

基于.NET Core 架构的 PLC 数据采集和监控系统^①

禹鑫燚^② 唐权瑞 殷慧武 施甜峰 王煦焱 欧林林^③

(浙江工业大学信息工程学院 杭州 310023)

摘 要 为了解决工厂设备品牌多样化、通讯协议各不相同、采集难度大等问题,研究了基于.NET Core 架构的远程数据采集和监控系统。该系统借鉴了异构的数据存储方式、分布式网络模型和数据分析模型,使用.NET Core 框架开发完成。首先将嵌入式工控机和 PLC 设备通过工业以太网连到同一局域网并进行通信,对可编程逻辑控制器(PLC)设备的运行状态进行实时数据采集,并上传到云端服务器,Web 端可以远程查看 PLC 设备的实时数据,以及对 PLC 设备的工作参数进行数据分析,当分析结果显示设备出现故障时,云端服务器通过短信通知现场工程师 PLC 设备出现故障,当现场工程师无法调试设备时,远程工程师还可以通过 Open VPN 对设备进行远程调试。最后通过实验验证了系统的可行性。

关键词 可编程逻辑控制器(PLC); .NET Core; 监控; 数据采集; 分布式; 异构数据存储

0 引 言

随着制造业技术水平的不断发展,制造业在全球范围内出现了产能过剩的问题,德国首先于 2011 年提出了“工业 4.0”的概念。随后美国提出了“工业互联网”、中国提出了“中国制造 2025”等战略。虽然在称呼上有所差别,但在本质上都是为了提升制造业的智能化水平^[1-4]。在制造业领域,可编程逻辑控制器(programmable logic controller, PLC)扮演着十分重要的角色,但在许多生产车间,根据生产需求的不同,可能存在多种品牌的 PLC 设备,不同品牌 PLC 设备的通讯接口各不相同,即所支持的通讯协议不同,这给数据采集带来了很大的困难,同时大部分老式工厂设备的历史数据还依赖人工抄写,并且设备出现问题现场无法检修时还会造成很大的麻烦,这会给工厂推进智能化进程加大难度。

国内外很多学者对 PLC 数据采集和监控系统

进行了研究,文献[5]和[6]先后提出了云制造模式或智能工厂模式。在 PLC 设备的数据采集技术部分,文献[7]通过串口对西门子 S7-200PLC 进行数据采集。文献[8]对西门子 S7-300 PLC 进行数据采集,利用 PROFIBUS-DP 通信接口、CP5611 通信卡与 Win CC 组态软件实现了对 PLC 控制系统的数据采集及监控。此外,文献[9]开发了系统数据采集和处理软件,在 VB 6.0 的环境下,设计了监控界面,该系统已实际应用于船舶机舱的监测与报警。文献[10]设计了温室大棚环境智能检测与调节系统,通过调整温度、湿度和 CO₂ 浓度,使作物处于最佳生长环境,具有参数设置、数据采集、数据保存和历史数据查询等功能。文献[11]设计了 PDA 高速数据采集分析系统,并将其从站协议集成到普通以太网卡中。文献[12]研究了基于 B/S 模式的 3 种 PLC 远程监控系统的系统结构、技术特点和实现方法,并与专业教学相结合开发了实践案例,文献[13]针对大型分布式工业控制系统中 PLC 控制

① 国家重点研究发展计划(2018YFB1308000)和浙江省自然科学基金探索(LY21F030018)资助项目。

② 男,1979 年生,博士;研究方向:机器人控制与规划;E-mail: yuxinyinet@163.com

③ 通信作者,E-mail: linlinou@zjut.edu.cn

(收稿日期:2020-01-12)

器位置分散等实际情况,实现了控制系统内数据的分析和处理。

从目前的研究成果来看,所设计的数据采集系统大多基于现场总线搭建一个局域网,使用 PC 机或嵌入式工控机对 PLC 设备进行数据采集,许多数据并没有存储到云端数据库,导致远程无法监控工厂设备的实时数据。而随着制造业技术水平的不断发展,开发一款适合工业现场的 PLC 设备数据采集和监控系统有很大的应用价值。此外在现有的产品中,对 PLC 设备进行远程调试的研究过少,对 PLC 设备的实时数据进行分析,判断其是否发生故障的研究也比较少,而远程控制和实时数据分析是智能工厂生产制造过程中的重要部分,因此,对设备工作参数进行实时分析和远程控制对推进制造业智能化起着非常重要的作用。

