

电磁阀远程控制及水压监测系统设计

张德茂¹, 袁晓¹, 陈文杰²

(1. 四川大学 电子信息学院, 成都 610065; 2. 成都华为研究所, 成都 611700)

摘要: 为了实现远程控制电磁阀及监测输水管道水压, 提高农业灌溉效率; 采用 STM32 微控制器及 Android 嵌入式系统, 开发出一套对电磁阀进行控制及水压监测系统; 微控制器通过串口连接 GPRS 模块, 从而接收命令控制电磁阀和发送水压数据; Android 手机客户端实现阀门控制界面和水压数据显示功能; 云服务器负责连接 GPRS 模块和手机客户端, 并且管理底层设备与用户信息; 该系统已运用在某智能节水灌溉公司的实验大棚基地中, 实验结果表明, 系统能实时进行远程控制及监测, 并能确保输水系统正常运行; 该系统能够推动农业现代化发展, 减少人力成本, 提高生产效率。

关键词: STM32 微控制器; 云服务器; 手机客户端; 农业现代化

Design of Remote Control of Solenoid Valve and Water Pressure Monitoring System

Zhang Demao¹, Yuan Xiao¹, Chen Wenjie²

(1. College of Electronics and Information, Sichuan University, Chengdu 610065, China;

2. Huawei Research Institute of Chengdu, Chengdu 611700, China)

Abstract: In order to realize the remote control of the solenoid valve and monitor the water pressure of the water conveyance pipeline, and improve the efficiency of irrigation in agriculture, a STM32 micro-controller and Android embedded system were used to develop a system of controlling solenoid valve and monitoring water pressure. The micro-controller through the serial port to connect the GPRS module to receive commands, and then control the solenoid valve and send water pressure data; Android phone client achieve the interface of controlling valve and display water pressure data; Cloud server is responsible for connecting the GPRS module and mobile client, and managing the underlying equipment and user information. The system has been applied in the experimental greenhouses of an intelligent water-saving irrigation company, the experimental results show that the system can carry out remote controlling and monitoring in real time, and ensure the normal operation of the water conveyance system. The system can promote the development of agricultural modernization, reduce labor costs, and improve production efficiency.

Keywords: STM32 micro-controller; cloud server; mobile client; agricultural modernization

0 引言

随着政府对农业现代化的重视以及物联网的快速发展, 传统的农业生产模式正逐渐向农业智能化方向发展, 智能化设备被广泛使用到农业生产过程中。这样的发展背景为电磁阀远程控制及水压监测系统的建立及推广奠定了良好的基础。目前关于研究农业环境监测的比较多, 农业环境监测对于指导种植以及了解农作物的生长环境有非常重要的意义^[1]。目前我国农业现代化的发展还处于初级阶段, 农业生产自动化是其中重要环节之一。智能节水灌溉系统有着非常广泛的市场需求, 譬如甘肃、陕西、新疆等缺水地区正逐步地推广节水灌溉系统。灌溉是农业生产过程中非常重要的一部分, 如何把控灌溉时间以及灌溉量是一个重要的问题。利用覆盖面广、技术成熟且传输速度快的 GPRS 网络实现远程控制电磁阀可以有效解决这个问题^[2]。灌溉系统中爆管以及电磁阀损坏现象经常发生, 在灌溉系统中添加一个水压监测模块, 如果水压超过阈值, 就关闭水

泵开关并及时调整水泵功率, 这样可以有效防止该类现象的发生。在农业生产过程中, 使用电磁阀远程控制及水压监测系统具有很重要的意义, 不仅解放劳动力, 降低生产成本, 并且该系统具有广泛的应用市场。

1 系统设计方案

电磁阀远程控制及水压监测系统主要分为七个部分如图 1 所示: 传感器、电磁阀、电机驱动模块 DRV8823、微控制器 STM32、GPRS 通信模块、云服务器、Android 手机客户端。微控制器通过 GPRS 模块接收来自云服务器的控制命令和发送传感器数据给云服务器。云服务器负责管理底层设备及用户信息并且缓存控制命令和传感器数据。手机客户端与云服务器之间通过互联网进行连接, 用户通过手机 APP 登录即可进行控制操作和获取水压数据并进行显示。

