上海格西信息科技有限公司

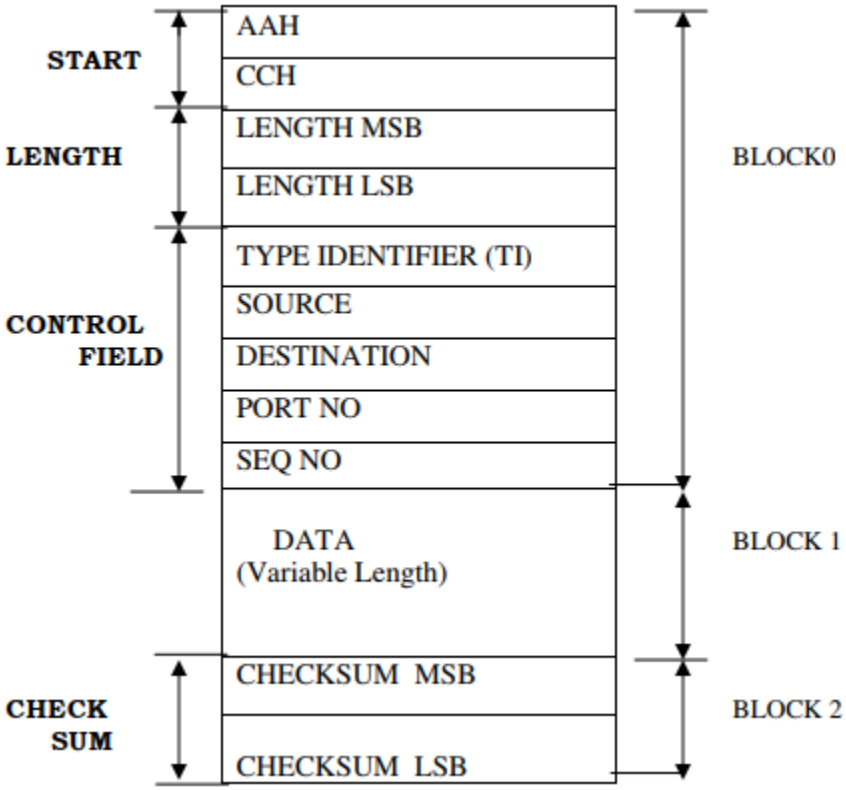
通信协议监听和篡改例子

版本 **0.1**

目录

1. 概述

MASTER 设备和 SLAVE 设备通过串口连接，串口设置默认为波特率 57600，数据位 8，停止位 1，无校验。设备间的通信协议结构和通信命令如下所示。



命令类型 Type-Identifier	名称	描述
80H	Link check	For checking link.
81H	Data Request	Request SLAVE for Data.
82H	Upload result	Ack. To the above Data Request Command.

命令 Link check 请求帧	命令 Link Check 响应帧
BLOCK 0 TI – 80H RECORD LENGTH = 07H	BLOCK 0 TI – C0H RECORD LENGTH = 07H
BLOCK 1 {No DATA BYTE is available}	BLOCK 1 {No DATA BYTE is available}
BLOCK 2	BLOCK 2

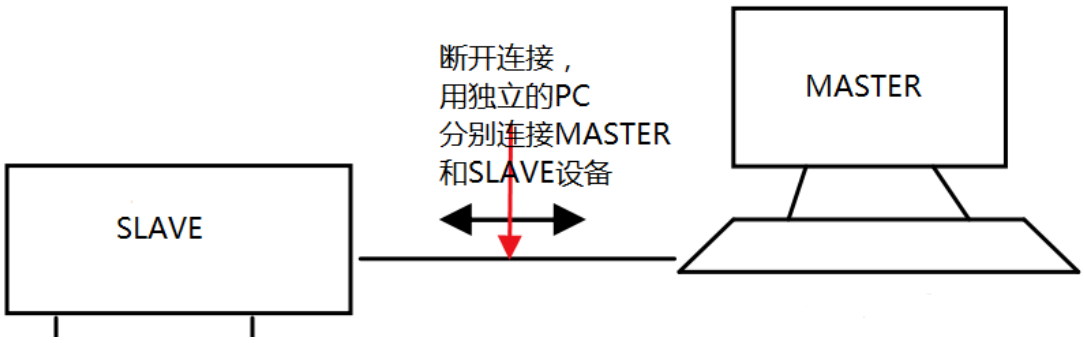
命令 Data Request 请求帧	命令 Data Request 响应帧
BLOCK 0 TI – 81H RECORD LENGTH = 07H	BLOCK 0 TI – C1H RECORD LENGTH = N+07H
BLOCK 1	BLOCK 1

