

基于Winform框架的OPC UA通信方法研究与实现

陈 刚, 李洪强, 叶志坚, 周 闯, 刘小华

武汉船用机械有限责任公司, 湖北 武汉

收稿日期: 2023年2月21日; 录用日期: 2023年3月21日; 发布日期: 2023年3月29日

摘 要

本文提出并实现了一种基于C#和Winform框架的OPC UA通信方法与系统。介绍了OPC UA通信协议的工作原理, 利用基于C#语言的OPC UA开源客户端OPCUaHelper开源库, 作为监控系统中OPC UA客户端开发工具, 实现对西门子OPC UA服务器的访问。采用消息订阅机制实现对OPC服务器状态数据的实时读取, 同时利用数据改变触发事件机制实现控制数据的实时写入。最后, 该通信方法在化学品船智能液货系统中进行了试验验证, 数据通信实时性满足项目要求。

关键词

Winform, OPC UA, 消息订阅, 事件

The Research and Implementation of OPC UA Communication Method Based on Winform Framework

Gang Chen, Hongqiang Li, Zhijian Ye, Chuang Zhou, Xiaohua Liu

Wuhan Marine Machinery Plant Co., Ltd., Wuhan Hubei

Received: Feb. 21st, 2023; accepted: Mar. 21st, 2023; published: Mar. 29th, 2023

Abstract

This paper proposes and implements an OPC UA communication method and system based on the C# and Winform framework. The working principle of the OPC UA communication protocol is in-

文章引用: 陈刚, 李洪强, 叶志坚, 周闯, 刘小华. 基于 Winform 框架的 OPC UA 通信方法研究与实现[J]. 仪器与设备, 2023, 11(1): 86-93. DOI: 10.12677/iae.2023.111012

roduced, and the OPCUaHelper open source library of the OPC UA open source client is used based on C# language as the development tool of the OPC UA client in the monitoring system to realize the access to Siemens OPC UA server. The real-time reading of OPC server status data is realized by the message subscription mechanism and the real-time writing of control data is realized by the data change triggering event mechanism. Finally, the communication method is tested and verified in the intelligent liquid cargo system of the chemical tanker, and the real-time data communication meets the project requirements.

Keywords

Winform, OPC UA, Message Subscription, Event

Copyright © 2023 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

随着船舶智能化发展, 化学品船装卸控制系统越来越趋于信息化、智能化。化学品船装卸控制系统的上位机软件, 作为系统功能的最终展示界面, 不仅需要集成阀门遥控、液位遥测、集成控制、自动控制等各个模块的控制与反馈数据, 还需要对控制指令及反馈数据有较好实时性响应要求。这些功能对化学品船装卸控制系统的通信模块提出了极高要求, 既能够实现与多个不同厂家硬件通信的要求, 还需要满足数据通信的实时性要求。

基于以上要求, 本文提出了一种基于 Winform 框架的 OPC UA 通信方法[1] [2] [3] [4], 基于 Visual Studio 2019 开发平台, 采用 C#及 Winform 开发框架[5], 设计开发 OPC UA 客户端通信模块, 实现了化学品船装卸控制系统高速、高效的通信模块设计与开发, 并进行了试验验证。结果表明, 采用的通信协议及数据读写机制, 能够对多个不同厂家、不同驱动程序设备的数据统一采集, 实现了化学品船智能液货系统上位机应用程序与控制器 PLC 的高效、实时数据交互, 为后续化学品船智能液货系统上位机中各种监控功能的开发奠定集成。

2. 系统架构

2.1. 化学品船装卸控制系统功能架构

化学品船装卸控制系统框架如下图 1 所示。先通过上位机制定装卸货作业时序计划并将制定的装卸货作业计划发送至下位机, 装卸货作业时序计划为各货舱的装卸货顺序及装卸货量、各压载舱的压排顺序及压排海水量, 下位机按照装卸货作业时序计划制定各个货舱和压载舱在不同时序阶段下的控制逻辑, 并将控制逻辑下发至各个货舱和压载舱中的控制执行设备, 由控制执行设备完成自动化装卸货。

