

## 基于 5G 与阿里云的六轴运动实验平台监控系统设计<sup>①</sup>

徐建明<sup>②</sup> 张宇轩 朱俊威 俞 立

(浙江工业大学信息工程学院 杭州 310023)

**摘 要** 设计了一种基于 5G 与阿里云的六轴运动实验平台监控系统,包括云端 WEB 监控平台和本地运动控制平台。云端 WEB 监控平台采用前后端分离技术搭建,前端使用原生的 HTML 文本标记语言、CSS 样式和 JS 脚本实现,后端使用 Spring 技术基于 Spring Boot 框架搭建。本地控制平台使用嵌入式系统开发技术搭建,包括 5G 模组和运动控制器,5G 模组作为联网通道使用 USB3.0 传输技术与运动控制器交互数据。运动控制器搭载 MQTT 客户端、进程通讯机制和 Codesys Runtime,MQTT 客户端向阿里云服务器订阅或发布消息,Codesys Runtime 通过 EtherCAT 总线控制六轴运动实验平台运行,进程通讯机制实现 MQTT 客户端和 Codesys Runtime 间的通信。最后创建六轴运动实验平台的 Codesys Pou 工程,利用云端 WEB 监控界面验证监控系统的可行性,并且通过 SpeedManager 软件测试了 4G 和 NSA 模式下 5G 网络的网络状况。

**关键词** 运动控制器; MQTT; 5G; Codesys Pou; WEB 监控

### 0 引 言

工业生产不断向信息化发展过程中,监控系统起到了不可或缺的作用。4G 无线监控系统可以用于一些特定环境,但是在要求大规模数据传输和设备远程操控作业环境下比较无力。随着 5G 的到来这些问题将迎刃而解。

5G 相比于 4G 网络具有更低的延时和更高的速率及容量<sup>[1]</sup>。低延时与大容量可以使 5G<sup>[2-3]</sup>应用于实时高精度的操作环境下,如王浩等人<sup>[4]</sup>基于 5G 网络对城市轨道交通全自动无人驾驶列车控制系统研究。5G 网络的高速率、低延时同样适用于监控系统,使监控系统更加高效、稳定和可靠,柳曦等人<sup>[5]</sup>和 Rehmani 等人<sup>[6]</sup>提出了 5G 通信技术在火电厂的应用。应巨林等人<sup>[7]</sup>进行了基于 5G 技术的农作物视频监控平台的研究与应用。

传统的工业监控系统大多采用硬可编程逻辑控

制器(programmable logic controller, PLC),如邓鑫<sup>[8]</sup>使用 S7-300 设计煤矿带式输送机运行监控系统。郭恒瑞等人<sup>[9]</sup>提出了基于 PLC 设计的矿用装煤平车线监控系统。但是硬 PLC 通常用于一些逻辑控制系统,难以胜任多轴运动控制。随着计算机技术的迅速发展及 PLCopen<sup>[10-11]</sup>标准的提出,嵌入式软 PLC 技术的出现打破了传统 PLC 的局限性。嵌入式软 PLC 具有开放的硬件体系结构,丰富的指令集,更高的性价比,适用于计算机网络和开发方便等特性。软 PLC 在监控系统中越来越流行,如徐建明等人<sup>[12]</sup>使用 Soft PLC Running System<sup>[13]</sup>实现基于 WEB 的工业机器人 3D 虚拟动态监控系统。目前的监控系统大多采用 C/S(客户端/服务器)模式搭建,如张爱民等人<sup>[14]</sup>基于 TCP/IP 协议设计的工业机器人远程监控与诊断系统,此模式下,用户需要安装客户端。然而采用 B/S(浏览器/服务器)框架搭建的监控系统简洁方便,如吴丽梅<sup>[15]</sup>设计基于 B/S

① 国家自然科学基金-浙江省自然科学基金联合基金两化融合项目(U1709213)和国家自然科学基金面上项目(61374103)资助。

② 男,1970 年生,博士,教授;研究方向:迭代学习控制,电机伺服控制技术,机器人控制技术;联系人,E-mail: xujm@zjut.edu.cn。  
(收稿日期:2020-10-15)

构架的 EPICS (experimental physics and industrial control system) 实时监测系统,董玉德等人<sup>[16]</sup>设计基于物联网的路灯可视化监控系统。

针对上述问题,本文在基于 B/S 架构搭建云端监控平台前提下,前端使用原生代码实现,后端使用简洁的 Spring Boot 框架搭建。本地控制平台由运动控制器和 5G 模组组成,运动控制器通过 USB 驱动程序识别 5G 模组,使用消息队列遥测传输 (message queuing telemetry transport, MQTT) 协议通过 5G 网络与阿里云服务器进行数据交互,并且使用 Codesys IDE 创建了一个 Pou 六轴运动实验平台工程和 WEB 前端界面验证基于 5G 与阿里云的六轴运动实验平台监控系统。

