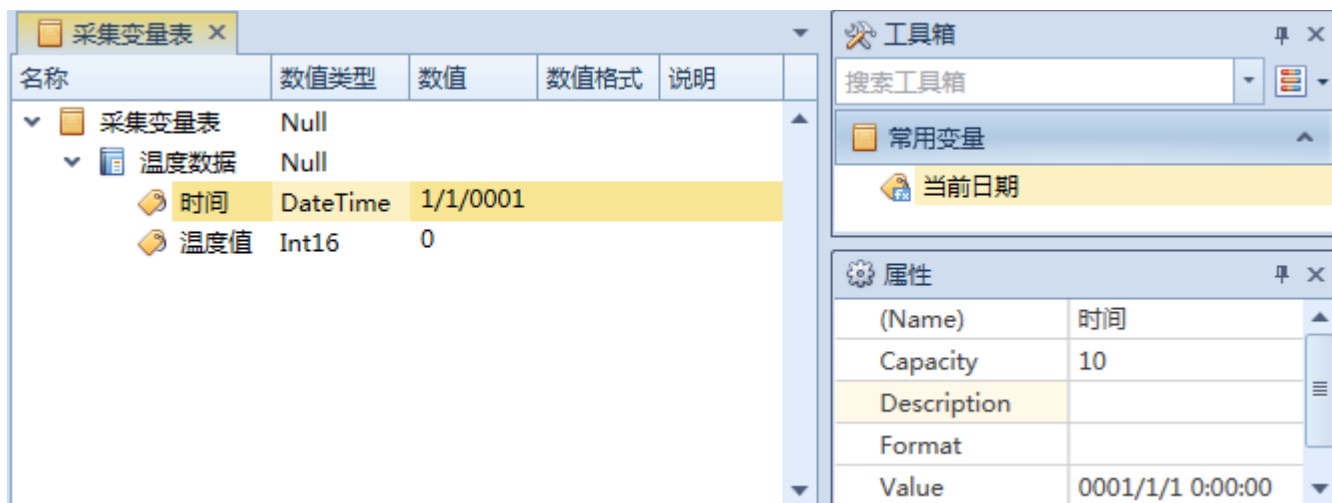
接下来，选中新创建的 TXT 文件变量，在属性面板，依次按以下属性表修改属性值。

属性	值	描述
(Name)	温度数据	变量名称
Capacity	10	变量缓存容量，采集速度越高，缓存要越大。 该属性值必须大于 0，否则不能触发存储机制，无法保存数据，也无法在变量数据面板中查看数据。
Directory	D:\Temp	文件存储的目录
FileName	温度数据.txt	文件名

接下来，选中“温度数据”节点，单击鼠标右键，在弹出菜单中选择“添加 -> 变量”，连续添加两个变量，分别命名为“时间”和“温度值”，在属性面板，依次按以下属性表修改属性值。

属性	值	描述
(Name)	时间	
Capacity	0	
ValueType	DateTime	

属性	值	描述
(Name)	温度值	
Capacity	0	
ValueType	Int16	

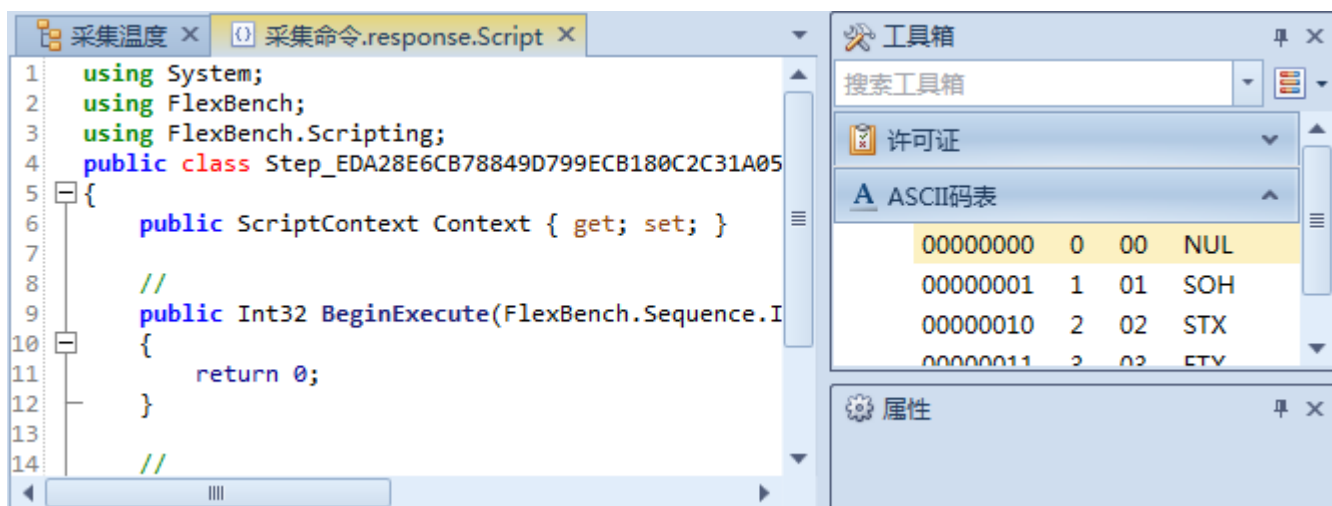


文本文件类型变量，当其所有子变量都改变过后，形成一条记录，保存到文件中。本例子中的温度数据，每次采集完毕，通过脚本把采集时间和温度值分别赋给“时间”和“温度值”变量，即可生成一条记录。