装卸货作业时序计划的制定步骤为, 先由上位机中搭载的专家系统根据前三载化学品的化学性质以及人工输入的约束条件生成多个配载方案, 并确定多个配载方案下的船舶姿态, 然后通过上位机选取船舶姿态最好的配载方案作为最优配载方案, 对最优配载方案进行人工校核, 通过人工校核后以该最优配载方案作为装卸货作业时序计划输出至下位机, 约束条件包括装货/卸货、货物种类、货量、码头允许的扬程和流量, 该设计一方面根据装卸货需求, 按照先后顺序给出货舱的目标装卸量和压载舱的压载量,

同时满足了船舶姿态稳定、船体结构不受伤害的要求，另一方面，经人工确定后再将得到的装卸货作业时序计划输出至下位机，人机交互性好。

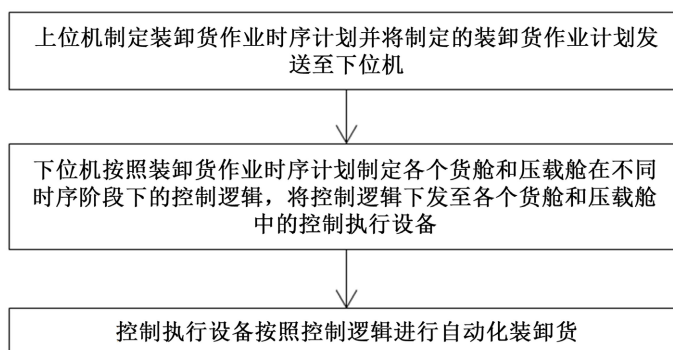


Figure 1. Functional architecture diagram

图 1. 功能架构图

以上系统中，上位机与下位机的通信采用 OPC UA 协议进行实现。

2.2. 通信模块软件框架

如下图 2 所示，基于 OPC UA 通信接口，完成化学品船智能液货集成控制系统应用程序与 PLC 控制器的数据交互，最终实现对货品状态数据的采集以及泵操作阀门操作等动作控制。



Figure 2. Communication module framework

图 2. 通信模块框架

化学品船智能液货集成控制系统应用程序要正常工作，对 OPC UA 通信模块提出以下要求：

- 1) 整个系统的数据量大概有 1 万个点，需要实现对这些数据的高效实时读取；
- 2) 在满足数据采集的同时，软件界面的控制指令需要通过通信模块，及时地发送至被控对象，延时时间不能超过 500 ms。

3) 为通信系统的冗余性,本项目配置的 2 个 OPC UA 服务器,客户端软件需实现 OPC UA 通信的短线重连及自动切换功能,以保证通信的稳定性。

3. OPCUA 客户端开发

3.1. OPCUA 客户端插件安装

利用基于 C#语言的 OPCUA 开源客户端 OPCUaHelper 开源库,作为本项目中 OPCUA 客户端开发工具,实现对西门子 OPCUA 服务器的访问。OPCUaHelper 是基于 OPC UA 协议的开源 OPC UA 客户端组件,提供一个 opc ua 的二次扩展类,基于 .net 4.6.1 创建,基于官方 opc ua 基金会跨平台库创建,封装了节点读写,批量节点读写,引用读取,特性读取,历史数据读取,方法调用,节点订阅,批量订阅等操作。还提供了一个节点浏览器工具。

3.2. 建立服务器连接

采用异步方式连接 OPC UA 服务器,核心代码如下。实例化 1 个 OPC UA 客户端类 OpcUaClient,制定 OPC UA 服务器的 IP 地址及端口号,通过异步方式建立与 OPC UA 服务器连接。

```
public string connectString = "opc.tcp://192.168.16.206:55105";
public OpcUaClient opcUaClient;
public async void OpcUa_Connect()
{
    opcUaClient = new OpcUaClient();//
    errTxt = "OPC 服务器联接成功";
    try
    {
        await opcUaClient.ConnectServer (connectString);
    }
    catch (Exception ex)
    {
        errTxt = "OPC 联接失败: " + ex.Message;
    }
    CommonVariable.uaopcConnected = true;
}
```

4. OPC UA 客户端读写操作实现

4.1. 消息订阅读取 OPC 服务器

为实现对大量数据的实时读取,需尽可能实现有效读取,即只要数据发生改变时才进行读取,数据没有发生改变或者改变量较小时不进行读取通信,这样可以极大地利用带宽,实现大量数据的实时读写,该数据读写机制就是数据的发布/订阅机制。

所谓发布/订阅机制是一种以基于内容的消息传输方式,每个消息都有一个主题(Topic),网络中的客户可以发布和订阅某一主题的消息,一旦发布了某一个主题的消息,所有订阅者都将会及时收到该主题的消息,数据流向如下图 3 所示。