{No DATA BYTE is available}	(N Bytes) {Length is Variable}
BLOCK 2	BLOCK 2

命令 UpLoad Result 帧（无回复）

BLOCK 0 TI – 82H RECORD LENGTH = 07H
BLOCK 1 NO DATA BYTES
BLOCK 2

本演示例子，演示如何监听和篡改 MASTER 设备和 SLAVE 设备之间的通信数据。



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

文件说明：

✓ MessageMonitor.gpj – 通信协议监听和篡改演示项目 – 中文 – 串口版

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，虚拟出两对串口（一对为 COM2 和 COM3，一对为 COM4 和 COM5）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。

2. 创建项目

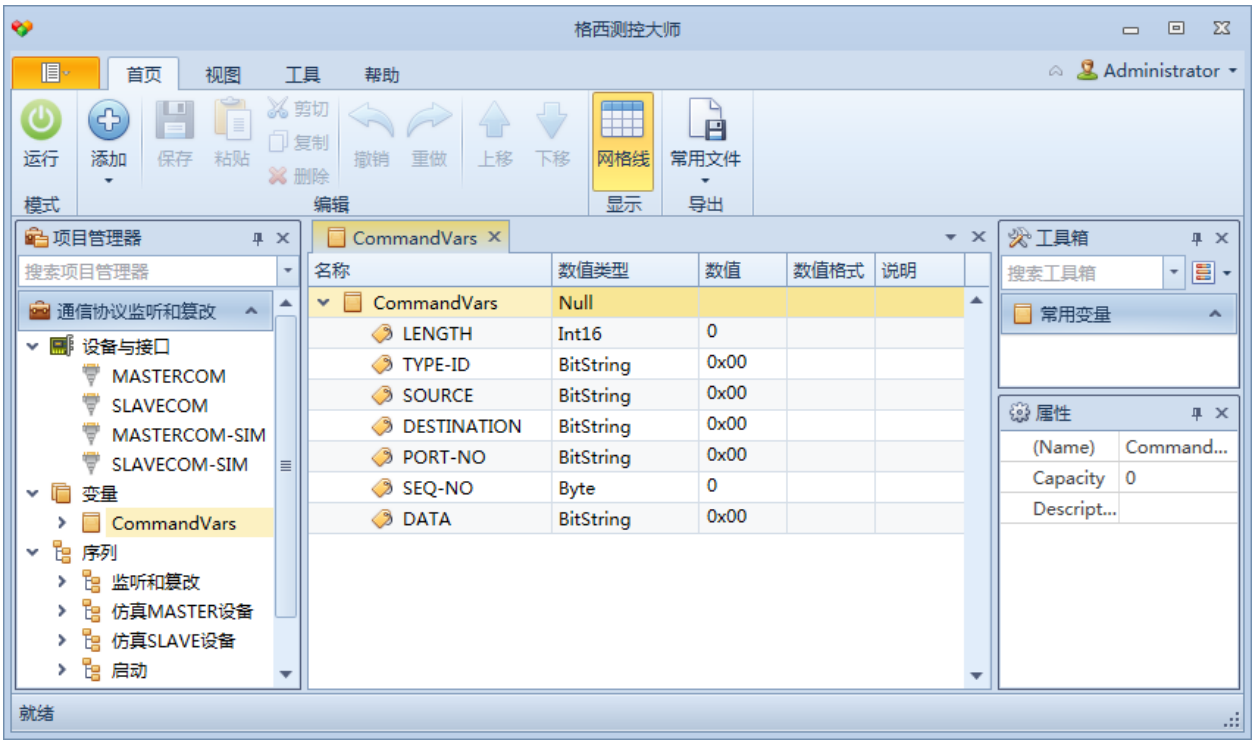
2.1 第 1 步 添加串口设备

本演示项目需要和两台设备连接，分别是 MASTER 设备和 SLAVE 设备，对应“MASTERCOM”和“SLAVECOM”；另外，由于本项目仿真了 MASTER 设备和 SLAVE 设备，分别占用一个串口，对应“MASTERCOM-SIM”和“SLAVECOM-SIM”。