## 1 系统整体框架

### 1.1 系统硬件框架

监控系统硬件框架如图 1 所示。六轴运动实验平台由 6 个电机与 6 个伺服驱动器组成,由运动控制器通过 EtherCAT 总线传输信号到伺服驱动器控制电机的运动。运动控制器选取 Cortex-A8 为核心的工业控制 MCU<sup>[17]</sup> AM335X 芯片,运动控制器安装了 Linux 操作系统并在 Linux 系统中搭载 Codesys Runtime 系统组成了嵌入式软 PLC。联网部分采用 MH5000-31 5G 模组,模组接入天线并且插入具有 5G 业务 SIM 卡处于 5G 网络覆盖范围内拨号连入 5G 网络,用于本地控制平台与云服务器数据互传。监控

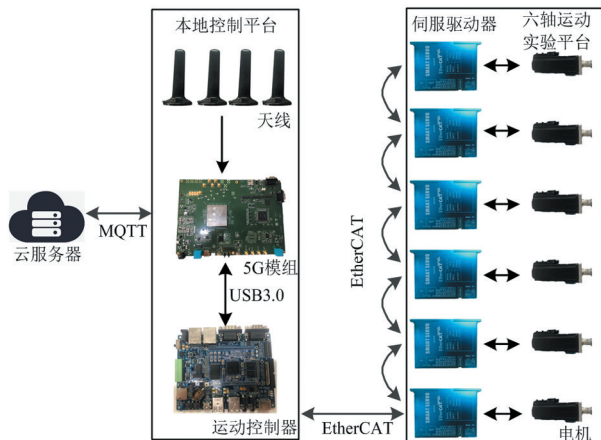


图 1 监控系统硬件框架图

平台部署在云服务器上,用户通过浏览器访问特定 IP 地址的服务器监视和控制六轴运动实验平台。

### 1.2 系统软件框架

六轴运动实验平台监控系统软件架构主要由两大部分组成,本地控制平台和云端监控平台。监控系统软件框架图如图 2 所示。

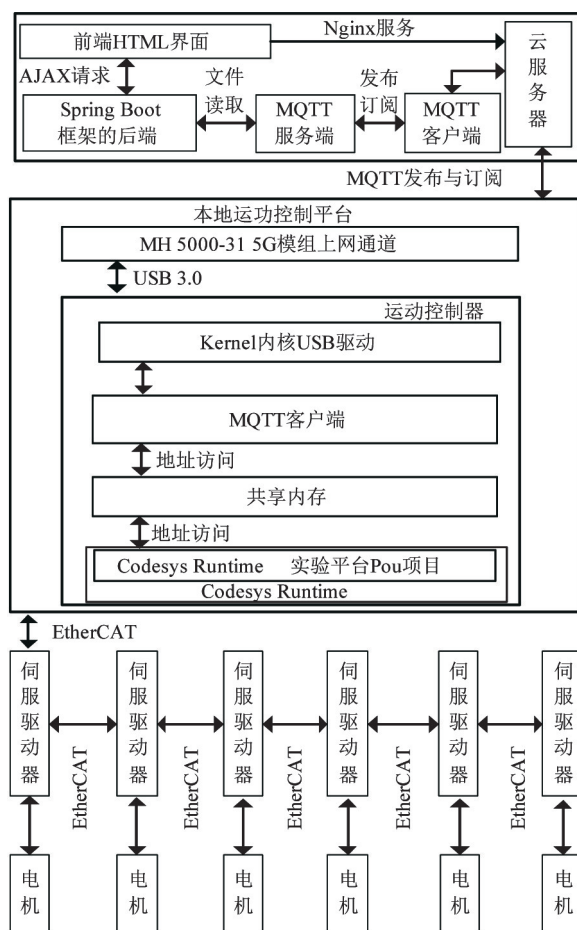


图 2 监控系统软件框架图

云端监控平台用于处理本地控制平台发送的实验平台数据且显示到前端界面或者发送实验平台控制指令到本地控制平台,平台由 WEB 前端界面、Spring Boot 框架后端、MQTT 客户端和服务端组成。为了避免同源的问题存在,WEB 前端使用 Nginx<sup>[18]</sup>部署在云服务器上, Spring Boot 后端主要负责处理前端的 AJAX (asynchronous JavaScript and XML) 请求。

本地控制平台分为 5G 模组和运动控制器。运动控制器使用的是搭载了 MQTT 客户端、Codesys Runtime 等服务的 Linux 系统,由于 Codesys Runtime