## 2.5 第5步 使用脚本关联序列数据和变量

序列中采集到的温度数据，可以通过脚本给变量表中的变量赋值，进而关联采集数据和变量值。软件系统支持的脚本类型有 C#、Visual Basic 和 Python，可以在项目属性中设置，本例子使用 C# 脚本进行演示。

在“采集温度”序列编辑页面，选中“采集命令.response”节点，单击工具栏的“脚本”按钮，即可打开脚本编辑页面。



在打开的脚本编辑页面中，显示“采集命令.response”步骤的脚本代码。下面是 C# 版本的步骤脚本模版。

```

// 命名空间
using System;
using Genesis;
using Genesis.Scripting;

```

```
///  
/// 脚本类, 类名  
///  
public class Step_EDA28E6CB78849D799ECB180C2C31A05  
{  
    // 脚本上下文  
    public ScriptContext Context { get; set; }  
  
    // 步骤开始执行之前执行。  
    // 参数: context - 步骤运行时上下文  
    //       step - 当前执行的步骤  
    // 返回: 暂不定义  
    public Int32 BeginExecute(Genesis.Sequence.IStepContext context,  
        Genesis.Sequence.IStep step)  
    {  
        return 0;  
    }  
  
    // 步骤执行完毕之后执行。  
    // 参数: context - 步骤运行时上下文  
    //       step - 当前执行的步骤  
    // 返回: 暂不定义  
    public Int32 EndExecute(Genesis.Sequence.IStepContext context,  
        Genesis.Sequence.IStep step)  
    {  
        return 0;  
    }  
}
```

接下来, 在 EndExecute 函数中实现提取采集的数据并赋值给指定的变量。

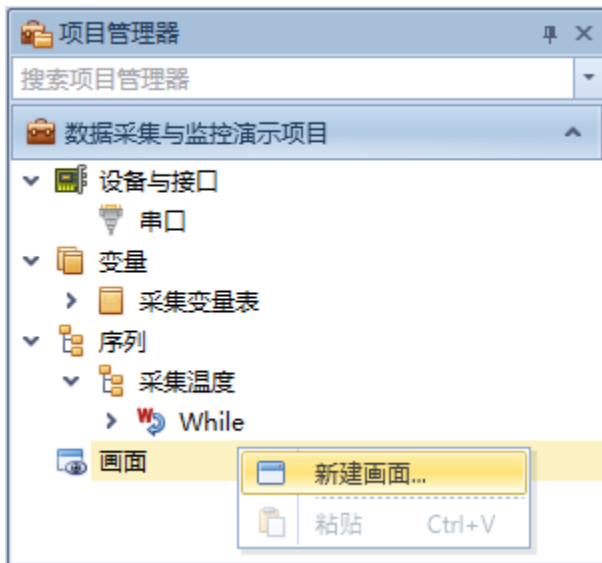
```
public Int32 EndExecute(Genesis.Sequence.IStepContext context,  
    Genesis.Sequence.IStep step)  
{  
    if (step.Result.Status == (int) Genesis.Sequence.ResultStatus.Passed)  
    {  
        // 提取采集的温度值, 索引号是 3。  
        Int16 temperature = (Int16)step.Result.DataFields[3].Value;  
        DateTime time = DateTime.Now;  
        // 设置变量表的变量值, 变量表为 Variants 容器, 通过路径的方式访问。  
        context.Variants["采集变量表/温度数据/时间"] = time;  
        context.Variants["采集变量表/温度数据/温度值"] = temperature;  
    }  
    return 0;  
}
```