“MASTERCOM”和“MASTERCOM-SIM”为 VSPD 虚拟的一对串口 COM2 和 COM3，虚拟交叉线连接，即 COM2 和 COM3 可以互相通信；“SLAVECOM”和“SLAVECOM-SIM”同理。

2.2 第 2 步 添加变量

建立变量组，保存通信过程的关键数据，作为转发或者篡改的中介桥梁。



2.3 第 3 步 添加序列

本演示项目建立四个序列来实现。

1）“启动”序列：通过脚本实现自动化配置；同时，监听的全局类 MonitorMasterGlobal 也在这里实现。

```
using System;
using Genesis;
using Genesis.Scripting;
using Genesis.Sequence;
using Genesis.Workbench;
using Genesis.Device;

public class Step_C920C3896BD342DEB36C7B7ABA721631
{
  public ScriptContext Context { get; set; }

  //
  public Int32 BeginExecute(IStepContext context, IStep step)
  {
    MonitorMasterGlobal.Reset();
    //
    IDeviceSession dev = this.Context.GetDeviceSession("MASTERCOM");
    if (!dev.State)
    {
      dev.Open();
    }
```

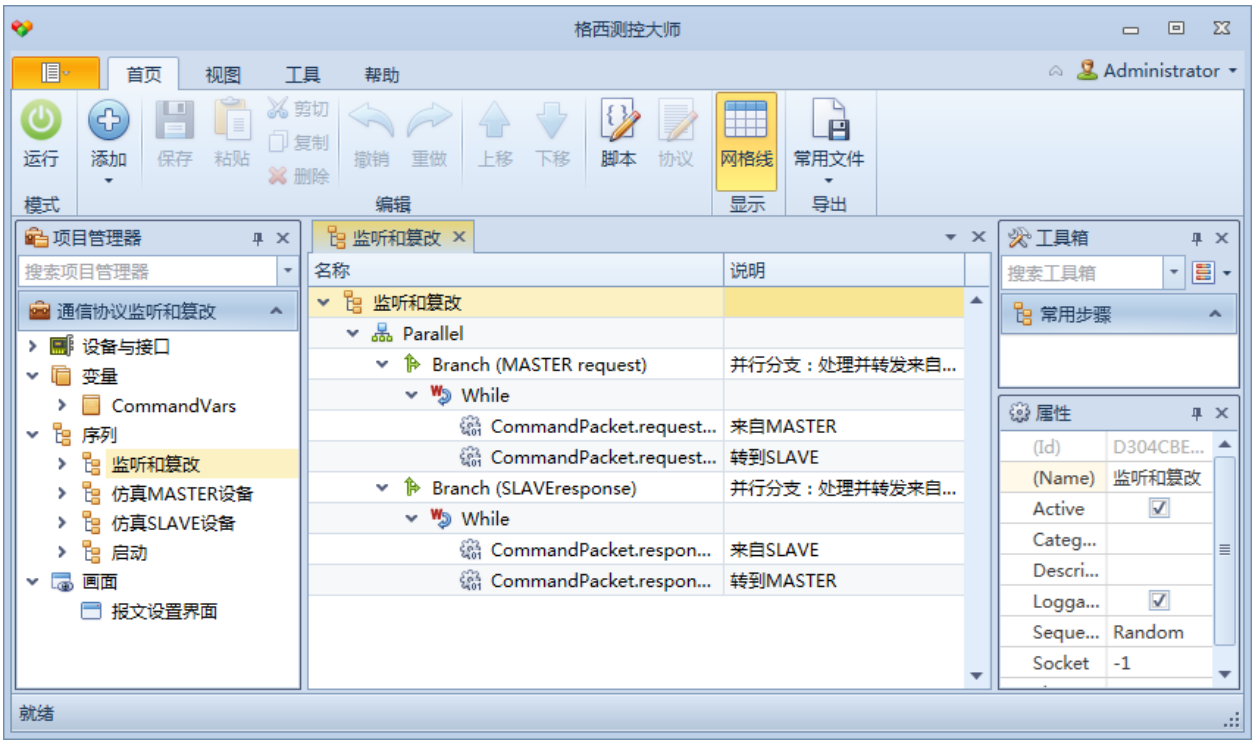
```
    dev = this.Context.GetDeviceSession("SLAVECOM");
    if (!dev.State)
    {
      dev.Open();
    }
    dev = this.Context.GetDeviceSession("MASTERCOM-SIM");
    if (!dev.State)
    {
      dev.Open();
    }
    dev = this.Context.GetDeviceSession("SLAVECOM-SIM");
    if (!dev.State)
    {
      dev.Open();
    }
    // 启动监听和篡改序列和仿真 SLAVE 设备序列
    this.Context.StartStep("监听和篡改");
    this.Context.StartStep("仿真 SLAVE 设备");
    //
    this.Context.OpenStepDataEditor();
    this.Context.OpenSchema("报文设置界面");
    return 0;
  }

  //
  public Int32 EndExecute(IStepContext context, IStep step)
  {
    return 0;
  }
}

// 监听全局配置类，用于 MASTER 转至 SALVE 的报文修改
public class MonitorMasterGlobal
{
  public static Int16 Length {get;set;}
  public static byte Source {get;set;}
  public static byte Destination {get;set;}
  public static byte PortNo {get;set;}
  public static byte SeqNo {get;set;}
  public static string Data {get;set;}

  public static bool CanModifyLength {get;set;}
  public static bool CanModifySource {get;set;}
  public static bool CanModifyDestination {get;set;}
  public static bool CanModifyPortNo {get;set;}
  public static bool CanModifySeqNo {get;set;}
  public static bool CanModifyData {get;set;}
```

```
  public static void Reset()
  {
    CanModifyLength = false;
    CanModifySource = false;
    CanModifyDestination = false;
    CanModifyPortNo = false;
    CanModifySeqNo = false;
    CanModifyData = false;
  }
}
```