本文在此研究背景下,设计了基于 .NET Core 架构的 PLC 数据采集和监控系统的总体框架设计,根据系统的功能需求开发了 B/S 和 C/S 结构的交互控制软件,实现了对工厂 PLC 设备的远程控制,可在本地端和网页端实时显示 PLC 设备的运行状态,并且将采集到的数据进行数据分析,预测工厂 PLC 设备是否会出现故障。通过实验对所设计的 PLC 数据采集和监控系统进行了验证。

1 系统需求及总体框架设计

本文开发的基于 .NET Core 架构的 PLC 数据采集和监控系统,主要解决工厂内设备的工业协议标准不统一且数据开发性不够、工业数据采集实时性不足、数据采集存在着数据安全隐患等问题。基于数据采集和监控系统的经典设计思想,把本系统设计分成硬件系统构成和软件系统设计,系统的总体框架如图 1 所示。系统的硬件构成主要围绕使所有的接入设备位于同一网段、软件系统运行所需要的开发环境、硬件设备连接等需求,进行了嵌入式工控机的选型、系统环境搭建、硬件设备连接等操作。

为了让软件系统完全兼容 Linux-Arm 操作系统,采用了 .NET Core 3.0 和 Win 64 作为软件系统的开发框架,使用 C#编程语言完成数据采集的开

发。考虑到远程监控、数据显示和数据分析等功能,系统采用了消息队列遥测传输(message queuing telemetry-transport, MQTT)协议进行网页和程序的交互功能。为了区分不同的用户,设计了系统用户登录权限,根据不同的用户名和安全证书,给予用户不同的权限。为了使工厂实现真正的智能化,云端服务器会对 PLC 的工作参数进行实时数据分析,当结果显示 PLC 出现故障时,云端服务器将会给工程师发送一条预警信息,通知工程师需要对 PLC 设备进行维护。同时,为了各功能模块的用户可操作性,通过应用程序接口(application programming interface, API)函数的方式整合到用户界面中实现较好的用户体验。

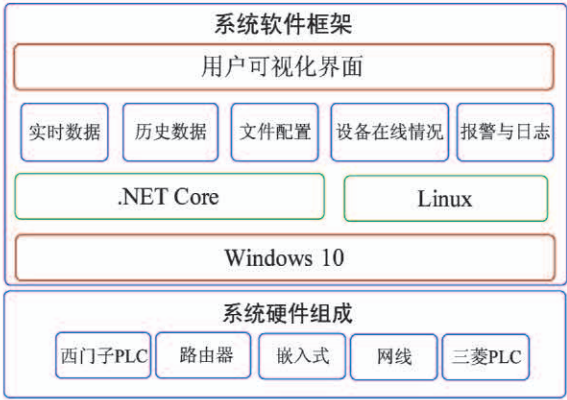


图 1 系统总体框架

拟以嵌入式工控机为核心,对整个系统进行逻辑控制,并通过交互控制系统,在工厂 PLC 设备与嵌入式工控机之间进行双向交互通讯,实现以下功能:(1)工厂 PLC 设备数据实时采集,并存储到云端数据库;(2)工业现场设备远程监控;(3)对采集数据进行实时分析,并支持现场异常报警;(4)PLC 设备远程升级,上传下载程序。

2 系统硬件构成

2.1 硬件总体方案设计

本系统硬件构成如图 2 所示,系统硬件主要由 1 个运行 Linux-Arm 系统的嵌入式工控机和 1 个交换机组成,嵌入式工控机和 PLC 设备与路由器分别

通过工业以太网的方式进行连接,并接入同一局域网,使用 S7 协议和 MC 协议采集 PLC 设备的实时信息。将采集的数据通过 4G 或无线的方式上传到云端数据库进行存储,并在客户端实时显示数据。

硬件构成中将嵌入式工控机作为采集控制设备。嵌入式系统平台相较于 PC 机具有功能强大、功耗低、成本低、体积小等特点、也不受设备所处环境的限制,目前市场上大部分数据采集和监控系统都采用独立系统,只能对设备数据进行实时采集和存储,并不能对设备进行远程的控制,而该系统不仅可以实现远程的数据采集,还能远程对 PLC 设备进行控制和故障维修,这主要是由于利用了嵌入式工