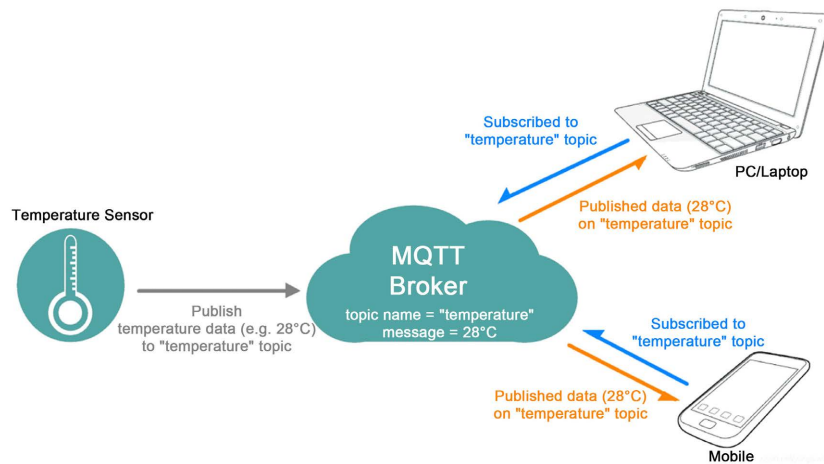


Figure 3. Data flow for publish/subscribe mechanisms
图 3. 发布/订阅机制的数据流向

1) 获取 OPC UA 中需要订阅数据的变量路径

要订阅变量节点,就需要获取 OPCUA 服务器中变量节点的绝对路径,如下图所示,主要包括 DB112、DB114、DB115、DB200、DB201、DB202、DB203 等数据块共计 2400 个变量需要订阅。

```
public Opcua()
{
    DB112 OPCUAData.TagNameSet();
    DB114 OPCUAData.TagNameSet();

    DB111MonitorNodeTagsBooleanList = DB111 OPCUAData.BoolListTagNameSet();
}
```

2) 批量节点订阅

节点路径得到后,就可以通过 OPCUA 客户端实例 `opcUaClient` 进行订阅了,如下图所示,将所有需要读取的变量进行批量订阅。

```
public void subMultiNode()
{
    // 多个节点的订阅
    opcUaClient.AddSubscription("DB111BooleanList", DB111MonitorNodeTagsBooleanList, MultiSubCallback);
}
```

3) 实时监控数据变化及读取操作

一旦完成了数据订阅,只要被订阅的任何一个数据发送变化,将触发读取订阅数据的回调函数,实时读取被订阅变量的最新值。同时,数据读取模式在数据没有变化时不会触发任何代码,不仅大大节省了带宽、提高通信效率,还为监控系统的实时性提供了底层支撑。

4.2. 事件触发写入 OPC 服务器

为保证控制指令下发的实时性,只要有新的控制指令产生,立即触发写入事件,将新的控制指令相关数据即可写入到 OPCUA 服务器中,以保证控制数据写入的实时性。

1) 获取 OPC UA 中需要写入数据的变量路径

同理,要实现写入,就需要获取 OPCUA 服务器中数据吸入变量节点的绝对路径,如下图所示,写入数据主要分布在 DB112、DB114 数据块中

```

public Opcua()
{
    DB112 OPCUAData.TagNameSet();
    DB114 OPCUAData.TagNameSet();
    public static string[] TagNameSet()
    {
        for (int i = 0; i < DB112_DataNames.Length; i++)
        {
            DB112_DataNames[i] = DB112_DataNamesPrefix + DB112_DataNames[i];
        }
        return DB112_DataNames;
    }
}
public class DB112 OPCUAData
{
    private static string DB112_DataNamesPrefix = "ns=6;s=S7-1500RH-System_1.PLC_1.DB112_SystemGloableData.";
    public static string[] DB112_DataNames = new string[]
    {
        "Auto_Control_Command", //0
        "Intergrated_Control_Command", //1
    }
}

```

2) 监听类属性变化触发事件

利用开源插件 Fody 以及 PropertyChanged.Fody，来监听类的属性(即需写入数据的变量)，一旦类的属性发送值改变，立即触发 PropertyChanged 事件，在该事件函数中可定义相关的操作，如向 OPCUA 服务器中写入最新数据。如下图所示。

```

//综合监测数据
public class DB112ParameterSetting : INotifyPropertyChanged
{
    public event PropertyChangedEventHandler PropertyChanged;
    public int Id { get; set; }
    public DateTime Timestamp { get; set; }
}

