

目 录

18. 集成 OPC Client, 及使用步骤	2
18.1 概述	2
18.2 ServerSuperIO 的 OPC Client 使用过程	2
18.2.1 OPC 服务端测试工具, 并创建测点	2
18.2.2 使用配置工具, 配置基本信息和读取的测点	3
18.2.3 打开实时数据库服务	5
18.2.4 运行 UI 程序或控制台程序	5
18.3 注意事项	6
18.4 应用扩展	7
18.5 视频演示	7

官方网站: <http://www.bmpj.net>

18. 集成 OPC Client, 及使用步骤

18.1 概述

OPC 是以 C/S 结构存在且广泛应用在工业领域的协议, 在《[16.集成 OPC Server, 及使用步骤](#)》简要的介绍了 OPC 的基本情况。一般 OPC Server 端向外提供可读数据信息的接口, 例如: ServerSuperIO 的设备驱动读取上来的测点数据可以直接写到 OPC Server 中; OPC Client 端主动连接 OPC Server, 并且通过同步、异步、数据变化的事件等方式读取服务端的数据信息。OPC Server 与 OPC Client 是多对多的关系, OPC Server 可以为多个 OPC Client 提供数据源, OPC Client 也可以读取多个 OPC Server 的数据信息。

OPC Client 作为采集数据信息的一种方式, 所以 ServerSuperIO 集成了此项服务, 它是以服务的形式存在, 有别于设备驱动采集硬件数据的方式。集成 OPC Client 服务只涉及到使用过程, 不涉及到代码开发的问题, 只需要配置就可以把读取过来的数据直接保存到实时数据库中。