和 MQTT 客户端属于不同的进程,所以运动控制器还需要搭载进程通讯服务,通过共享内存实现。Codesys Runtime 向共享内存写入实验平台的运行数据或者从共享内存读出实验平台的控制指令, MQTT 客户端从共享内存中读出设备数据并且上传到云服务器或者接受服务器发送的指令写入共享内存中。Codesys Runtime Pou 工程通过 EtherCAT 总线发送指令给各轴伺服驱动器实现六轴实验平台运动控制功能。

## 2 本地控制平台设计

本地控制平台主要包括 5G 模组和运动控制器两大部分,编写 USB 驱动程序建立 5G 模组与运动控制器的通讯,移植 MQTT 协议在运动控制器中建立 MQTT 客户端。

### 2.1 内核 USB 驱动编写

本地控制平台与云端服务器进行数据交互的前提是本地控制平台接入互联网。由于 5G 模组与运动控制器通过 USB 线连接,所以使用模组需要编写 Linux 内核 USB 驱动。本文使用的是 Linux-3.2.0 内核,对 5G 模组没有很好的兼容,需要对内核中的 CDC 驱动和 ECM 驱动进行修改,主要过程如下。

(1) 在 linux \_src/include/linux/usb. h 中添加设备的宏。

(2) 在 linux \_src/drivers/usb/serial/option. c 文件中 usb \_device \_id option \_ids[ ] 的 ID 列表添加用于匹配设备数据的 ID。

(3) 在 linux \_src/drivers/usb/serial/usb \_wwan. c 中增加对 bcdUSB 值和华为 VID 值的定义。

(4) linux \_src/drivers/usb/serial/usb \_wwan. c 文件中 usb \_wwan \_write 函数添加对零包的判断。

USB 驱动修改后更新至文件系统,在系统设备中出现 ttyUSB0、ttyUSB1、ttyUSB2 和 ttyUSB3 4 个端口表示 5G 模组已被识别,如图 3 所示。

模组正确识别后使用脚本向 ttyUSB1 接口发送信息使本地控制平台实现上网功能,具体脚本及解释如下:

```
echo "AT + COPS = 0" > /dev/ttyUSB1
sleep (1)
```

```
echo "AT^NDISDUP = 1,1," > /dev/ttyUSB1
```

```
sleep (1)
```

```
dhcpcd -i eth0
```

首先注册到相应的运营商网络,其次拨号上网,最后通过 dhcpcd 获得动态 IP 地址用于上网所用,本地控制平台正确获得动态 IP 地址,如图 4 所示, IP 为 169.254.129.250。

```
usb 1-1.4: new high-speed USB device number 3 using musb-hdrc
usb 1-1.4: New USB device found, idVendor=12d1, idProduct=15c3
usb 1-1.4: New USB device strings: Mfr=12, Product=13, SerialNumber=14
usb 1-1.4: Product: HUAWAI_MOBILE
usb 1-1.4: Manufacturer: HUAWAI_MOBILE
usb 1-1.4: SerialNumber: 68T0220212000233
cdc_ether 1-1.4:2.0: eth2: register 'cdc_ether' at usb-musb-hdrc.1-1.4,
option 1-1.4:2.2: GSM modem (1-port) converter detected
usb 1-1.4: GSM modem (1-port) converter now attached to ttyUSB0
option 1-1.4:2.3: GSM modem (1-port) converter detected
usb 1-1.4: GSM modem (1-port) converter now attached to ttyUSB1
option 1-1.4:2.4: GSM modem (1-port) converter detected
usb 1-1.4: GSM modem (1-port) converter now attached to ttyUSB2
option 1-1.4:2.5: GSM modem (1-port) converter detected
usb 1-1.4: GSM modem (1-port) converter now attached to ttyUSB3
net eth1: CPSW phy found : id is : 0x4dd072
```

图 3 5G 模组设备端口信息图

```
dhcpcd[1073]: can0: up_interface: Invalid argument
dhcpcd[1073]: all: IPv6 kernel autoconf disabled
dhcpcd[1073]: usb0: IPv6 kernel autoconf disabled
dhcpcd[1073]: eth0: IPv6 kernel autoconf disabled
dhcpcd[1073]: eth1: IPv6 kernel autoconf disabled
dhcpcd[1073]: eth2: IPv6 kernel autoconf disabled
dhcpcd[1073]: eth0: waiting for carrier
dhcpcd[1073]: eth1: waiting for carrier
dhcpcd[1073]: usb0: using IPv4LL address 169.254.129.250
dhcpcd[1073]: usb0: adding host route to 169.254.129.250 via 127.0.0.1
dhcpcd[1073]: usb0: adding route to 169.254.0.0/16
dhcpcd[1073]: forked to background, child pid 1083
```