系统的总体框架如图 1 所示, 分成三大模块。底层控制模块的核心是基于 ARM Cortex-M4 核的 STM32L476RE 微控制器。中间服务模块是依托云平台搭建的云服务器。上层应用模块是在 Android 手机平台上开发的 APP。

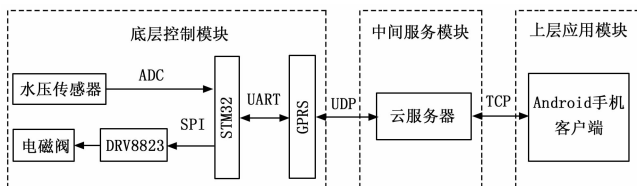
2 系统硬件设计

系统硬件部分的主要任务是完成电路设计。以微控制器为核心, 需要设计 4 个电路模块分别为: 微控制器电路, 电磁阀控制电路, GPRS 通信模块电路, 水压采集电路。由于整个底

收稿日期: 2016-12-28; 修回日期: 2017-02-13。

作者简介: 张德茂(1993-), 男, 湖北黄石人, 硕士研究生, 主要从事嵌入式系统方向的研究。

袁晓(1964-), 男, 四川成都人, 副教授, 主要从事现代电路设计与研究。



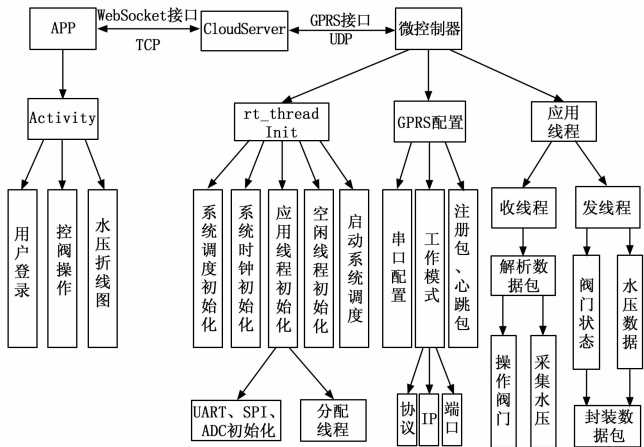


图 5 系统软件设计框图

实时、嵌入式系统相关的各个组件: TCP/IP 协议栈, 文件系统, libc 接口, 图形用户界面等。RT-Thread 操作系统的初始化包括: 系统时钟, 系统调度, 应用线程, 空闲线程。系统初始化和启动系统调度在主程序内完成。有两个应用线程分别是接收线程和发送线程。在启动线程之前, 需要配置好相应的硬件接口: SPI、UART、ADC 等。

3.1.3 收发线程

微控制器与服务器之间通信是通过 GPRS 模块, 采用的传输协议是 UDP。中间传输的数据协议格式如表 1 所示。

表 1 数据协议格式

设备地址	命令	选项	保留	序号	长度	数据	CRC 校验
(1Byte)	(4bits)	(2bits)	(2bits)	(1Byte)	(1Byte)	(n Bytes)	(1Byte)