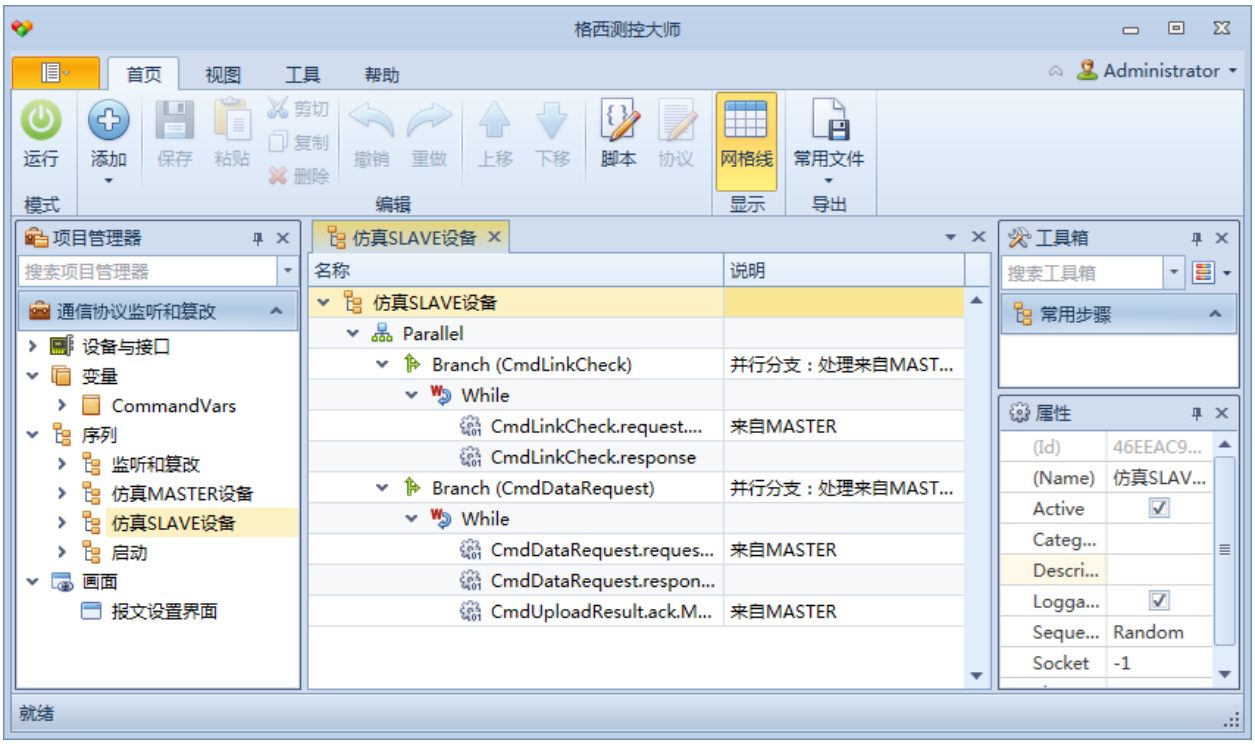
2）“监听和篡改”序列：该序列是核心序列，实现监听、转发和篡改功能。本例子的通信协议只有一个协议结构，故只需要两个并行分支，一个分支处理并转发来自 MASTER 设备的报文，一个分支处理并转发来自 SLAVE 设备的报文。