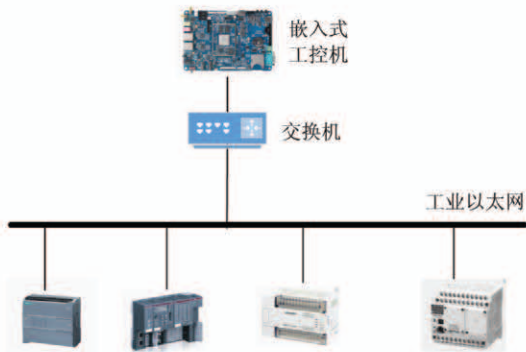


图2 系统硬件构成

控机的集成度高、免维护等优势,满足了在户外设站、无人值守等工业要求。

在图 2 中,所有 PLC 设备和嵌入式设备经过交换机以工业以太网的方式连接起来。嵌入式工控机目前采用的是飞凌嵌入式工控机,搭载了 ARM Cortex - A15 处理器。嵌入式工控机的结构图如图 3 所示,图中主要包括了以太网接口、电源接口、USB 接口、UART 串口和通用扩展接口。

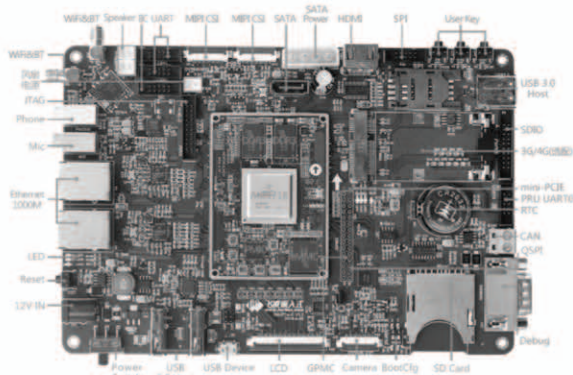


图3 嵌入式工控机的结构图

2.2 嵌入式工控机选型与环境配置

在调查了市场上部分嵌入式工控机后,选择了符合系统需求的几款嵌入式产品进行了对比,对比结果如表 1 所示。

表 1 嵌入式对比表

品牌	飞凌嵌入式 OK5718-C	飞凌嵌入式 AM335x	米尔科技 MYC-YA157C	有人物联网 USR-G805
处理器	ARM Cortex-A15	ARM Cortex-A8	双 Cortex-A7	无处理器
接口	2 个以太网口	1 个以太网口	1 个以太网口	1 个以太网口
其他功能	支持 Wifi、4G	其他功能网口较少， 不支持 4G 和无线	其他功能网口较少， 不支持 4G 和无线	其他功能网口较少， 支持 4G

综合各项指标以及设计需求,选择了飞凌嵌入式 OK5718-C 工控机。

由于成本等原因,嵌入式工控机资源有限,直接在嵌入式系统的硬件平台上编写软件比较困难,这时就需要使用交叉编译。交叉编译是在某个主机平台上(比如 PC 机上)用交叉编译器编译出可在其他平台上(比如 Linux-Arm 上)运行的代码的过程^[14]。首先搭建交叉编译环境,进行交叉编译工具链安装;然后修改根文件系统,用交叉编译工具编译 glibc 库

源码,下载适合版本的 libstdc++ 库文件,将旧版本文件系统的动态库文件删除并重新构建软链接;最后制作镜像文件,TF 卡内制作系统烧写盘,上电烧写系统镜像。

3 系统软件设计

3.1 软件总体框架设计

本系统软件采用分布式网络结构,将软件系统

设计成数据采集模块、网络通信模块和远程控制模块,具体结构如图 4 所示。数据采集模块通过嵌入式工控机采集 PLC 设备的数据信息,接着通过网络通信模块将采集的数据存储到云端服务器,当需要远程控制 PLC 设备时,需通过网络通信模块的权限管理给予权限,才能对 PLC 设备进行远程控制,实现一个完整的控制系统。