18.2 ServerSuperIO 的 OPC Client 使用过程

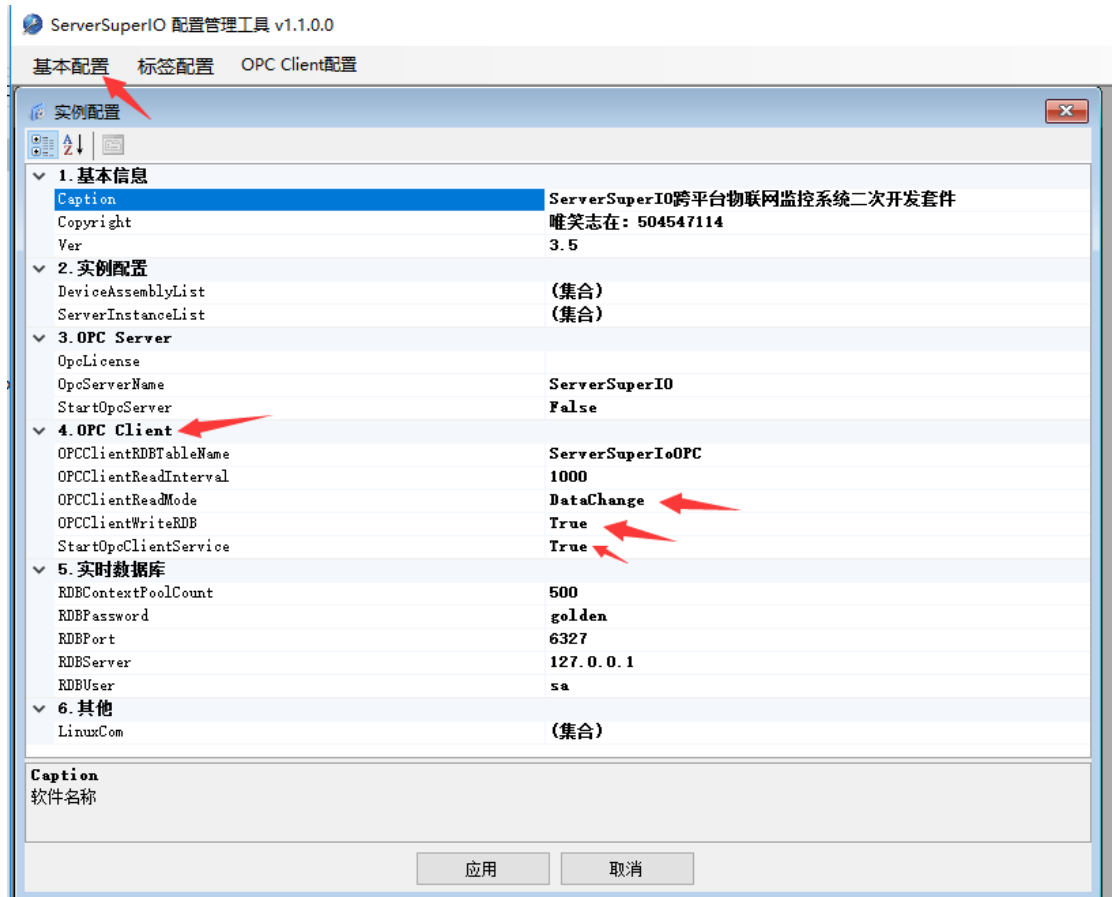
18.2.1 OPC 服务端测试工具, 并创建测点

在 ServerSuperIO v3.5 版本的“OPC 工具和组件\OPC Server 测试工具”目录中可以找到 OPC Server 的测试工具, 打开工具可以创建 OPC 测点及动态生成数据。如下图:

