

基于Qt5开发的面向工业控制的显控软件

曹策, 贺广健, 付云博, 姜良旭

(中国科学院长春光学精密机械与物理研究所, 吉林 长春 130000)

摘要: 本文介绍了如何使用Qt5在x86计算机上开发面向工业控制的显控软件。本文在Qt Creator中利用C++语言进行了计算机串口通信与网口通信的开发, 并利用Qt Creator内部集成的Qt Designer进行了人机交互界面的设计。本文开发的计算机软件, 通过串口与STM32微控制器完成通信, 并基于自拟的通信协议利用人机交互界面对电磁阀、步进电机、舵机等常用的工业控制模块进行控制; 通过网口与UNIVERSAL ROBOTS协作型机器人完成通信, 并基于TCP/IP通信协议对机器人发送控制指令, 控制机器人的位置与姿态; 通过网口与PLC完成通信, 基于标准的Modbus TCP协议完成对PLC一个工作循环的启停控制。通过本文中显控软件开发的过程可以看出, 使用Qt5可以方便地在x86计算机上开发面向工业控制的显控软件。另外, 开发者还可以自拟通信协议, 进行功能更加复杂的串口、网口通信。

关键词: Qt5; C++; 显控软件; 工业控制; 自动化

中图分类号: TP311.1 文献标识码: A

文章编号: 1009-3044(2020)23-0016-04

DOI: 10.14004/j.cnki.ckt.2020.2454

开放科学(资源服务)标识码(OSID):



Display and Control Software Design for Industrial Control Based on Qt5

CAO Ce, HE Guang-jian, FU Yun-bo, JIANG Liang-xu

(Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun 130000, China)

Abstract: The paper describes how to use Qt5 to develop display and control software for industrial control on x86 computers. The paper uses C++ language in Qt Creator to develop computer serial communication and network communication, and uses Qt Designer which is integrated in Qt Creator to design the user interface. Based on a self-made communication protocol, the software developed in this paper communicates with the STM32 microcontroller through a serial port, and controls common industrial control modules such as solenoid valves, step motors, and servos through the user interface. The software communicates with UNIVERSAL ROBOTS collaborative robot through network port, and sends control commands to the robot based on the TCP/IP communication protocol to control the robot's position and posture. The software communicates with a PLC through network port, and controls the PLC to start and stop based on the Modbus TCP protocol. Through the development process of display and control software in the paper, it can be seen that using Qt5 can easily develop display and control software for industrial control on x86 computers. In addition, developers can also compose their own communication protocols to realize more complex functions through serial and network ports.

Key words: Qt5; C++; display and control software; industrial control; automation

当前我国制造业正快速发展, 智能制造^[1,2]、柔性制造^[3,4]、物联网^[5]等概念已渐渐进入相关研究人员的视野中。高度自动化、智能化的设备正在逐步替代传统的劳动密集型人工流水线, 如生产PCB中常用的AOI(Auto Optical Inspection, 自动光学检测机)^[6]、轻工业中常用的AGV(Automated Guided Vehicle, 自动导航车)^[7]、快递行业中的自动分拣设备^[8]等。在研发此类设备时, 通常使用单片机、微控制器、PLC等作为系统的核心控制部件; 而计算机作为人机交互的媒介, 命令设备按照人的意图完成不同的工作。本文针对这种架构的自动化设备, 介绍了一种基于Qt5的计算机显控软件的开发方法。

Qt5隶属于Qt跨平台开发框架。Qt5中的Qt Creator是一个

开源的、集成的软件开发套件, 仅需开发人员掌握C++语言, 即可开发出高质量的软件。Qt Creator有两个特点: 一是完全面向对象的编程方法; 二是信号与槽的机制。所谓完全面向对象的编程是指在Qt Creator中, 所有的界面控件、计算机外围硬件、系统内部资源等都可以视为“对象”。此处“对象”的概念与C++语言中的“对象”基本一致。因此, 开发人员可以方便快捷的使用计算机、操作系统中的各类资源, 无须具体了解这些资源的底层实现方法。信号与槽的机制是Qt的特有机制, 在Qt5中, 每一个对象都可以发送“信号(Signal)”和执行“槽(Slot)”, 而二者可以通过“连接(Connect)”进行关联, 通过“断开(Disconnect)”取消关联。比如开发人员连接了串口对象的“已接收