```

3) 数据写入 OPCUA 服务器

以 DB112 数据块中类的属性 AutoControlCommand 为例，一旦界面中产生了新的与上一次不一样的 AutoControlCommand 指令，程序自动触发 OnAutoControlCommandChanged 事件，并在该事件中执行 OPCUA 的写入方法 writeNode，将最新的 AutoControlCommand 数值写入到 OPCUA 服务器中。

```

private void OnAutoControlCommandChanged()
{
    string nodeIdTag = DB112 OPCUAData.NodeIdTagGet("AutoControlCommand");
    CommonVariable.opcua.writeNode(nodeIdTag, AutoControlCommand);
}

```

5. 验证

智能液货集成控制系统应用程序界面如下图 4 和图 5 所示，主要包括对左舷 6 个货舱、右舷 6 个货舱共 12 个舱的装货及卸货控制。根据统计，通信数据大概有 1 万个，系统连续运行 7 × 24 小时，测试结果表明，该通信模块既能对大量数据实时读取，也能实时响应控制指令实现对被控对象的即时控制。OPC UA 通信模块的读写通信速度及实时性满足项目工程要求。

6 总结

基于化学品船智能液货系统上位机软件的通信需求，本文设计并开发了基于 C# 及 Winform 平台设计的 OPC UA 客户端通信模块。为实现对多达 1 万个数据点的实时读取，采用基于发布/订阅机制的数据读取机制，在极大地利用带宽的同时，实现大量数据的实时读写。同时，为保证控制指令的响应实时性，