基于以上数据采集与监控系统的设计思想,并结合通用性和特殊性的要求,设计了图 5 所示的软件系统框架。它包括信息采集子系统、网络通信子系统和系统监控子系统。

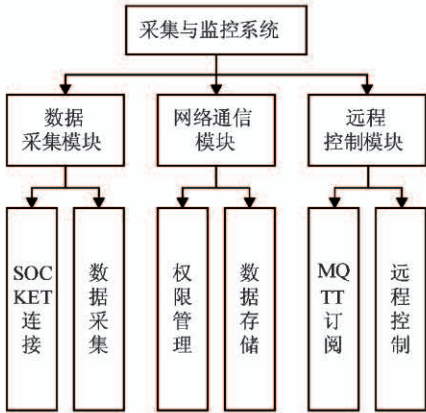


图 4 系统软件层次图

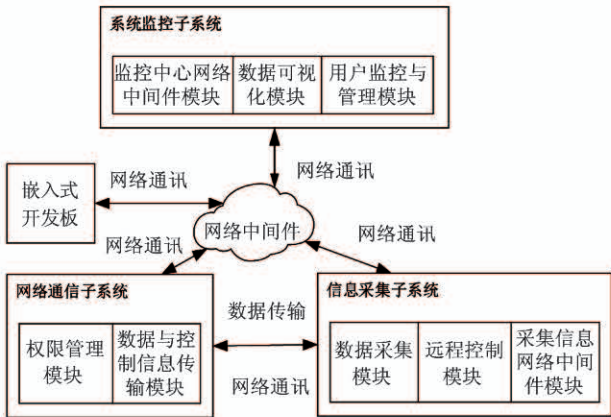


图 5 软件系统框架

(1)信息采集子系统

为了满足实时采集 PLC 设备的数据信息,接收网络通信子系统发送的控制指令等需求,设计了信息采集子系统。信息采集子系统包括数据采集模块、远程控制模块和采集信息网络中间件模块。数据采集模块主要完成对设备的数据采集的任务,并

将采集的数据发送到数据与控制信息传输模块;远程控制模块主要是根据接收到的控制指令,对设备下达对应的控制命令;采集信息网络中间件模块,主要完成数据的封装和数据的解析任务,将采集到的数据按照对应的数据报文解析数据和将需要发送的数据按照对应的数据报文进行封装。

(2)网络通信子系统

为了满足对登录用户的身份验证和传输数据等需求,设计了网络通信子系统。网络通信子系统包括权限管理模块和数据与监控信息传输模块。权限管理模块主要完成对用户的权限信息进行审核的任务,包括动态码发放、行为合法性验证、连接证书生成及发放策略,其中保存了所有设备的基础证书;数据与控制信息传输模块主要是接收采集的数据上传到云端服务器进行存储和接收用户的控制指令传送给远程控制模块。

(3)系统监控子系统

为了满足对 PLC 设备的实时监控和远程控制等需求,设计了系统监控子系统。系统监控子系统包括监控中心网络中间件模块、数据可视化模块和用户检测与管理模块。用户监控与管理模块主要是完成对设备数据的实时分析,预测设备运行状态和进行远程控制时的用户令牌发送的任务;数据可视化模块主要将采集的实时数据通过合适的方式呈现出来;监控中心中间件主要完成接收网络通信子系统上传的实时数据和控制信息,提供给用户监控与管理模块完成的对实时数据的进一步分析结果的任务。

3.2 数据采集系统

本文的数据采集系统需要通过设备厂商提供的设备 API 开发设备驱动。为了可以同时采集多台 PLC 设备,设计了采用线程池的方式进行多任务处理;同时为了能够采集更多型号的设备,设计了插件化的开发方式,提高程序的可扩展性。在系统设计中,采用异构的数据存储方式来满足数据的持久化存储。

3.2.1 线程池

电脑客户端交互软件可能会同时进行多项任务,需要使用多线程来并发执行任务,传统方式是事

先给每项任务分配好线程,任务启动时直接启动对应线程来执行任务,任务结束后再关闭线程,但是传统方式占用了很大内存空间和资源,不利于程序的执行。故设计了线程池来管理多个线程,在程序执行之前先设定一定数量的线程,当有任务执行时,先将任务加入到任务队列,然后分配一个空闲线程给这个任务进行处理,当任务结束后释放当前线程来缓解内存压力。

使用线程池技术可以有效管理多个线程,本文使用 C# 中的 ThreadPoo 类的 ThreadPoo. QueueUserWorkItem() 静态方法,只需在参数中加入对应任务的方法,线程池就可以自动创建线程来执行任务。

3.2.2 数据采集

数据采集系统为整个系统的核心部分,本文采用 S7 协议对西门子 S7 系列 PLC 设备进行数据采集,采用 MC 协议对三菱 PLC 设备进行数据采集。数据采集程序的流程图如图 6 所示。首先与 PLC 进行 SCOKET 连接,连接成功后则显示“PLC 连接成功”;接着与数据库进行 SOCKET 连接,连接成功后则显示“与数据库连接成功”;然后根据用户配置信息对指定数据进行数据采集;最后将采集的数据上传到云端数据库。