图 4 dhcpcd 获得动态 IP 信息图

### 2.2 MQTT 客户端设计

MQTT 是一种基于发布/订阅范式的消息协议,具有开放、简单、轻量的特性,是为了物联网场景而设计的轻量级传输协议<sup>[19-20]</sup>。MQTT 通讯结构图如图 5 所示。

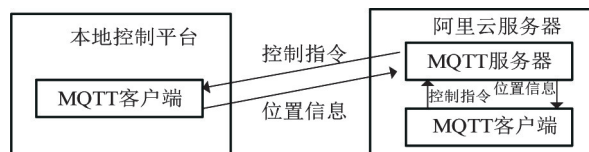


图 5 MQTT 通讯结构图

MQTT 的服务器部署在云服务器上,客户端可以向服务端发布和订阅消息,主要过程如下。

(1) 运动控制器移植 MQTT 库用于消息的发布与订阅程序的设计,使用“arm-linux-gnueabi-hf-gcc”编译器编译 MQTT 源码包获得动态链接库,将动态链接库放入运动控制器的 Linux 系统下。

(2) 消息发布程序用于向云服务器发布消息,

在发布程序中编写 Publish() 函数,该函数主要用于发布数据,待数据发生变化时使用 open() 函数打开文件获得文件描述符 fd, Write() 函数通过 fd 寻找到文件并且写入数据。配置 MQTT 发布的地址、主题、用户、发布质量及密码,通过 mosquitto \_ pub 将数据发布到服务器,消息发布流程图如图 6 所示。

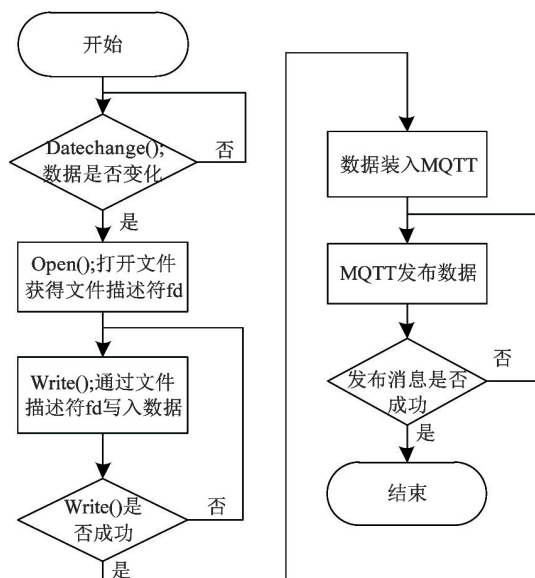


图 6 消息发布流程图

(3) 消息订阅用于接收从服务器发布的消息,在订阅程序中使用 mosquitto \_ lib \_ init 初始化 MQTT 为数据接收做准备,打开共享内存用于把接收的数据放入内存中。使用 mosquitto \_ subscribe 订阅数据,数据订阅成功后放入共享内存方便 Codesys Runtime 获得数据。消息订阅流程图如图 7 所示。

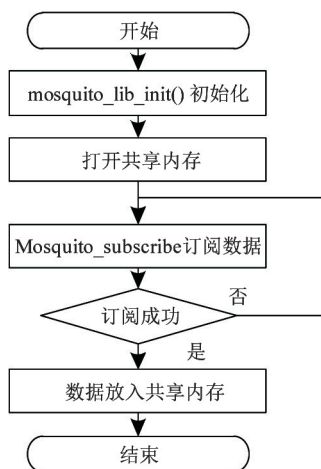


图 7 消息订阅流程图

## 2.3 进程通讯共享内存

Linux 进程间的通信主要有 3 种,即管道通信、消息缓冲通信和共享内存通信。管道通信局限于单向通信,消息缓冲不再局限于父子进程,但是信息的复制需要额外的 CPU 的时间。共享内存利用缓冲区直接交换信息,无需复制,快捷、信息量大。本系统采用共享内存的方法实现 MQTT 客户端和 Codesys Runtime 传递数据。共享内存的创建首先使用 Shm \_ open 创建共享内存,然后 Ftruncate 重新定义共享内存的大小,最后 Mmap 获得共享内存的地址,方便数据的读取和写入。操作结束后使用 Close、munmap 和 shm \_ unlink 关闭和释放共享内存。共享内存创建流程图如图 8 所示。