转发和篡改是以变量为桥梁，通过脚本实现。

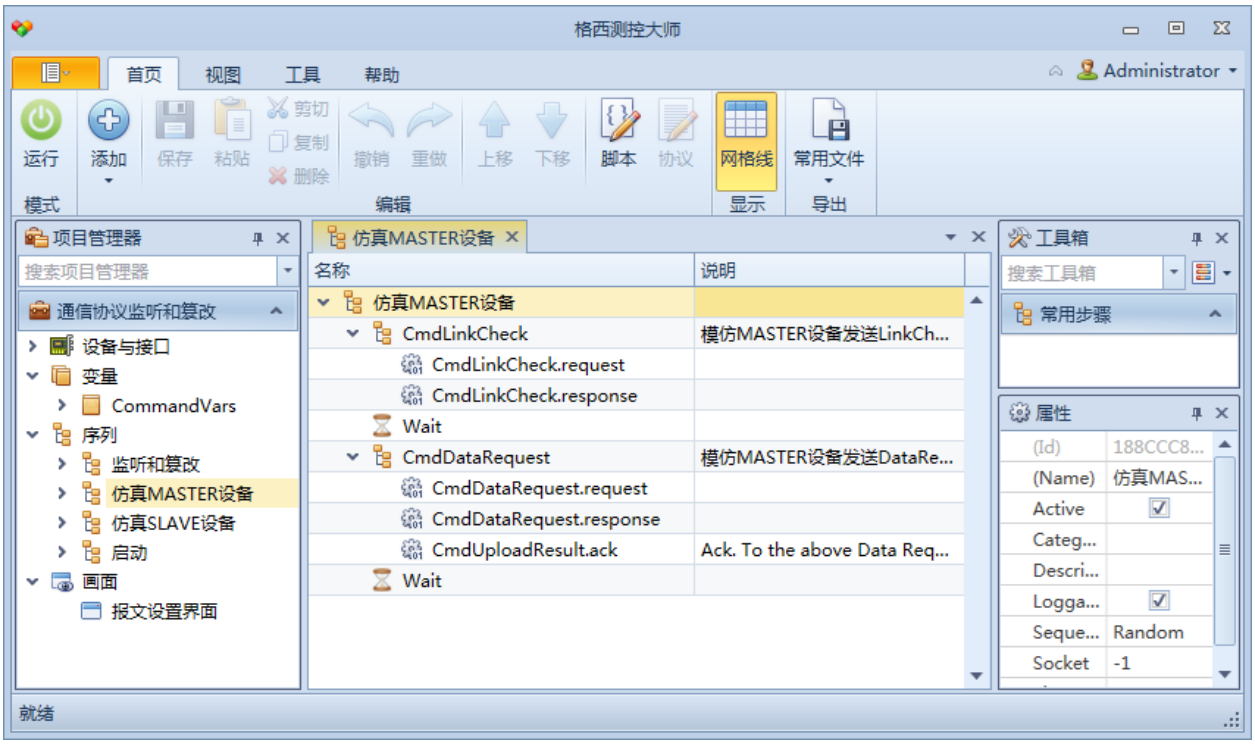
总的流程：收到报文->通过绑定协议字段的方式提取数据字段存入对应的变量->根据篡改配置情况设置转发帧对应的协议字段->转发。



3）“仿真 SLAVE 设备”序列：该序列实现 SLAVE 设备的仿真，需要两个并行分支，一个分支处理来自 MASTER 的 LinkCheck 命令，一个分支处理来自 MASTER 的 DataRequest 命令。



4）“仿真 MASTER 设备”序列：该序列实现 MASTER 设备的仿真，MASTER 设备为主动设备，每个命令单独实现即可。执行的时候，可以单独执行一个命令，也可以连续执行多个命令。