至此，已经完成温度数据的采集和保存功能，保存的数据格式如下图所示。

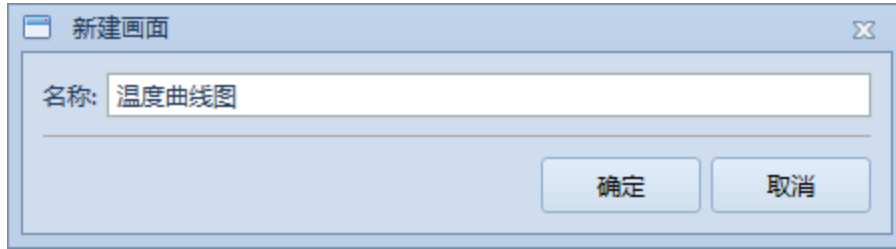


## 2.6 第6步 添加界面

在项目管理器中选择“画面”节点，然后点击鼠标右键，在弹出菜单中选择“新建画面...”。

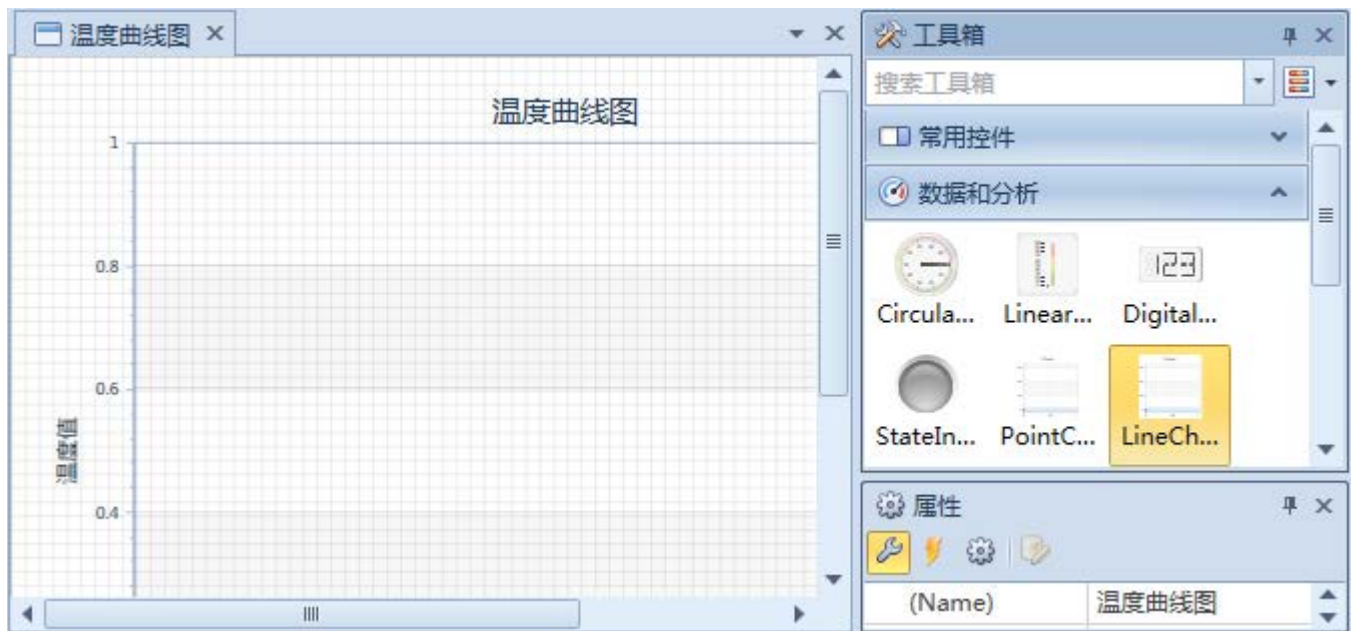


弹出新建画面对话框中，填写“名称”，点击“确定”。



在项目管理器中选择新创建的“温度曲线图”节点，然后双击鼠标左键，或者点击鼠标右键，在弹出菜单中选择“编辑...”，打开画面编辑页面。

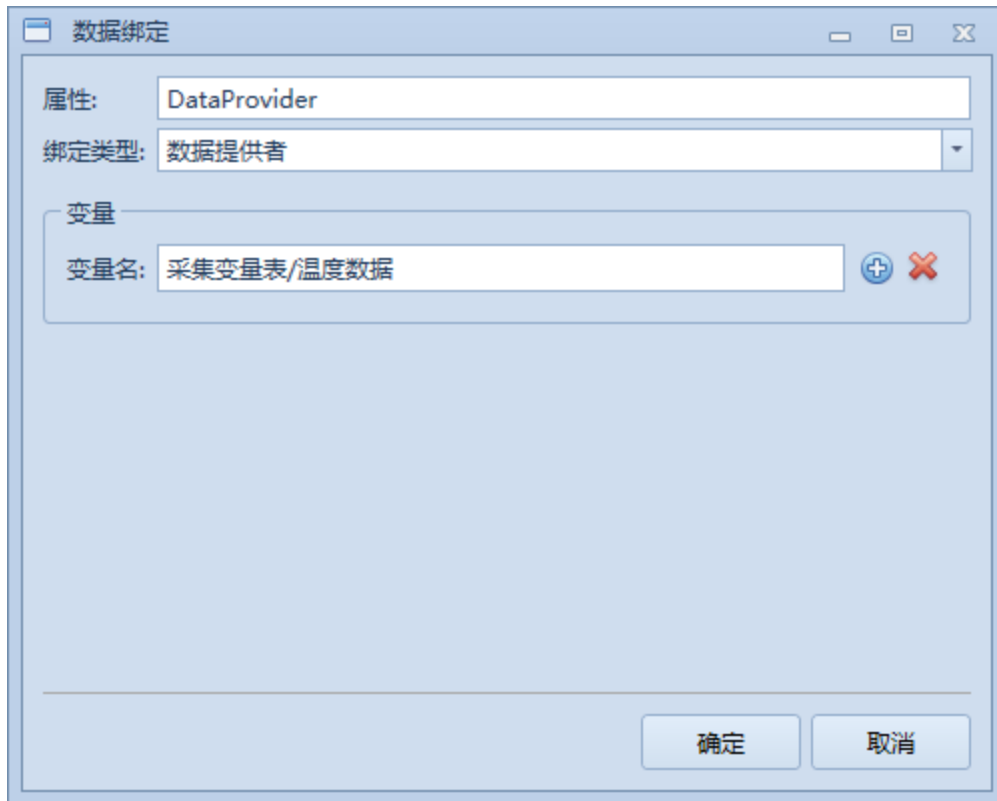
点击工具箱右上角菜单，在弹出菜单中选择“打开模具...”，在 Controls 目录下选择模版文件“DataAndAnalytics.schema”，点击“打开”；然后用鼠标在工具箱中选中“LineChart”条目，在画面中创建一个曲线显示控件。