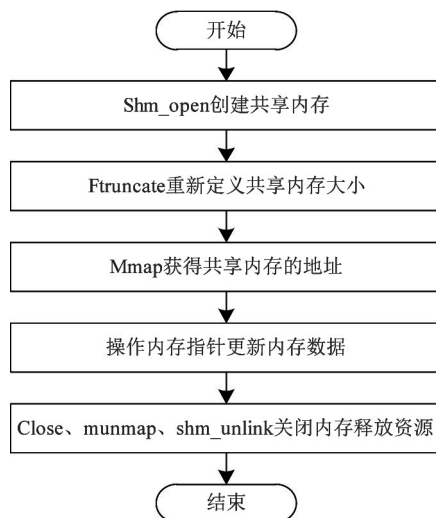


图 8 共享内存创建流程图

## 2.4 建立六轴运动实验平台 Pou 工程

在 Codesys 开发环境下基于 PLCopen 规范采用 ST 语言以及 Codesys SoftMotion 提供的功能块设计六轴实验平台控制程序。在 Codesys IDE 环境下建立一个 Standard Project,在项目中添加运动控制器设备以及 EtherCAT Master 设备,EtherCAT Master 下添加 6 个 LiWei \_ SMD002 \_ X 电机设备用于传递信息给伺服驱动器,之后添加 Pou 单元并使用 ST 语言进行编程设计及控制程序。

Pou 单元由回零和位置跟随两个功能组成,回零部分 Home 采用了 MC \_ MoveAbsoulte 模块实现,通过设定加减速度、速度和执行位使电机运行到指

定的位置。位置跟随部分采用了 SMC\_FollowPosition 模块来实现,模块会根据 fSetPostion 的值使电机运行到相应的位置,所以只需要使 fSetPostion 的值按照正弦变化即可使电机按照正弦运动,fSetPostion 的值变化如式(1)所示。其中  $y_1, y_2, y_3, y_4, y_5, y_6$  分别表示一至六轴的 fSetPostion 值,其中  $t$  为时间,  $\omega$  表示频率,  $A$  表示幅值。  $x_1, x_2, x_3, x_4, x_5, x_6$  表示初始相位角, SMC\_FollowPosition 模块图如图 9 所示。

$$\begin{aligned} y_1 &= A \sin(\omega t + x_1) \\ y_2 &= A \sin(\omega t + x_2) \\ y_3 &= A \sin(\omega t + x_3) \\ y_4 &= A \sin(\omega t + x_4) \\ y_5 &= A \sin(\omega t + x_5) \\ y_6 &= A \sin(\omega t + x_6) \end{aligned} \quad (1)$$



图 9 SMC\_FollowPosition 模块图

### 3 云端 WEB 监控设计

云端 WEB 监控采用 B/S 架构的方式实现,使用

阿里云服务器作为云端监控平台载体。本文 WEB 监控平台主要由前端和后端组成,前端采用原生的 HTML 文本标记语言、CSS 样式和 JavaScript 脚本编写而成。由于传统 WEB 框架过于复杂,并不能满足快速开发的要求, Spring Boot 框架则具有快速和高效的特性<sup>[21]</sup>, 所以后端采用 Spring Boot 框架搭建而成。

#### 3.1 WEB 前端设计

WEB 前端提供了交互界面,用户通过浏览器访问指定的 IP 地址的服务器即可访问到交互界面。WEB 前端界面主要包括两部分,六轴实验平台运行状态监视和远程控制部分。六轴实验平台监视部分用于显示六轴运行时实际位置,其由 Multi Axis Detection System 和六轴曲线图两部分组成。数值显示部分显示各个轴的实际位置数值,曲线图部分显示 6 个轴实际位置随时间的动态变化。六轴曲线图部分使用 Echarts 实现,使用 Echarts.init 初始化 DIV 标签获得 Echarts 图的标识,通过标识符配置横轴、竖轴和 Serial 等参数,纵轴表示实验平台中电机运行时的位置,横轴为时间轴(时间单位为秒), Series 存放了实验平台运行时的各个轴的位置值。通过 setPostion 生成曲线图,数据的更新功能采用定时更换 Series 选项下各个轴的数据实现,由 setInterval 实现定时更新数据的功能,WEB 交互界面如图 10 所示。