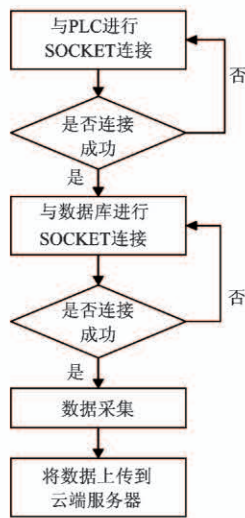


图 6 数据采集系统的流程图

3.2.3 数据存储模型设计

工厂的历史数据会给定期检修带来至关重要的作用。在进行数据存储模块设计时,采用非关系型

数据库 Redis 和关系型数据库 MySQL 相结合的方式,Redis 作为数据缓存和热数据存储等作用,MySQL 用以存储大量的历史数据。数据存储时,采集的实时数据首先上传到 Redis 数据库存储,接着将 Redis 中数据同步到 MySQL 中存储。数据查询时,经常使用的热数据存储在 Redis 中,首先从 Redis 中搜索需要的数据,Redis 中不能命中则转向 MySQL 历史数据库查询。采用 Redis 和 MySQL 相结合的方式大大提高了系统的存取速度。这样可以解决直接将数据存入到 MySQL 数据库可能进发太高的问题,也可以提高缓存速度,实时存储数据。

3.3 远程监控系统

系统的数据处理工作大部分都是在云端服务器完成的,这样的设计大大减小了数据上下行的负担,根据用户的不同需求设计了 B/S 和 C/S 结构的客户端(人机交互界面)。基于 Winform 的人机交互界面,可以供用户在工厂进行连接 PLC 设备,实时查看 PLC 设备运行参数等操作;基于 Web 服务的远程监控平台,可以供用户随时随地进行远程管理设备、查看实时数据、历史数据等操作。当现场无法调试设备时,还可以由现场工程师给定权限由专业工程师远程对设备进行调试。

3.3.1 远程控制

为了能够在远程调试 PLC 设备,设计了 Open VPN 模块(见图 7),用户通过 Open VPN 可以远程接入搭建的本地局域网,VPN 基于 OpenSSL 加密算法对传输数据进行加密,并建立 SSLVPN 服务,对每一嵌入式设备附加不同的证书副本,用以验证接入用户的合法性。当 Open VPN 连接成功后,便可以远程通过博途软件上传下载 PLC 程序,远程控制 PLC 设备。

Web 端作为 MQTT 的发布端,嵌入式工控机作为 MQTT 的接收端。远程控制时,在 Web 端点击某一事件按钮,发布端就会在主题下发布一个事件,当接收端接收到对应的消息后,根据消息来触发对应的事件。

在通过 Open VPN 接入本地局域网之后,可以通过博途软件连接 PLC 设备,上传下载程序,远程调试 PLC 设备,实现本地化操作。



图 7 Open VPN 登陆成功界面

3.3.2 基于 Winform 的客户端界面设计

PLC 是一种用于工业自动化的控制装置,但在数据采集、数据存储、可视化显示及用户界面等方面功能较差,如果使 PLC 和上位机结合起来,充分利用两者的优势就可以组成功能完好的工厂监控系统,因此,实现上位机和 PLC 设备之间的通讯成为关键技术。

上位机采用 Microsoft 公司的 Microsoft Visual Studio 软件编写,基于 Winform 架构,将上位机软件(人机交互软件)设计成具有设备连接、数据库连接、数据接收、数据显示、数据存储、图形化显示以及 PLC 运行状态显示等功能的软件,Winform 界面如图 8 所示。



图 8 基于 Winform 的客户端界面

为了实现实时采集 PLC 设备数据,需要将采集数据的程序进行移植,移植到嵌入式工控机中,嵌入式工控机和 PLC 设备通过以太网的方式连接,对 PLC 设备进行数据采集,数据存储。

3.3.3 数据分析模块的设计

Web 端对 PLC 设备的工作参数进行数据分析,

如果数据超出了模型范围,云端服务器将会给用户发送一条预警信息,提醒用户 PLC 设备出现故障。

基于文献[15]和[16]的数据分析方法,本文设计了 Web 端的数据分析模型。首先是智能分析部分,通过专家对基本工业设备的各项通用数据进行模型建立,借助支持向量机(support vector machine, SVM)等对初始化模型进行多次拟合学习,并建立模型迭代接口,使数据可以被有效使用;其次是预测性维护;最后通过对实时数据的运行模型拟合来对模型进行分析并提供结果。