收稿日期: 2020-05-22

作者简介: 曹策(1993—), 男, 硕士, 研究实习员。

到数据”信号与界面对象的“显示文本”槽,那么当该信号被触发时(即计算机收到串口数据),界面对象会自动地执行对应的槽(即在计算机屏幕上显示收到的数据)。Qt5支持一个对象拥有多个信号和多个槽,也支持多对多的连接方式。这种机制相对于传统的回调函数的方法,其优点在于参数的类型安全与对象间的松散耦合;其缺点在于槽函数的响应速度远低于回调函数,不过由于现在计算机硬件的高速发展,用户几乎感受不到二者响应时间的差异。

本文使用的框架版本为 Qt 5.6.1,编译套件为 MinGW 4.9.2。运行的平台为标准 x86 计算机,使用的操作系统为 Windows 10。下面本文借助 Qt Creator 作为开发工具,详细介绍人机交互界面、串口、网口的开发方法与关键代码,并在实际项目中进行应用。

1 人机交互界面

人机交互界面是设备操作者与设备之间的桥梁,操作者通过界面控制设备的运作,设备通过界面告知操作者设备的相关信息。因此人机交互界面的设计应该尽量直观、简洁。Qt Creator 中集成了 Qt Designer 模块,在建立 Qt Widgets Application 时,系统默认添加了一个 .ui 格式的文件,这个文件中记录的就是人机交互界面。在 Qt Creator 中左上角的项目文件中双击这个 .ui 文件即可对界面进行编辑。

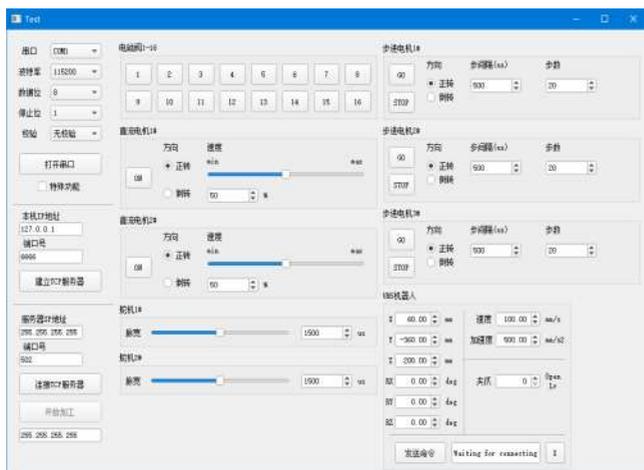


图1 人机交互界面的设计

图1中所示为通过 Qt Designer 创建的人机交互界面的设计,本项目中项目名为 test,使用的基类为 QWidget,派生类为 Widget。人机交互界面使用的控件均为 Qt Designer 中的标准控件。Qt 在创建项目时,在类的声明中创建了一个指向自己的指针变量 Ui::Widget *ui,利用这个指针即可使用 C++ 代码来操作界面中的所有控件对象。例如在界面中创建一个按钮控件,通过右下角的属性面板将其对象名改为 btnTest,然后在 Widget 类的构造函数中添加以下代码:

```
ui->btnTest->setCheckable(true);
ui->btnTest->setText("Test");
```

即可将该按钮设置为自锁按钮,且按钮上显示的文本为 Test。

在实际使用过程中,开发者往往需要点下某个按钮,软件执行相应的动作,此时就需要使用信号与槽的功能将按钮的点击信号与要执行的槽函数进行连接。此时需要先声明一个槽函数,例如在 Widget 类的声明中添加下列代码:

```
private slots:
void mySlotFunc(bool checked);
```