1) 接收线程实现流程:

```
(1)rt_thread_create(参数); //创建线程
(2)rt_thread_startup(参数); //启动线程
(3)While(true)
{
RecvAndProcessData(参数) //接收并处理数据
{
rt_sem_take(参数);
...
rt_device_read(参数);
//操作系统读函数
...
rt_sem_release(参数);
}
}
(4)Parse_From_Array(参数); //解析命令函数,按照表 1 数据协议格式解析数据包
(5)相关命令:
发送注册请求 SendRegisterReq(参数);
处理注册应答 ProcessRegisterRpl(参数);
处理操作请求 ProcessCmdReq(参数)。如:开阀、关阀、采集水压等操作请求。
(6)返回处理结果给服务器。
2) 发送线程实现流程:
(1)rt_thread_create(参数); //创建线程
(2)rt_thread_startup(参数); //启动线程
```

```
(3) While(true)
{
SendRegisterReq(参数); //注册函数:与服务器建立连接并提供注册信息
{
ToArray(参数); //封包函数:按照表 1 数据协议格式打包数据
SendData(参数) //发送函数:通过操作系统的设备写函数调用串口发送数据
{
rt_sem_take(参数);
rt_device_write(参数);
//操作系统写函数
...
rt_sem_release(参数);
}
}
rt_thread_delay(times);
//操作系统延时函数
}
```

信号量是一种轻型的用于解决线程间同步问题的内核对象, 线程可以获取或释放它, 从而达到同步或互斥的目的。在接收和发送线程之间就采用了信号量机制来保证两个线程之间的同步和互斥。

信号量的创建: rt_sem_create(参数)。

信号量的抢占: rt_sem_take(参数)。

信号量的释放: rt_sem_release(参数)。

3.1.4 控制阀门开关

开阀函数 OpenValve(阀门号)和关阀函数 CloseValve(阀门号)是通过操作系统的设备写函数调用 SPI 向驱动芯片 DRV8823 写入 16 位二进制数据(相应的数值可以查芯片手册)。

3.1.5 采集水压数据

采集水压函数 GetWaterPressure(通道号)是通过操作系统的读函数调用 ADC 接口将模拟量转换成数字量。12 位的 ADC 以内部参考电压 3.6 V 作为参考值, 因此将参考电压分成 4096 等份。测量的电压值=AD 读出的值/4096 * 3.6 (V)。实际的水压值要根据传感器的转换公式来进行转换处理。

3.2 服务器端软件框架及主流程

服务器的开发环境为 IDEA15.0.6 + JDK8 + Tom-Cat8.0.3 + MySQL5.7。服务器使用的是 Spring MVC 框架。MVC, 以设计界面应用程序为基础的设计模式, 它主要通过分离模型、视图及控制器在应用程序中的角色将业务逻辑从界面中解耦^[7]。Model 负责封装应用程序数据在视图层展示。View 仅仅只是展示数据, 不包含任何业务逻辑。Controller 负责接收来自用户的请求, 并调用后台服务 (Manager/Dao) 来处理业务逻辑。处理完之后, 后台业务层可能会返回一些数据在视图层上展示。控制器收集这些数据及准备模型在视图层展示。

云服务器是一种基于 WEB 服务, 提供可调整云主机配置的弹性云技术, 整合了计算、存储与网络资源的 Iaas 服务, 具备按需使用和按需即时付费能力的云主机租用服务^[8]。在灵活性、可控性、扩展性及资源复用性上都有很大的提高。云服务器软件框架如图 6 所示。

Entity 模块主要功能是定义云服务器与微控制器、手机

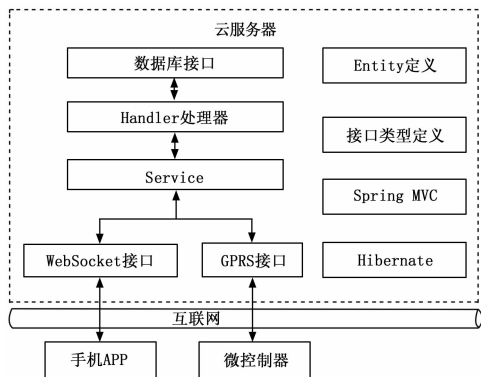


图 6 服务器软件框架

APP 应用进行数据传输的基本数据类型。Entity 实体同时也对应着数据库中的表结构，本设计中需要建立的实体有：用户信息、节点信息、传感器信息、开关信息。

接口类型定义模块主要功能是实现 WebSocket 接口中的基本通信类型的定义，主要包括命令类型定义及应答类型定义等。WebSocket 接口采用 WebSocket 作为承载方法，接口采用的数据格式为 JSON 格式。