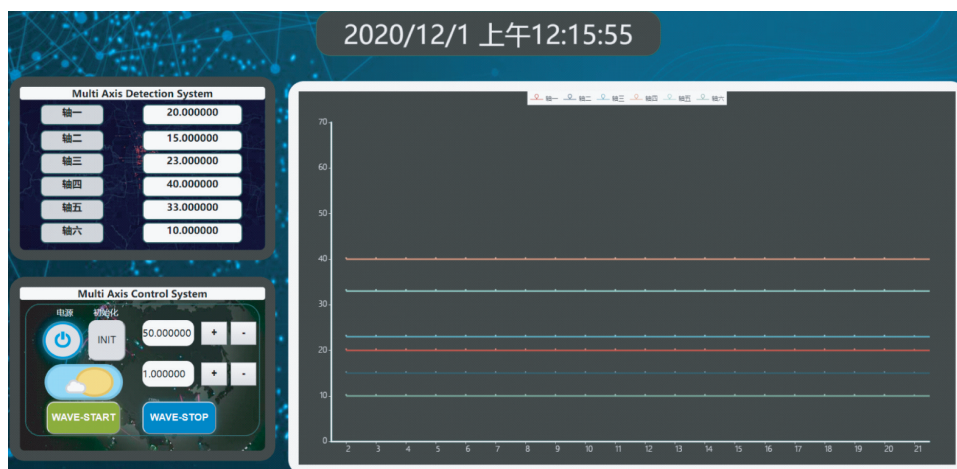


图 10 WEB 交互界面图

#### 3.2 WEB 后端设计

WEB 后端使用 JAVA 语言基于 Spring Boot 框架建立, Spring Boot 是一个基于 Spring 的全新框架,

可简化 Spring 项目的初始搭建和开发过程<sup>[22]</sup>。后端用于处理接收前端发出的 AJAX 请求,并兼顾向 MQTT 服务器发布和订阅数据。后端程序设计流程

图如图 11 所示。后端作为前端与本地控制平台交互的桥梁,根据前端发送 AJAX 请求内容判断是向文件中写入实验平台控制指令还是从文件中读取实验平台的位置数据。

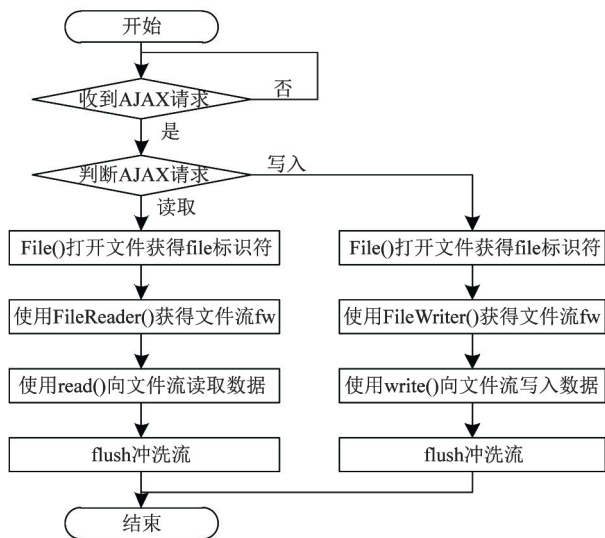


图 11 后端程序设计流程图

## 4 实验数据及分析

以六轴运动控制实验平台为基础,对部署在阿里云的监控系统进行实验测试。

(1)5G 网络相比于 4G 网络具有更大的容量和速率及更低的延时,使用 SpeedManager 软件测试 4G 网络和 NSA 模式下的 5G 网络,结果见图 12。5G 网络比 4G 网络具有更高的下行速度和上行速度,并且 5G 网络延时及抖动相当于 4G 网络的一半。由于当前 5G 网络采用的组网模式为非独立组网(non-stand alone, NSA)在 4G 网络的基础设施上部署 5G 网络,并不能完全体现出 5G 网络的优势,但是独立组网模式(stand alone, SA)需要新建 5G 网络,包括基站、回程链路及核心网,所以 SA 没有普及。

(2)电机的运动控制由 Codesys Pou 项目实现,控制 Codesys Pou 中的 fSetPostion 值规划电机运动。fSetPostion 值由式(1)得到,这里选取幅值  $A$  为 50 cm,频率  $\omega$  选择 1 rad/s,各个轴的初始相位角由  $x_i = 2 \times \pi \times (i \div 24)$  决定, $i$  取值 0~5 分别表示 1~6 轴,设备运行时 6 个电机之间同步组成正弦运动,每个电机都是沿着正弦函数运行,但是初始相位角的不同导致各个轴之间存在相位差,六轴运动实