接下来，在属性面板，依次按以下属性表修改控件的属性值。

属性	值	描述
AxisXScaleType	DateTime	X 轴的数据类型
AxisXTitle	时间	X 轴标题
AxisYTitle	温度值	Y 轴标题
Title	温度曲线图	总标题
ChartSeries	温度曲线, 时间, 温度值	图表曲线系列，格式为“<系列名称 1>, <X 变量名>, <Y 变量名>; <系列名称 2>, <X 变量名>, <Y 变量名>;...”
DataProvider		数据提供者，通过绑定数据的方法和变量关联。

接下来，处理数据绑定问题，选择 DataProvider 属性，点击“创建数据绑定”按钮，弹出数据绑定对话框，绑定类型选择“数据提供者”，然后点击“添加变量”按钮，在弹出的变量选择对话框中选择“温度数据”变量，点击“确定”。



至此，温度曲线图界面功能完成。

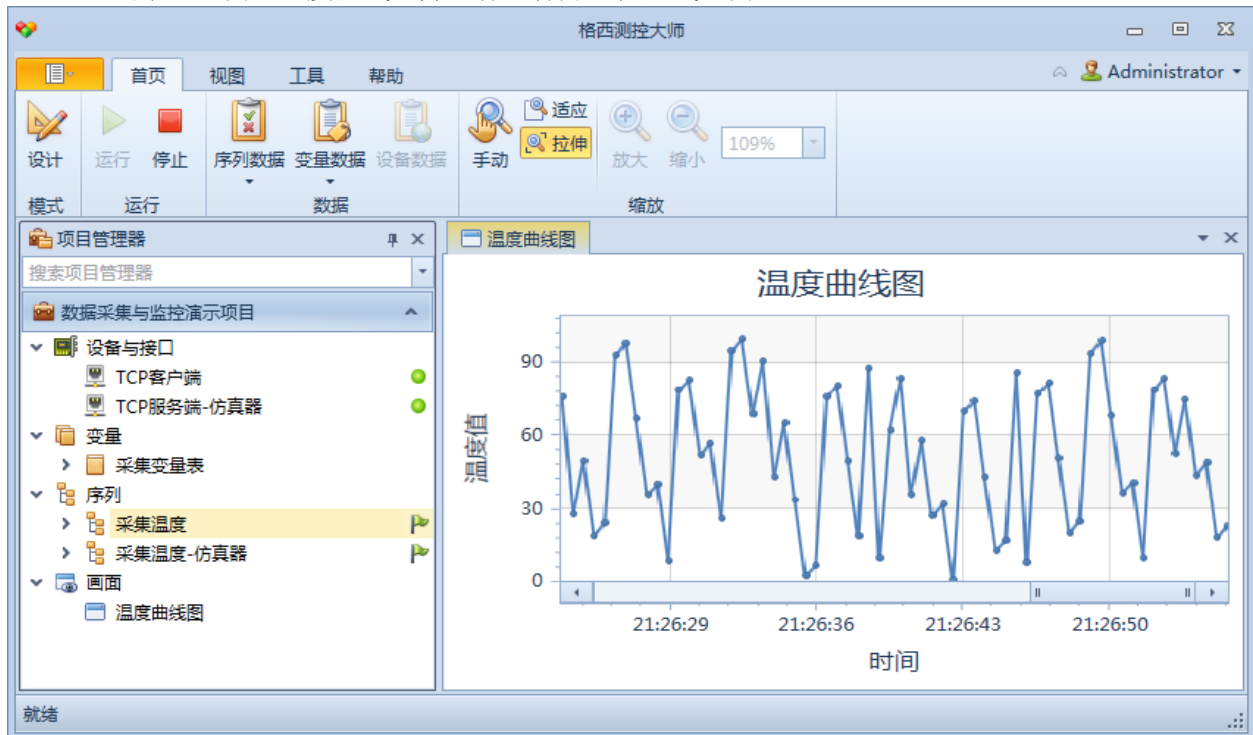


### 3. 运行项目

#### 3.1 打开并运行项目

从<软件安装目录>\Examples\Solutions\SCADA 目录中，打开 SCADA.Net.gpj 网口版项目文件。点击工具栏的“运行”按钮。

- 1) 选中“设备与接口->TCP 服务器-仿真器”并运行
- 2) 选中“设备与接口->TCP 客户端”并运行
- 3) 选中“序列->采集温度-仿真器”并运行，仿真温度传感器设备
- 4) 选中“序列->采集温度”并运行，启动采集命令
- 5) 选中“画面->温度曲线”并运行，打开监控曲线画面