4 实验验证

本文基于所设计的 PLC 数据采集和监控系统,搭建了数据采集和监控系统实验平台。实验平台包括 1 台西门子 S7-1200 PLC 设备、1 台飞凌嵌入式 TI AM5718 多核异构处理器 ARM Cortex-A15 开发板(Linux-arm 系统)、1 台路由器,以及 1 个笔记本电脑和若干根网线,如图 9 所示。

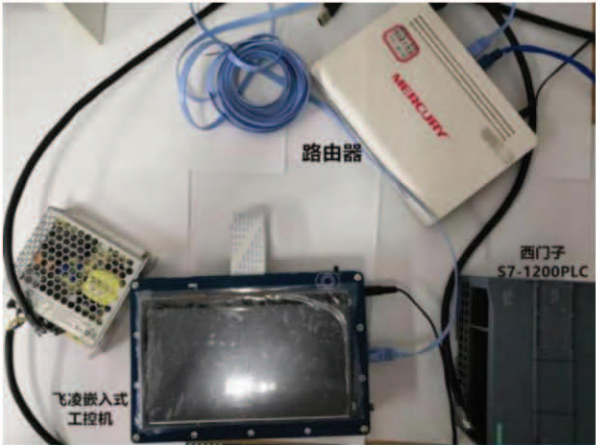
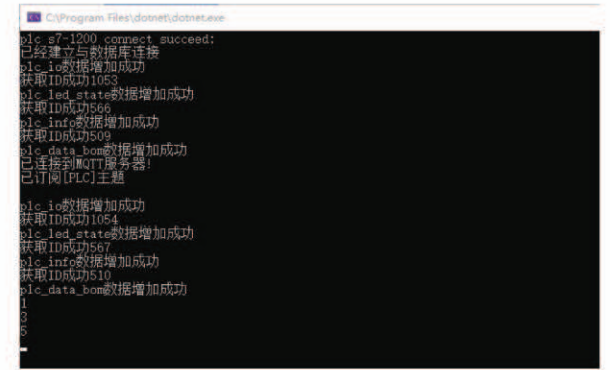


图 9 数据采集和监控系统实验平台

4.1 PLC 设备的数据采集

将个人电脑和一台 PLC 设备通过以太网的方式连接,并使两者位于同一网段中,在西门子 S7-1200 PLC 设备中载入编写好的梯形图,运行 Microsoft Visual Studio 软件的采集和数据存储程序,可以在个人电脑上对 PLC 设备进行采集,并将数据存储到云端数据库。结果如图 10(a)所示。在电脑上通过人机交互界面对 PLC 进行连接、数据采集和远

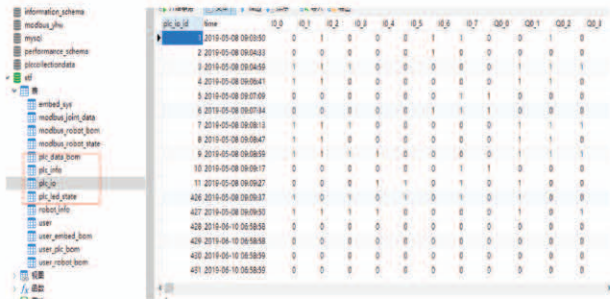
程控制等操作,结果如图 10(b)所示。将采集的数据存储到云端数据库,云端数据库中的数据如图 10(c)所示。