Spring MVC 模块主要实现 Spring MVC 框架环境。

Hibernate^[9] 模块主要实现数据库与 Hibernate 的绑定关系。

数据库接口模块主要实现常用数据库操作接口功能，如查询、修改及删除等操作。

Handler 模块主要功能是对外部接口请求的处理。

Server 主要是实现 Webservice 接口。

WebSocket 模块主要实现手机 APP 与云服务器之间的通信。

GPRS 接口采用 UDP 作为承载方法，接口使用 NCP 协议格式进行通信。

服务器在启动后，将首先运行 Spring 框架，加载相关配置，包括 Controller、Service、Hibernate 等基本功能组件。然后，加载 WebSocket 接口模块，开始监听 WebSocket 连接。最后，将加载 GPRS 接口模块，开始监听来自 GPRS 的 UDP 消息。此后，服务器主要的功能便是监听 WebSocket 连接并进行管理及消息命令的处理，以及监听 GPRS 的 UDP 消息和 GPRS 终端的管理。GPRS 终端的管理包括：（1）监测 GPRS 终端连接超时管理；（2）GPRS 终端连接合法性检验；（3）GPRS 命令消息收发处理。

3.3 手机客户端

3.3.1 Android 手机客户端架构

Android 是由 Google 开发的基于 Linux 内核的综合操作系统^[10]。本系统的手机客户端与服务器采用的是常用的 C/S 架构（客户端/服务器）模式，Android 手机客户端通过 WebSocket 接口与服务器进行通信。客户端发送命令给服务器，服务器再将命令通过 GPRS 接口发送给底层控制模块。底层控制模块将采集的数据上传到服务器内的数据库中进行存储。手机客户端设计架构灵活、操作稳定、数据可靠。

3.3.2 Android 客户端界面设计及主流程

Android 手机客户端支持 4.0 以上的手机，开环境为 Android—Studio2.2，采用的是 XML 布局。APP 有 3 个 Activity

包括用户登录、阀门开关、水压曲线图显示等界面。采用 MVC 模式进行系统开发。

APP 在启动时，将通过 WebSocket 与云服务器建立连接，并提交身份信息（包括用户名及密码）。服务器通过对用户身份进行验证后，将验证结果返回给 APP。APP 收到登陆验证结果后，将开始维护 WebSocket 的连接（通过心跳数据包）。此后，APP 将可以执行向服务器发送命令或从服务器读取信息等操作。通常，APP 将首先从服务器上读取当前用户相关的节点信息及开关信息等，并将其显示到界面上以使用户的操作。用户在对开关进行操作时，APP 将通过 WebSocket 向服务器发送开关操作命令，等待服务器的命令应答。服务器完成命令的处理后（包括向底层控制模块发送命令及接收命令应答），通过 WebSocket 向 APP 返回命令结果（错误码及错误信息等）。APP 收到来自服务器的命令应答后，将根据结果更新界面或通知用户。

水压数据采用了 AChartEngine 图表引擎进行折线图显示。AChartEngine 是为 Android 应用设计的绘图工具库，通过对其参数进行相应配置，以及对原有图表的重新封装后定制出需要使用的图像数据接口^[11]。

4 系统测试与结果分析

在实际的农场温室大棚中，对该电磁阀远程控制与水压监测系统软硬件测试。首先用户通过用户名和用户密码登录客户端。测试表明：1）电磁阀控制操作稳定可靠，并能正确反馈控制信息。控阀成功时，界面会弹出控阀成功的提示信息并且会显示阀门状态。2）水压监测能在允许误差范围内及时给用户反映当前输水管道的水压。用户还可以设定水压阈值。水压显示界面如图 7 所示。