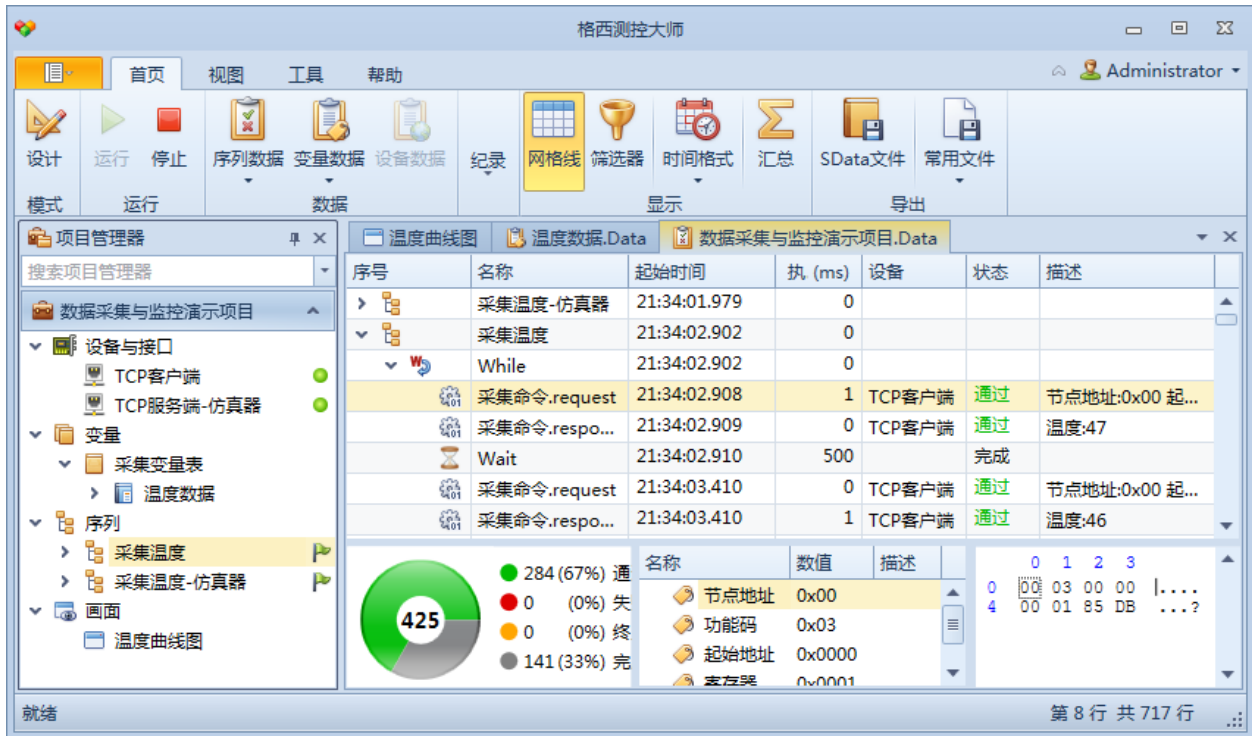


#### 3.2 使用序列数据面板监视步骤

点击工具栏的“序列数据”按钮，打开序列数据面板。数据面板上部分采用树形结构记录步骤执行结果，忠实反映序列的步骤结构；数据面板下部分显示选中执行结果的统计信息、结果字段的解析信息等内容。

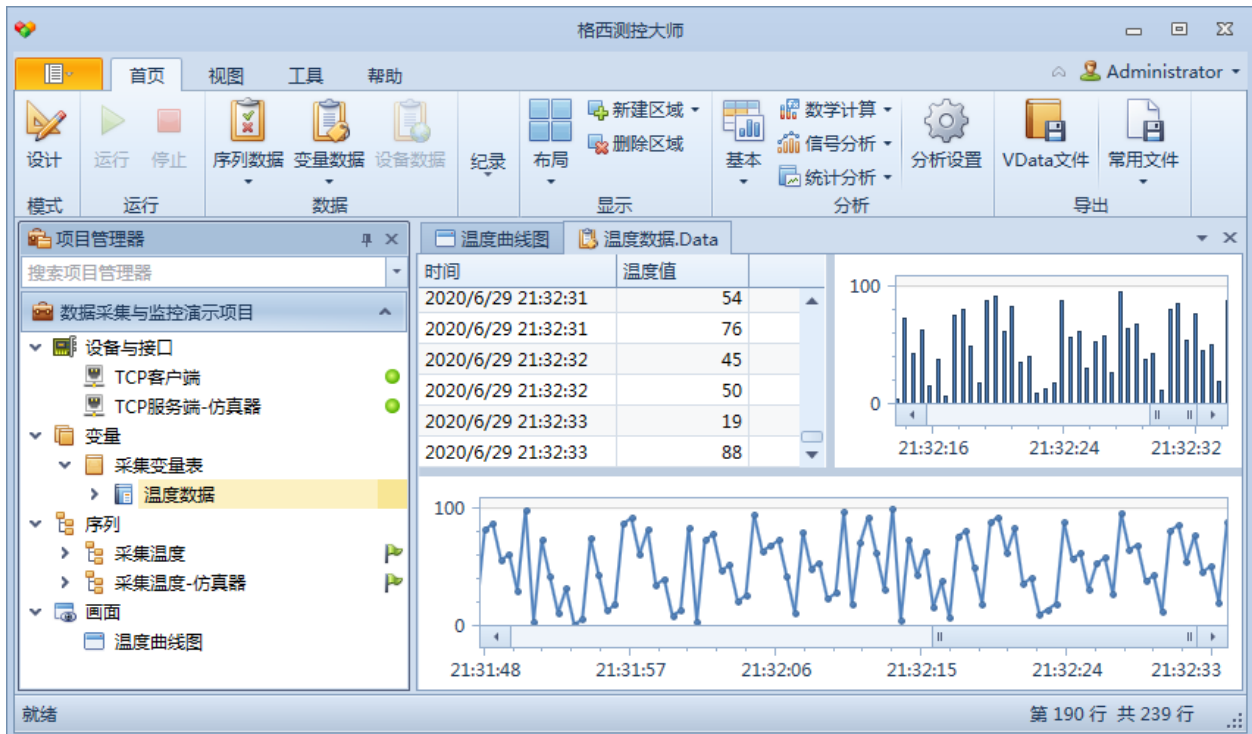
序列数据自动保存到目录 C:\Users\<用户名>\Documents\Geshe\GGenesis\Results 目录中，文件命名格式为<项目名>\_<时间戳>.sdata。