(a) 个人电脑程序运行结果



(b) 个人电脑 Winform 窗体界面



(c) 数据库历史数据

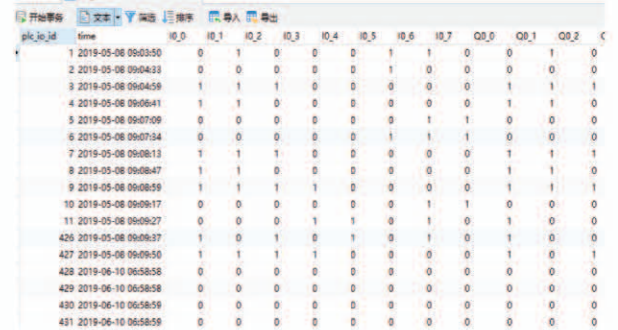
图 10 对 PLC 设备进行数据采集的实验结果

4.2 PLC 设备的远程监控

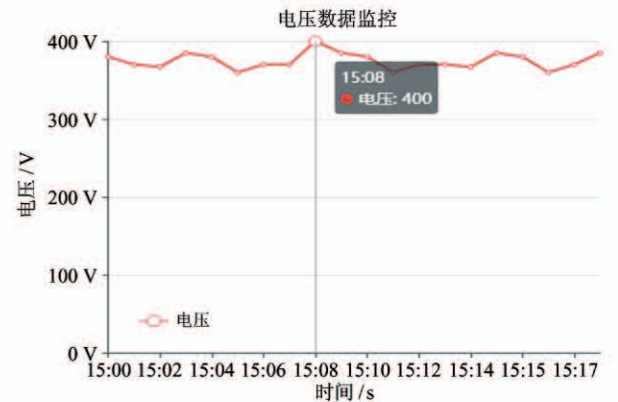
Web 网页端访问云端数据库获取对应的数据并显示在网页上,数据分析部分对采集到的 PLC 设备的电压、电流、温度等工作参数进行分析。当检测到 PLC 出现故障后,云端服务器会发送一条数据给现场工程师,通知工程师 PLC 设备出现故障,如果需要在远程对设备进行控制,则需要到现场工程师授权,远程可以通过博途软件对 PLC 设备进行远程

操作,直到 PLC 设备恢复正常。

在 Web 端查看 PLC 实时的设备运行状态,显示结果如图 11(a)所示。Web 端对 PLC 设备的电压和电流值等数据进行实时分析,如果超过了一定的阈值,就会发出警报。云端服务器会向用户发送短信



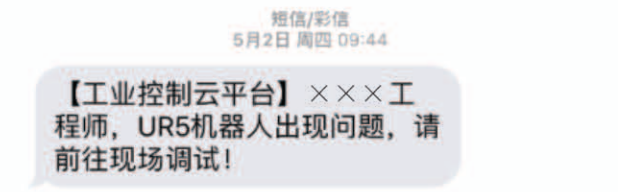
(a) PLC 设备实时数据图



(b) 电压数据分析图



(c) 远程请求控制 PLC 图



(d) 用户收到云端服务器短信图

图 11 对 PLC 设备进行远程监控的实验结果

通知,显示结果如图 11(b)所示。在 Web 端进行远程控制 PLC 设备,在通过权限校验后,可以对 PLC 设备进行远程控制,实现效果如图 11(c)所示。当设备发送故障,用户会收到短信通知,实现效果如图 11(d)所示。

4.3 数据采集和监控系统的评价指标

为了验证设计的系统是否能够在高并发的情况下正常运行,本文基于文献[17]的方法,在本系统实验平台进行高并发测试,给出了最大并发用户数、采集频率、平均事物响应时间和采集效率的具体量化指标值,如表 2 所示。由表 2 知,在给定的预期平均响应时间下,系统最多能支持的并发用户数量是 100,服务器的采集频率可在 0.1~1 s 间设置,平均事物的响应时间约为 0.84 s,采集效率达到 95%。以上性能指标值表明所设计的基于 .NET Core 架构的 PLC 数据采集和监控系统可以满足小型工厂对 PLC 设备数据采集和监控的需求,性能较好。

表 2 系统评价指标

评价指标	具体数据
最大并发用户数	100 个
采集频率	0.1~1 s
平均事物的响应时间	0.84 s
采集效率	95%

通过以上实验可知,该数据采集和监控系统利用安全性的加密通讯机制,结合远程监控平台,可为工厂实现低成本、快速的智能化升级。该系统支持多种 PLC 远程实时数据采集,并能将数据存储到云端数据库;通过 B/S 和 C/S 结构的客户端远程监控 PLC 设备、远程上传下载程序;对设备数据实时分析,并支持现场异常报警;当需要远程调试时,经现场工程师给定密匙,远程工程师可远程调试 PLC 设备等功能。该系统运行过程较稳定,可以有效地提高工厂的智能化水平,保证工厂设备安全正常的运行。

5 结 论

本文设计并实现了基于 .NET Core 架构的 PLC

— 100 —

数据采集和监控系统。该系统具有采集不同型号 PLC 设备的数据信息,对采集的信息进行数据存储,Web 端对 PLC 设备进行远程监控等功能。Redis 数据库实现了实时存储数据的效果,MQTT 使用户可以在远程 Web 端直接对 PLC 设备进行简单的操作,Open VPN 可以使用户在远程通过博途软件对 PLC 设备进行远程上传下载程序和远程控制 PLC 设备。此外,为了提高 PLC 数据采集的实时性,设计了异构的数据存储方式,将嵌入式工控机采集到的数据通过网络存储到云端数据库。实验结果表明所设计的 PLC 数据采集和监控系统运行效果良好。该系统解决了设备的远程控制等问题,为实际工业过程中的不同 PLC 设备的数据采集提供了一种新的设计思路。