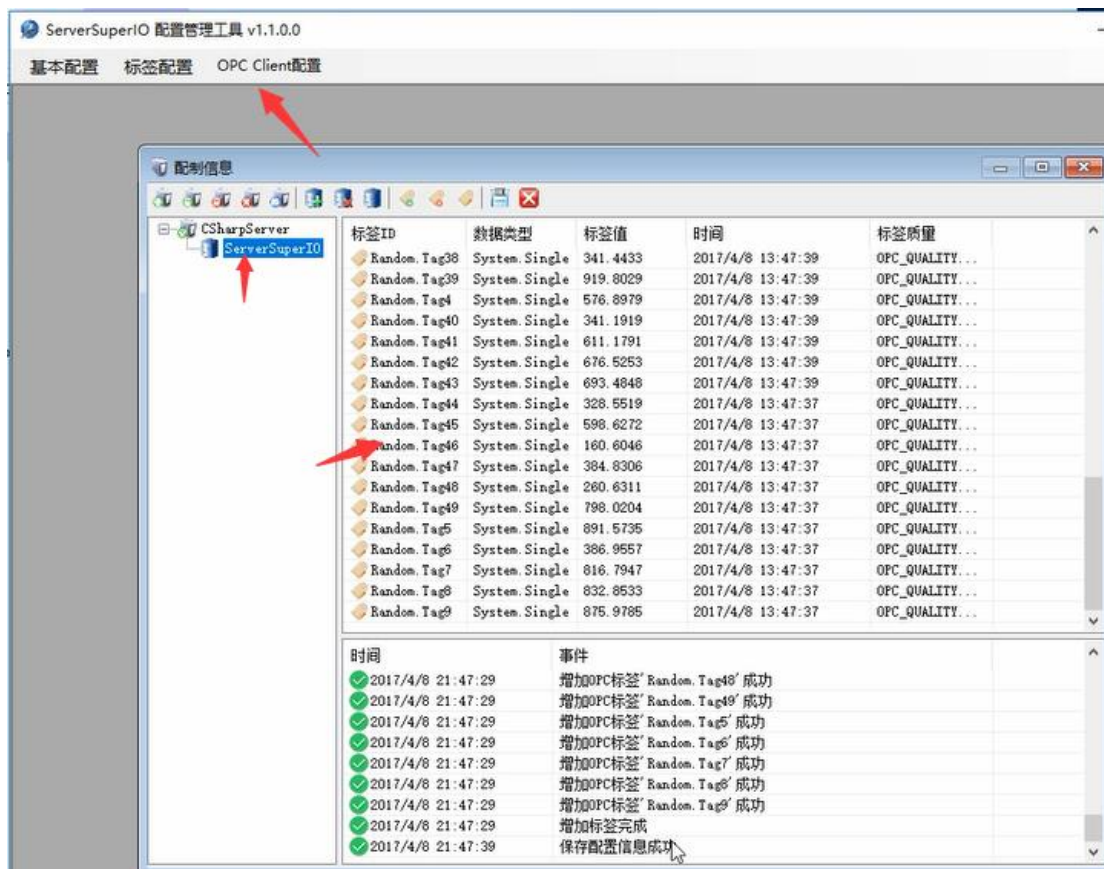


18.2.2 使用配置工具，配置基本信息和读取的测点

需要对【基本配置】菜单的内容进行设置，主要参数包括：（1）写入实时数据库的表名称，可以填写已经存在的表名称。（2）设置读取数据的间隔时间，适用于同步和异步读取数据的方式。（3）设置读取数据的方式，当服务端数据发生变化时自动触发读数据事件、同步读取数据和异步读取数据三种方式。（4）设置是否把读取过来的数据保存到实时数据库，如果需要保存，那么就要对实时数据库的配置信息进行正确设置。（5）是否开启 OPC Client 服务，如果开启的话，那么基于 ServerSuperIO 二次开发的宿主程序会具备此项服务，二次开发套件同样具备这项目服务。



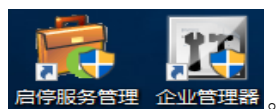
打开【OPC Client 配置】菜单, 把 OPC Server 提供的测点数据按服务、组别和标签的层次关系读取过来, 并对当前的配置信息进行保存, ServerSuperIO 主程序启动后会自动读取数据信息。如下图:



18.2.3 打开实时数据库服务

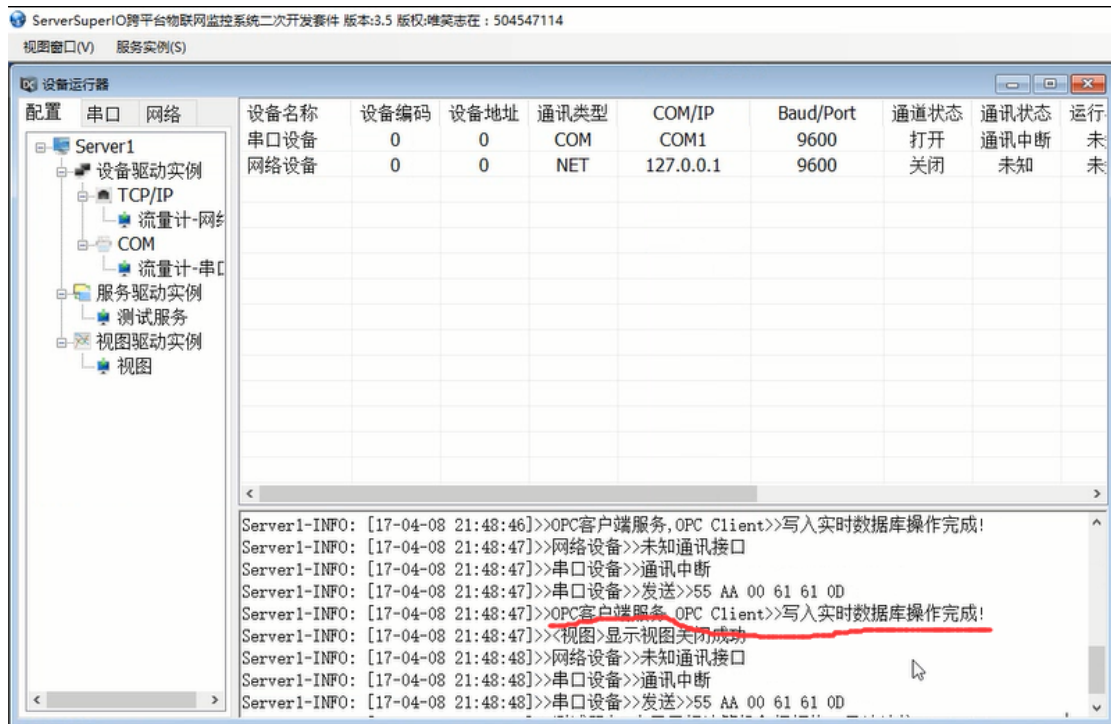
下载“实时数据库安装包及二次开发包.rar”([下载地址](#))并进行安装, 安装