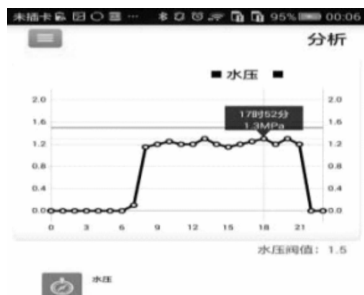


图 7 水压显示界面

结果分析：1）采用 GPRS 通信方式，每个模块都能独立稳定的连接公网。相比 WIFI 连接方式，GPRS 方式可以有效避免连接距离、连接数量的限制问题；不足点是需要插卡和续费。2）从实验基地的长期测试结果中估算出，远程控阀的成功率达到 90% 以上。根据不同型号的电磁阀，调整输出电流和电磁阀的结构，可以解决控阀成功而实际没有开阀放水的问题。3）管道水压监测，测量数据跟专业仪器测量的结果有一定的误差，但在一定程度上可以有效避免水压过高出现爆管的现象发生，达到预期的功能要求。4）APP 软件可以通过扫码的方式下载，便于用户注册使用。界面操作简单，功能稳定可靠。从实际用户的反馈中，可以了解到待开发的功能还有许多，比如控制水泵开关，周期性定时灌溉等有实际需求的功能。

5 结束语

针对传统农业生产过程中存在耗时耗力的问题，提出了电

磁阀远程控制及水压监测系统。该系统的特点: 引入 GPRS 无线通讯技术, 具有连接方便、覆盖面广、传输速度快的优势^[10]; 采用基于云平台搭建的服务器, 管理用户信息和设备信息; 设计了基于 Android 的移动客户端 APP 软件, 使农业灌溉智能化, 便捷化。随着国家对农业现代化的不断推进, 以及智能控制和物联网技术的不断发展, 农业智能化是必然的发展趋势。嵌入式系统和无线通信技术相结合的远程控制系统必将是发展的潮流。电磁阀远程控制及水压监测系统还有许多需要完善之处, 随着技术的不断提高以及系统的不断完善, 该系统必将给用户带来更人性化的操作。

参考文献:

[1] 王恩亮, 华 驰. 基于物联网技术的农业环境监测站的设计 [J]. 计算机测量与控制, 2016, 24 (5): 18-20.
[2] 李志军, 刘亚善. 基于 ARM 和 GPRS 的多功能智能表数据采集器 [J]. 计算机测量与控制, 2015, 23 (8): 2918-2920.

(上接第 76 页)

5 网络安全自检设计

对于“互联网+排涝泵站群”远程网络管控系统而言, 保证安全首当其冲。在该系统中主要考虑远程操作安全性和数据传输的安全性^[14]。在远程操作安全性方面, 主要是通过设置用户操作权限, 依据网络通信状态和水泵当前的运行状态来有效避免非授权人员任意操作和授权人员误操作。在数据传输安全性方面, 首先通过对系统可能出现的网络通信故障(网络线路断路、下位触摸屏离线不响应)分析, 设计采用设置“心跳包”的模式保证系统的安全: 首先由 PC 与下位触摸屏之间约定共同的一个参数, 并定期由 PC 向触摸屏进行数据写入, 触摸屏定期查询该变量值的变化, 若该变量值没有按照既定的方式改变, 则自动重新启动触摸屏; 重启后如果仍然没有收到上述正确信息, 则进行本地相应网络异常声光报警。

6 远程网络管控系统的应用调试

该管控系统(如图 5 所示)在常熟辛庄进行了实际安装调试。经过实际应用发现, 该系统不仅监控及时, 能够有效避免泵站排涝故障, 同时还能远程智能启停水泵, 方便无人值守。同时, 还能对整个城镇的各个主要水域的水情变化趋势进行实时监控, 水情数据进行存储, 便于进行有效的区域排涝站点设置与分析管理。