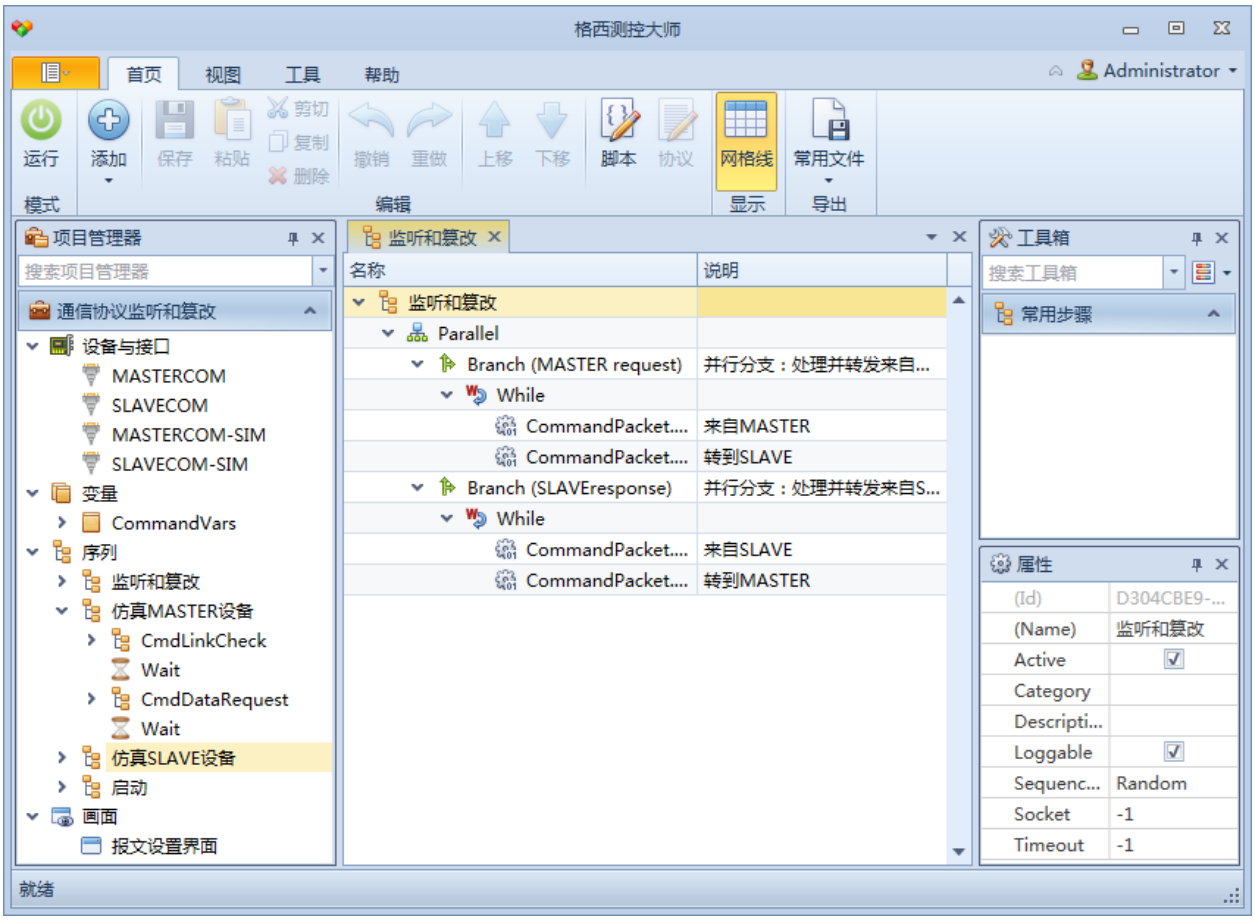
2.4 第 4 步 添加界面

用户画面主要用来配置篡改信息的。通过脚本把参数设置到全局类 MonitorMasterGlobal 中。

3. 运行项目

3.1 打开项目

从<软件安装目录>\Examples\Solutions\MessageMonitor 目录中，打开 MessageMonitor.gpj 串口版项目文件。

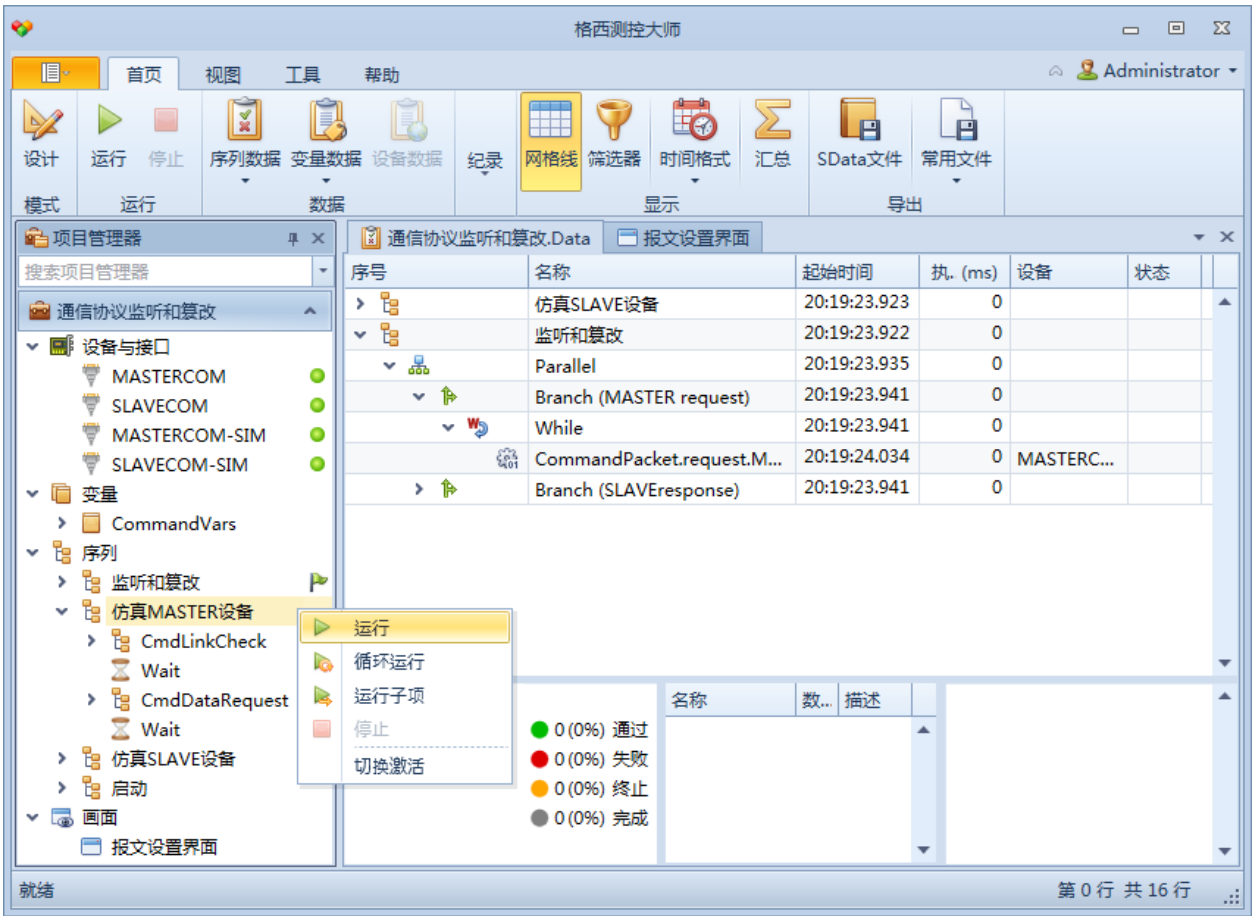


3.2 运行项目

3.2.1 第 1 步 正常运行

点击工具栏的“运行”按钮，进入运行模式，然后点击工具栏“序列数据”按钮，打开序列结果数据页面。

以上准备就绪，即可通过手工运行的方式，运行“仿真 MASTER 设备”序列。用鼠标右键弹出菜单，如下图。



3.2.2 第 2 步 篡改指定 TypeID 的报文并转发

本演示例子只演示如何篡改从 MASTER 设备到 SLAVE 设备的指定报文，一次触发设置修改，只对下一条符合 TypeID 的报文进行修改设置，清空的参数不进行修改设置。

篡改参数设定后，下一次触发的数据，可以到“序列数据”页面查看相关命令的篡改情况，一般篡改后，SLAVE 仿真设备不认得，故不会响应，从而导致 MASTER 侧的收不到响应帧而失败。