参考文献

[1] 陈振友,吴智政,许树业,等. 基于 SIMATIC NET 的数控磨床车间 SCADA 系统设计[J]. 计算机测量与控制, 2019, 27(1): 121-125

[2] 杨建云,殷鹏程. 智能制造应用:浅谈数据采集[J]. 工业控制计算机, 2016, 29(10): 89-89

[3] 赵颖,侯俊杰,于成龙,等. 面向生产管控的工业大数据研究及应用[J]. 计算机科学, 2019(B06): 45-51

[4] 梁涛,孙天一,姜文,等. 区域性多风电场数据采集与远程集控系统[J]. 高技术通讯, 2019, 29(4): 387-394

[5] Harrison R, Vera D, Ahmad B. Engineering the smart factory[J]. Chinese Journal of Mechanical Engineering, 2016, 29(6): 1046-1051

[6] Vanags J, Galvanauskas V, Grigs O, et al. Online model-based optimization and control of fed-batch processes using Matlab code, OPC server, SCADA, and PLC[J]. New Biotechnology, 2012, 29: 163

[7] 丁莉君. 监控计算机与 PLC 数据通信的研究与应用[D]. 西安:西安建筑科技大学机械工程学院, 2007: 34-36

[8] 周华,邓雯静,周建英. 基于 CP5611 的 WINCC 与 PLC 通讯研究[J]. 自动化与仪器仪表, 2012(3): 43-44

[9] 高燕,刘瑾,张菁,等. 基于 PLC 的多数据采集监测报警系统研究[J]. 仪表技术与传感器, 2016(5): 40-43

[10] 姚伟,徐世许,殷欣,等. 温室大棚环境智能检测与调节系统研究[J]. 青岛大学学报(工程技术版), 2015,

30(3):38-42

- [11] 刘安平. PDA 高速数据采集分析系统及 Profinet 在其中的应用[J]. 自动化仪表, 2018, 39(3):81-83
- [12] 郭琼, 姚晓宁, 钱晓忠, 等. 基于 PLC 的远程监控系统研究及实践[J]. 实验技术与管理, 2019, 36(5): 94-97
- [13] 王琦, 王磊, 江豪, 等. 基于 C# 的 PLC 数据采集监控系统研发与应用[J]. 自动化与仪器仪表, 2019(8): 176-178, 183
- [14] 王晓燕, 刘军霞, 杨先文. 基于 Qt 和 Flash 的嵌入式 Linux 软件架构设计[J]. 现代电子技术, 2017, 40(1): 13-16
- [15] 苗玉. 改进粒子群算法优化 SVM 的故障诊断方法研究[J]. 机械工程与自动化, 2019(6): 14-15, 18
- [16] 梁涛, 侯振国, 邹继行, 等. 基于云计算平台的风电场 SCADA 系统的设计与应用[J]. 高技术通讯, 2018, 28(1): 8-14
- [17] 徐敬波, 李天峰, 张锡年. 基于大数据技术的油库生产远程监控系统设计[J]. 自动化仪表, 2019, 40(9): 70-75

PLC data acquisition and monitoring system based on .NET Core architecture

Yu Xinyi, Tang Quanrui, Yin Huiwu, Shi Tianfeng, Wang Xuyan, Ou Linlin

(College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023)

Abstract

In order to solve the problems of factory equipment brand diversification, different communication protocols and collection difficulty, the remote data acquisition and monitoring system based on the .NET Core architecture is studied. The system is developed by using heterogeneous data storage methods, distributed network models and data analysis models. Firstly, the embedded industrial computer and programmable logic controller (PLC) equipment are connected to the same local area network through industrial Ethernet. The operating status of the PLC device is collected in the real-time way, and then is uploaded to the cloud server. Real-time data of PLC devices can be obtained and analyzed remotely from the web side. When the analysis result shows that the device is broken down, the cloud server notifies the field engineer by message. If the field engineer cannot debug the device, the remote engineer can also remotely debug the device through Open VPN. Finally, the feasibility of the system is verified by experiments.

Key words: programmable logic controller(PLC), .NET Core, monitoring, data collection, distributed, heterogeneous data storage