图 5 远程监控泵站示意

7 结论

本系统从现有泵站信息化改造实际出发, 结合经济性、系统稳定安全性等多方因素考虑, 设计了一套安全、稳定的智能

[3] ST 公司. STM32L476RE 使用手册 [EB/OL]. <http://www.st.com/>, 2016.
[4] TI 公司. DRV8823 使用手册 [EB/OL]. <http://www.ti.com/>, 2016.
[5] USR 公司. USR-GPRS232-7S3 使用手册 [EB/OL]. <http://usr.cn.makepolo.com/>, 2016.
[6] RT-Thread 编程指南 [EB/OL]. http://www.rt-thread.org/download/manual/rtthread_manual.zh.pdf, 2016.
[7] 戴 克, 林仪明, 崔 毅. Spring MVC 学习指南 [M]. 北京: 人民邮电出版社, 2015.
[8] 百度文库. 云服务器 [EB/OL]. <http://wenku.baidu.com/view/4630404dc850ad02de80419d.html>, 2016.
[9] 刘京华. Java Web 整合开发王者归来 [M]. 北京: 清华大学出版社, 2010.
[10] 李 宁. Android 开发权威指南 [M]. 北京: 人民邮电出版社, 2013.
[11] Achartengine [EB/OL]. <http://code.google.com/p/achartengine/>, 2016.

化排涝泵站群远程网络管控平台, 实现了就地以及远程水情、泵站工作状态的实时监控, 解决了传统泵站控制系统受地域限制的缺陷, 通过智能化的优化方法预先设置了多模式的泵站群调度模式, 不仅方便安全排涝, 而且能够有效节省能耗。通过实例运行验证了本文提出的系统架构和安全机制的正确性和可行性。该系统的研究, 对当前排涝站信息化改造具有一定的指导价值和现实意义。

参考文献:

[1] 李 琪, 许建中, 李瑞明, 等. 中国灌溉排水泵站的发展与展望 [J]. 中国农村水利水电, 2015 (12): 6-10.
[2] 陈 虹, 唐鸿儒, 杨 鹏, 等. 泵站监控系统的现状和趋势 [J]. 中国农村水利水电, 2002 (2): 55-58.
[3] 霍 宁. 泵站监控系统及其结构的发展趋势 [J]. 水电自动化与大坝监测, 2004 (4): 81-85.
[4] 刘瑞江, 张业旺, 闻崇炜, 等. 正交试验设计和分析方法研究 [J]. 实验技术与管理, 2010, 27 (9): 52-55.
[5] 魏效玲, 薛冰军, 赵 强. 基于正交试验设计的多指标优化方法研究 [J]. 河北工程大学学报(自然科学版), 2010, 27 (3): 95-99.
[6] 刘静森, 程吉林, 黄 勇. 不受潮汐影响城镇圩区排涝泵站群常规调度方案优化 [J]. 灌溉排水学报, 2015, 34 (3): 17-23.
[7] 刘静森, 程吉林, 龚 懿. 城镇圩区排涝泵站群日常运行方案优化 [J]. 农业工程学报, 2014, 30 (17): 141-147.
[8] 方国华, 曹 蓉, 刘 芹, 等. 改进遗传算法及其在泵站优化运行中的应用 [J]. 南水北调与水利科技, 2016, 14 (2): 142-147.
[9] 徐鹏鹏. 基于 S7-300/200PLC 和 Modbus 协议的电力参数采集系统 [J]. 工矿自动化, 2011 (12): 104-107.
[10] 周亚平. 建立在以太网基础上的泵站集控系统 [J]. 中国农村水利水电, 2008 (8): 133-136.
[11] 孙大林. 基于 Modbus-TCP 的 OPC 数据访问服务器的研究与开发 [D]. 北京: 北京交通大学, 2007.
[12] 孟 磊, 邹志云, 郭宇晴, 等. 基于 Modbus TCP/IP 的 HMI-PLC-SCADA 系统设计 [J]. 石油化工自动化, 2015, 51 (6): 40-42.
[13] 赵晓峰. 基于 OPC 技术的 Web 工控系统的研究与实现 [J]. 计算机应用, 2016 (2): 62-64.
[14] 黄 磊. 基于 Internet 的远程 PLC 控制研究 [D]. 西安: 西安工业大学, 2014.