验平台运行轨迹如图 13 所示,各个电机按正弦运动并且相互之间存在相位差。对比图 13 的 4G 和 5G

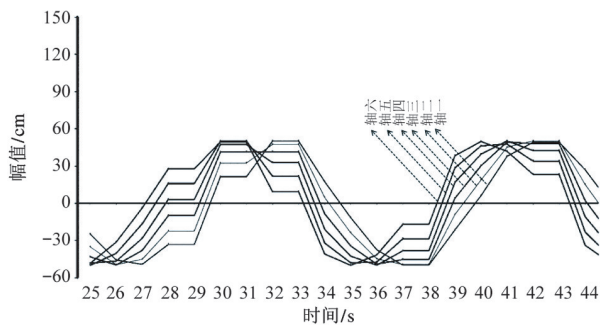


(a)4G LTE 网络参数

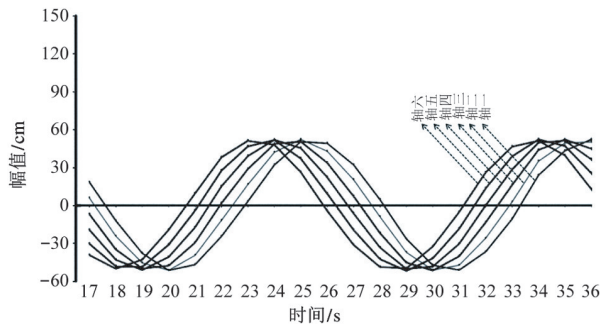


(b)5G NSA 网络参数

图 12 4G 与 5G 网络对比图



(a)4G 网络下的运动轨迹图



(b)5G 网络下的运动轨迹图

图 13 六轴运动实验平台运行轨迹图

网络下的运动轨迹图,得到 4G 网络下轨迹图数据传输的延时较大,时常轨迹图出现失真的情况,只能大致看出实验平台按正弦运动,而 5G 网络下的轨迹图则比较明显,并且传输相对稳定。

在 WEB 监控界面中开启电源并且对六轴运动实验平台进行初始化,运行时轨迹图如图 13 所示。实验平台各个轴的当前位置数值如图 14 所示,显示当前 6 个轴的位置值分别为 49.99388 cm、48.08789 cm、42.90479 cm、34.79780 cm、24.31940 cm 和 12.18367 cm。图 15 为当前六轴运动控制实验平台水晶球的位置,从左往右分别是轴 1 至轴 6,6 个水晶球所处位置与 WEB 界面显示位置一致。

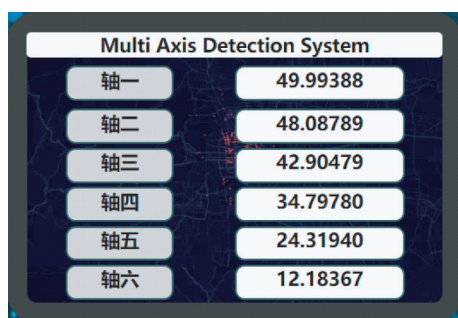


图 14 实验平台运行位置数值显示图

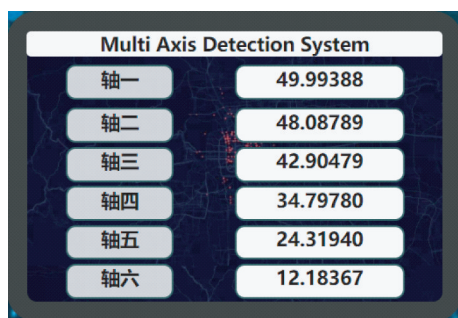


图 15 六轴运动实验平台电机位置图

## 5 结 论

本文采用 5G 通信、MQTT 协议、Codesys IDE、进程通讯技术和前后端分离技术搭建一种基于阿里云的六轴运动实验平台远程监控系统。本地控制平台使用开源的 Linux 系统降低了开发成本,提高了系统移植性,并且采用模块化的设计方式缩短了开发周期,可以在需要额外扩展的时候快速开发。WEB 云端监控采用前后端分离技术进行设计,挂载在阿

里云服务器上。系统针对六轴运动控制实验平台进行了实时数据采集,将数据上传到云服务器,经云服务器后端程序处理后将六轴运动位置数据在前端界面显示,并且前端操作界面可以对六轴运动实验平台进行远程控制。所设计系统满足工业监控系统的基本要求,对开发基于 5G 网络的工业监控系统具有一定的参考价值。