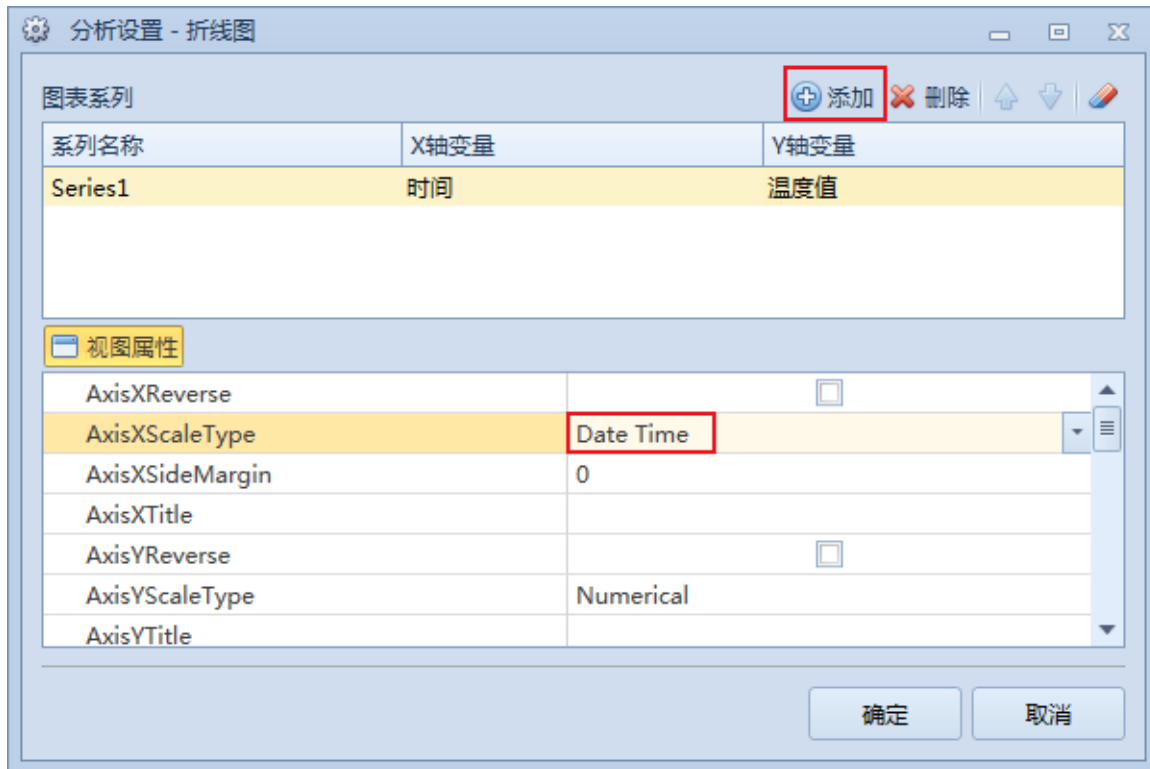
### 3.3 使用变量数据面板采集和监视数据

在项目管理器中选择“变量->温度数据”，点击工具栏的“变量数据->变量数据”按钮，即可打开“温度数据”的变量数据面板。用户可以使用工具栏中的“显示”和“分析”组中的命令按钮，对变量进行分区域进行显示和分析。