之后有服务管理器和企业管理器, 如下图:

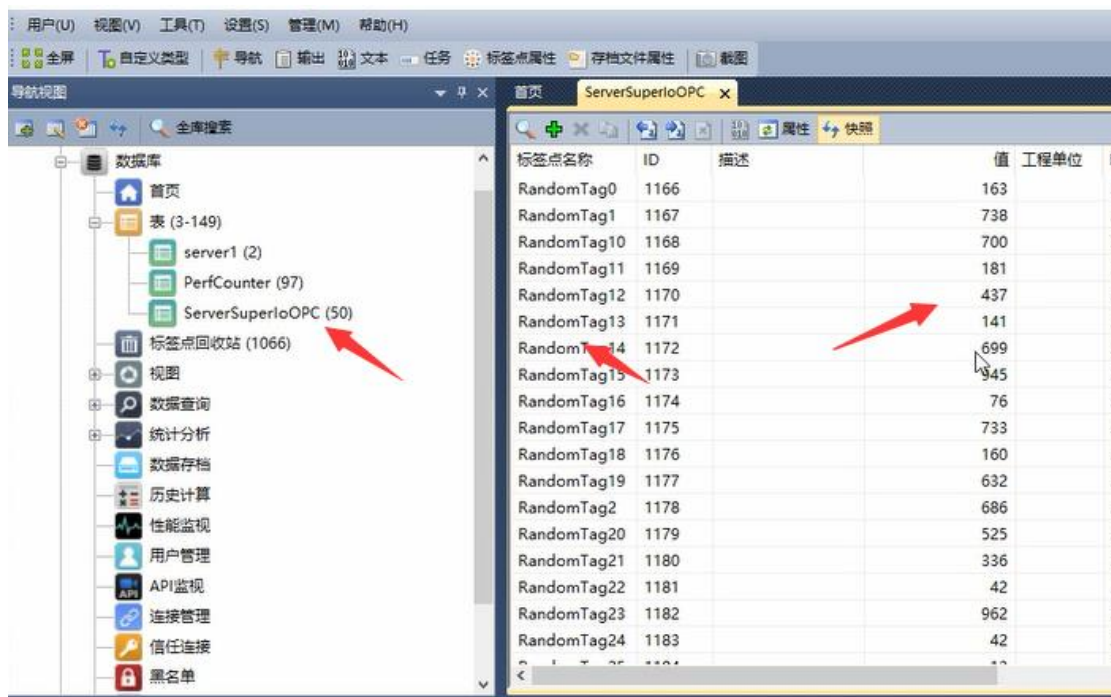


18.2.4 运行 UI 程序或控制台程序

“ServerSuperIO.UI.exe”和“ServerSuperIO.Host.exe”是二次开发套件中的两个主程序, 启动任意一个程序都会自动完成整个流程的设置功能。如下图:



主程序



实时数据库

18.3 注意事项

如果 OPC Server 端创建的测点名称为“Random.Tag1”，那么保存到实时数据

库的名称变为“RandomTag1”，把“.”点去掉了，主要考虑到各组件之间的兼容性的问题。

18.4 应用扩展

OPC Client 本身是一个独立的服务组件，那么一个独立的服务组件是如何与 ServerSuperIO 进行集成呢？框架本身提供 ServerSuperIO.Service. IService 服务接口，把 OPC Client 组件集成到 IService 接口的实体类中，ServerSuperIO 框架会自动实例化 IService 接口对象。

类似其他的服务也可以这样搞，例如：从其他数据源读取数据、模拟量的输出、LED 显示的输出等等。IService 接口的定义如下：

```
public interface IService : IServiceConnector, IPlugin
{
    /// <summary>
    ///     服务 Key,要求唯一
    /// </summary>
    string ServiceKey { get; }

    /// <summary>
    ///     服务名称
    /// </summary>
    string ServiceName { get; }

    .....
}
```

18.5 视频演示

<https://v.qq.com/x/page/r0391ggw7df.html>