## 参考文献

- [1] ZHANG P, YANG X L, CHEN J X, et al. A survey of testing for 5G: solutions, opportunities, and challenges [J]. *China Communications*, 2019, 16(1): 69-85
- [2] YOU X H, WANG D M, SHENG B, et al. Cooperative distributed antenna systems for mobile communications [J]. *IEEE Wireless Communications*, 2010, 17(3): 35-43
- [3] ADACHI F, PENG W, OBARA T, et al. Distributed antenna network for gigabit wireless access [J]. *AEU-International Journal of Electronics and Communications*, 2012, 66(8): 605-612
- [4] 王浩, 李大智. 基于 5G 网络的城市轨道交通全自动无人驾驶列车控制系统研究 [J]. *信息技术与信息化*, 2020(8): 193-194, 199
- [5] 柳曦, 胡波, 王宾, 等. 5G 通信技术在火电厂的应用 [J]. *热力发电*, 2020, 49(11): 8-13
- [6] REHMANI M H, KANTARCI M E, RACHEDI A, et al. IEEE access special section editorial smart grids: a hub of interdisciplinary research [J]. *IEEE Access*, 2015, 3: 3114-3118
- [7] 应巨林, 曾铮, 曾敏, 等. 基于 5G 技术的农作物视频监控平台的研究与应用 [J]. *农业与技术*, 2020, 40(15): 52-54
- [8] 邓鑫. 煤矿带式输送机运行监控系统设计 [J]. *机械管理开发*, 2020, 35(9): 248-250
- [9] 郭恒瑞, 孟祥印, 孟文, 等. 基于 PLC 的矿用装煤平车线监控系统设计 [J]. *电工技术*, 2020(16): 8-11
- [10] VAN DER WAL E. PLCopen [J]. *IEEE Industrial Electronics Magazine*, 2009, 3(4): 25
- [11] KIM I, KIM T, SUNG M, et al. An open-source development environment for industrial automation with EtherCAT and PLCopen motion control [C] // 2013 IEEE 18th Conference on Emerging Technologies & Factory Automa-

- tion (ETFA), Cagliari, Italy, 2013: 1-4
- [12] 徐建明, 吕汉泰, 张贵军, 等. 基于 Web 的工业机器人 3D 虚拟动态监控系统[J]. 高技术通讯, 2017, 27(3): 254-260
- [13] LIANG Q, LI L. The study of soft PLC running system [J]. *Procedia Engineering*, 2011, 15: 1234-1238
- [14] 张爱民, 孔得鹏, 王倩. 工业机器人的远程监控与诊断系统设计[J]. 机械, 2010, 37(10): 45-47
- [15] 吴丽梅. 基于 B/S 架构的 EPICS 实时监控系统的研究与实现[D]. 上海: 中国科学院上海应用物理研究所, 2019
- [16] 董玉德, 丁保勇, 张昌浩, 等. 基于物联网的路灯可视化监控系统设计[J]. 高技术通讯, 2015, 25(5): 515-523
- [17] 李要中, 刘钧, 马尚昌. 基于 AM335X 的农作物气象自动观测系统设计[J]. 气象科技, 2017, 45(5): 818-824
- [18] 周敏. Nginx[J]. 程序员, 2007, 10: 115-116
- [19] 阎坤, 李涛, 张孝杰. 基于 MQTT 的有序充电平台建设方案研究与应用[J]. 工业控制计算机, 2020, 33(8): 1-3
- [20] 王瑞, 兀玉洁, 李燕苹. 基于 MQTT 协议的物联网实训云平台设计[J]. 工业控制计算机, 2018, 31(9): 101-103
- [21] SALLINEN M, HEIKKILA T, KOSKINEN J. Sensor based flexibility for robotics in manufacturing applications [C] // 2009 IEEE International Symposium on Industrial Electronics, Seoul, Korea, 2009: 1422-1427
- [22] 孙思杰. Web 项目基于前后端分离模式的设计与应用[J]. 科技创新与应用, 2020(27): 96-97

## Design of six-axis motion experiment platform monitoring system based on 5G and Alibaba Cloud

XU Jianming, ZHANG Yuxuan, ZHU Junwei, YU Li

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

### Abstract

A six-axis motion experimental platform monitoring system is designed based on 5G and Alibaba Cloud in this paper, including WEB monitoring platform and local motion control platform. The cloud WEB monitoring platform is built with front-end and back-end separation technology. The front-end is implemented using native HTML text markup language, CSS styles and JS scripts, and the back-end is built by using Spring technology based on the Spring Boot framework. The local control platform is built with embedded system development technology, including 5G module and motion controller. The 5G module as a networking channel is interacted with the motion controller by USB3.0 transmission technology. The motion controller is equipped with MQTT client, process communication mechanism and Codesys Runtime. The messages are published and subscribed to Alibaba Cloud server by MQTT client. The six-axis motion experiment platform is controlled by Codesys Runtime via EtherCAT bus. The MQTT client is communicated with Codesys Runtime through process communication mechanism. Finally, the Codesys Pou project of the six-axis motion experiment platform is created, and the feasibility of the monitoring system is verified by the WEB monitoring interface. The network status under 4G and 5G network in NSA mode is respectively tested by SpeedManager software.

**Key words:** motion controller, MQTT, 5G, Codesys Pou, WEB monitor