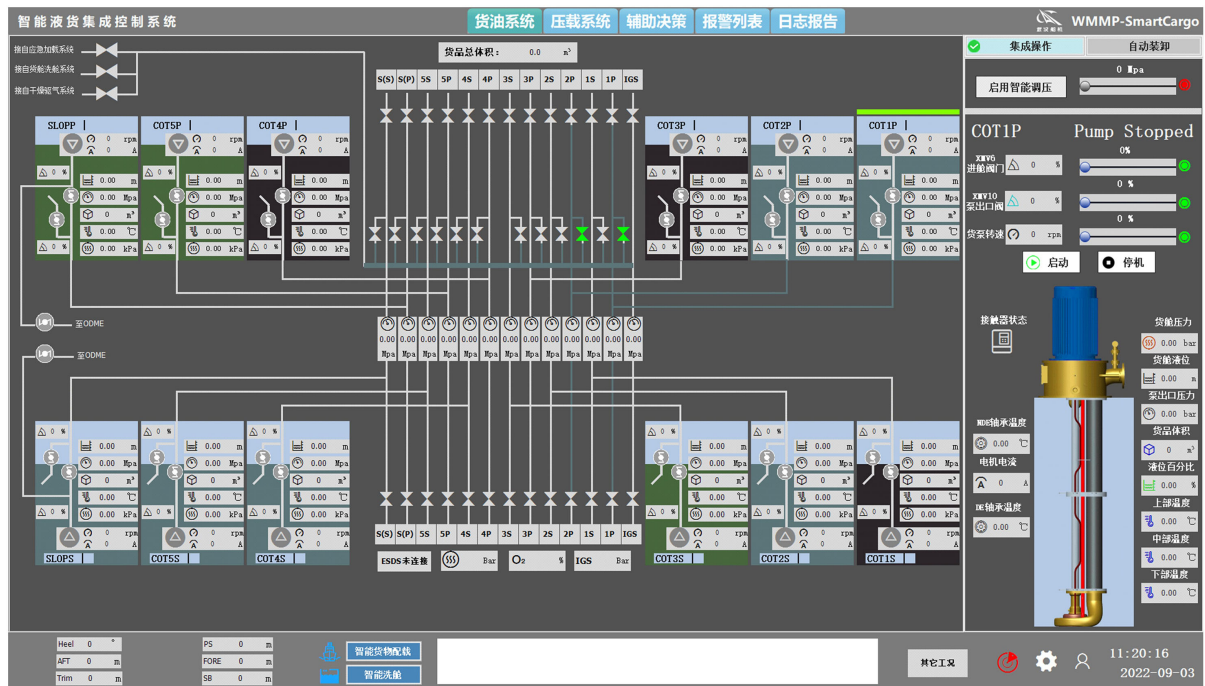


Figure 4. Cargo pump system monitoring interface
图 4. 货油泵系统监控界面

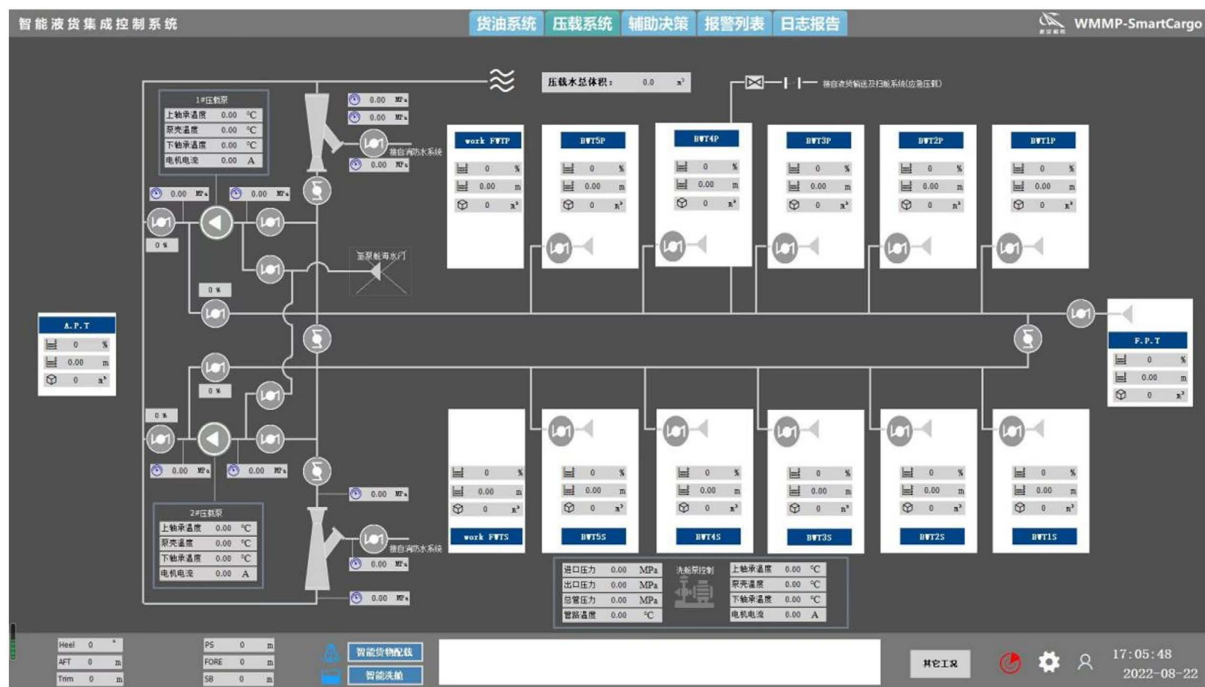


Figure 5. Ballast pump system monitoring interface
图 5. 压载泵系统监控界面

采用数据改变事件触发机制，一旦有新的控制指令马上，系统立刻触发事件，进行控制指令数据的下发。最后，在化学品船智能液货系统中进行了应用验证，验证结果满足设计需求。

参考文献

- [1] 石朝阳, 徐世许, 史世良. 基于 OPC UA 技术的物料搬运监控系统设计[J]. 制造业自动化, 2022, 44(3): 59-62.
- [2] 郑斯琪, 沈斌. 基于 OPC UA 的污水泵站监控系统设计[J]. 机电一体化, 2021, 27(6): 19-24.
<https://doi.org/10.16413/j.cnki.issn.1007-080x.2021.06.003>
- [3] 顾小红, 胡洋, 陈旭, 韩毅, 钱陈虎, 张辉, 彭斌. 基于 OPC UA 的电厂水务数据采集系统设计开发[J]. 工业控制计算机, 2022, 35(6): 114-115+118.
- [4] 范晋伟, 杨跃, 潘日, 李伟华. 基于 OPC UA 的数控机床远程监控系统开发[J]. 机床与液压, 2022, 50(18): 44-48+54.
- [5] 周雯. 基于 WinForm 的移动基站设备管理系统的设计与实现[D]: [硕士学位论文]. 成都: 电子科技大学, 2014.