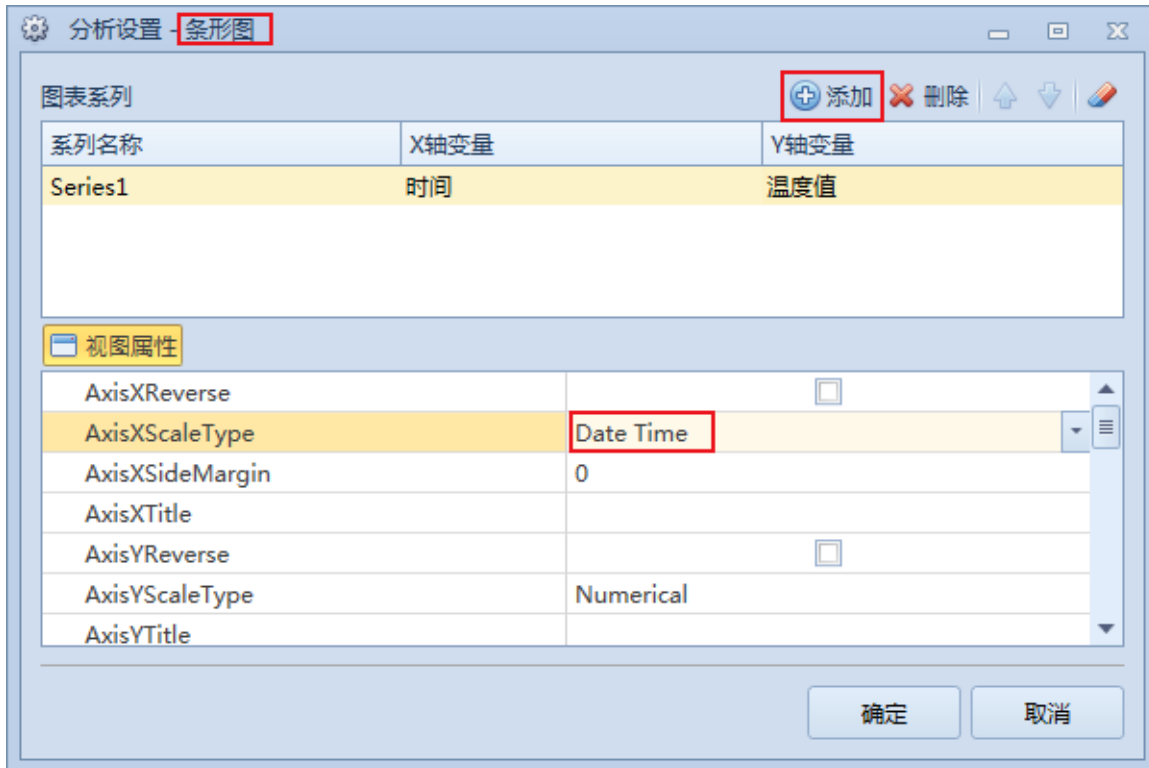


要实现上图所示，面板上部分显示一个数据表、一个条形图，下部分显示一个折线图，则按照下列四个步骤操作：

- 1) 选择工具栏“新建区域->下”，数据面板上下平分
- 2) 用鼠标选中下方区域，选择工具栏“基本->折线图”，弹出折线图设置对话框，按下图设置

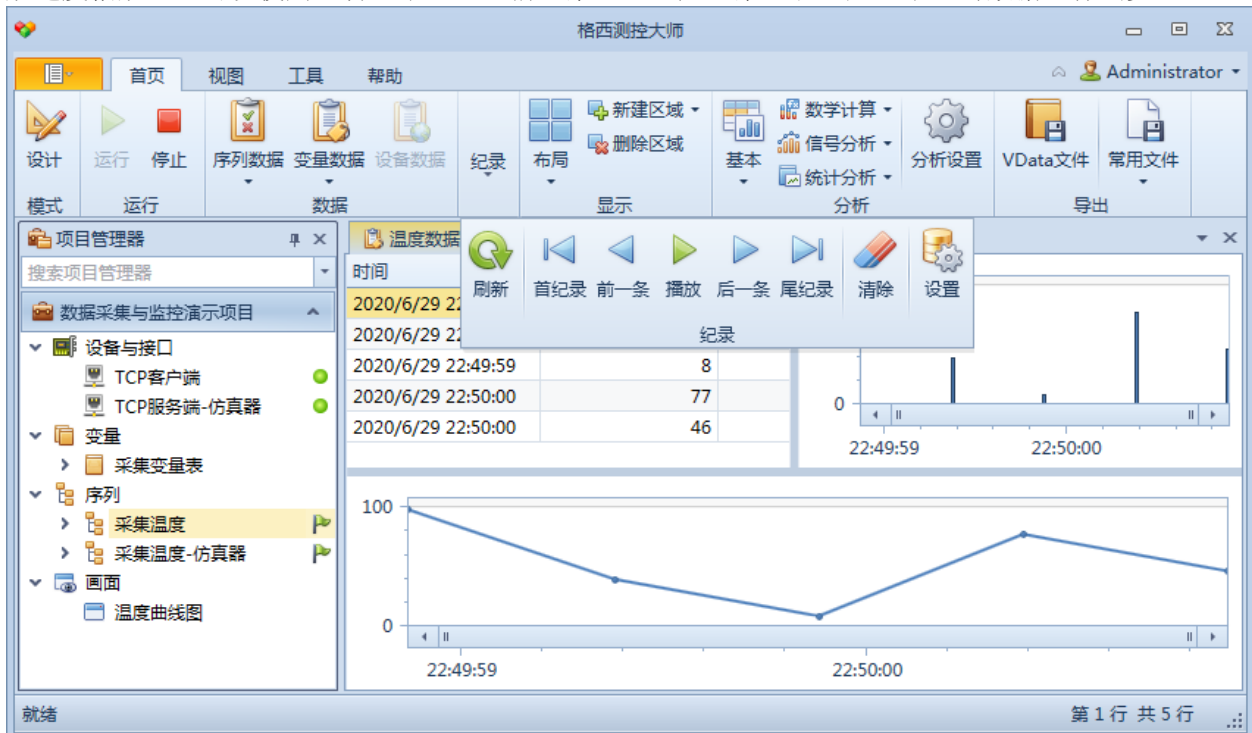


- 3) 用鼠标选中上方区域，选择工具栏“新建区域->右”，数据面板上方区域左右平分
- 4) 用鼠标选中右上方区域，选择工具栏“基本->条形图”，弹出条形图设置对话框并设置



### 3.4 使用变量数据面板回放数据

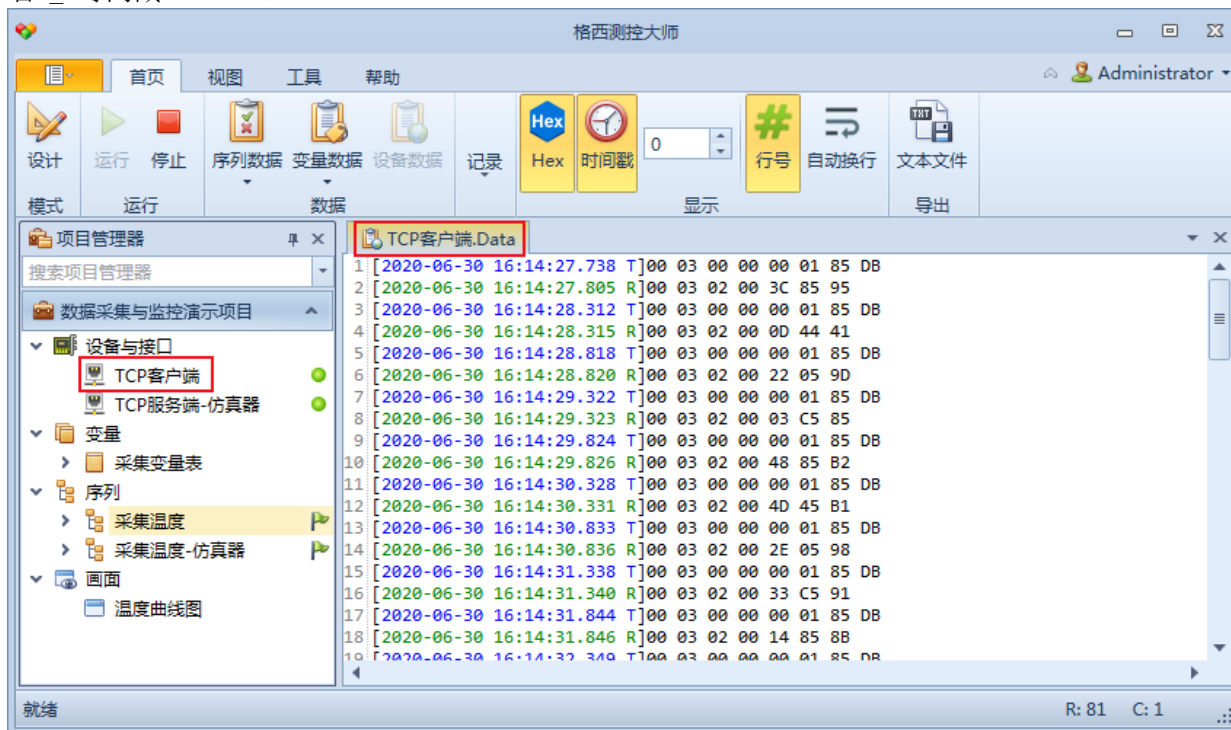
使用变量数据面板，可以对实时变量数据或者历史变量数据进行回放。进行回放操作时，需要先停止“刷新”，即工具栏的“刷新”按钮处于不选择状态。可以点击“播放”按钮自动按照设定的播放速度播放，也可以使用“首纪录”、“前一条”、“后一条”和“尾纪录”对数据进行浏览。



### 3.5 使用设备数据面板监视数据

在项目管理器中选择“设备与接口->TCP 客户端”，点击工具栏的“设备数据”按钮，即可打开“TCP 客户端”这个接口的设备数据面板。

所有和这个接口进行通信的数据都会在这个数据面板中显示，同时，设备数据自动保存到目录 C:\Users\<用户名>\Documents\Geshe\GGenesis\Results 目录中，文件命名格式为<项目名>\_<接口名>\_<时间戳>.ddata。



### 3.6 序列数据的导出和导入

序列数据面板支持将结果数据导出为.sdata文件，支持导出为常用的文本文件、Excel文件、PDF文件等。导出的.sdata文件，可以通过工具栏的“序列数据->历史序列数据”按钮打开。

### 3.7 变量数据的导出和导入

变量数据面板支持将变量数据导出为.vdata文件，支持导出为常用的文本文件、Excel文件、PDF文件等。导出的.vdata文件，可以通过工具栏的“变量数据->历史变量数据”按钮打开。