即可声明一个名为 mySlotFunc 的槽函数。

然后进行连接,在 Widget 的构造函数中添加下列代码:

```
connect(ui-> btnTest, SIGNAL(chicked(bool)), this, SLOT
(mySlotFunc(bool)));
```

这是最常用的一种连接方式,这里的 connect 函数有 4 个参数:第 1 个参数为信号发出者的指针;第 2 个为信号, SIGNAL() 是一个能将函数名转化为字符串的宏;第 3 个参数为信号接收者的指针,此处使用 this 指针指向 Widget 自己;第 4 个参数为槽函数, SLOT() 是一个与 SIGNAL() 功能类似的宏。当 Widget 对象进行构建时,程序将执行这一句代码,将指定的信号与槽进行关联。此时用户在界面上点击 btnTest 按钮即可执行 mySlotFunc 函数。在连接信号与槽时,可以发现有些对象的信号是带参数的,这表示这个信号发出的同时,还会给槽函数传递一个该类型的变量。比如上文中的 chicked(bool) 信号,这个 bool 型变量表示的是自锁按钮的状态,槽函数可以根据按钮状态实现不同的功能。开发者可以利用该特点进行信息的传递,实现部件间的通信。

2 串口通信

2.1 串口控制方式

本文在实际项目中使用的串口通信协议为自拟通信协议,主要为了控制电磁阀、直流电机、步进电机以及舵机。

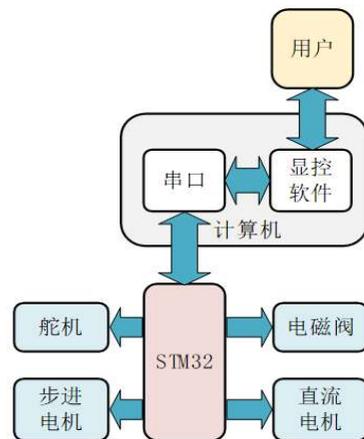


图2 串口控制框图

图2所示为实际项目中的串口控制框图。用户通过计算机上的显控软件操作设备,显控软件将用户的指令按照事先约定的通信协议通过串口发送给 STM32 微控制器。STM32 微控制器根据指令控制电磁阀、电机等模块。

2.2 串口调试

首先为软件添加串口模块,在 Qt Creator 中左上角的项目文件中双击 test.pro,在编辑器中找到 QT += core gui 这一行,在该行的下一行输入 QT += serialport,保存 test.pro 文件。然后在“widget.h”的最上方包含两个头文件:

```
#include <QSerialPort>//串口对象
#include <QSerialPortInfo>//串口信息对象
```

随后在 QWidget 类声明中声明一个公有串口对象及一个公有函数:

```
QSerialPort *serialPort = new QSerialPort(this);
```

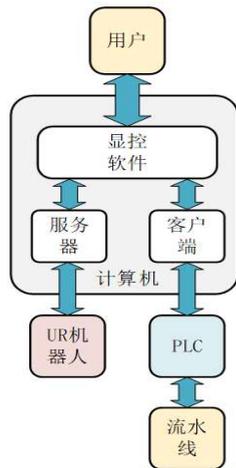



图3 网口控制框图

上图所示为实际项目中网口相关的控制框图。计算机同时建立服务器与客户端,服务器用来控制UR机器人,客户端用来控制PLC。

3.2 网口通信服务器的调试

由于项目中的计算机需要同时作为服务器与客户端与不同的设备通信,因此代码中将每个通信的套接字放在各自独立的线程中以实现并行通信。

类似于调试串口,首先添加网络模块,在test.pro中添加QT += network。然后以QTcpServer为基类派生一个子类robotServer,以下为派生类的声明。

```
class robotServer: public QTcpServer
{
    Q_OBJECT
public:
    robotServer(QObject *parent = 0):QTcpServer(parent){}
    ~robotServer(){}
    QTcpSocket socket;//客户端套接字
protected:
    void incomingConnection(qintptr socketDescriptor){
        socket.setSocketDescriptor(socketDescriptor);}
signals:
    void robotDataReply(QString str);
private slots:
    void serverInit(QString addr, uint port){
        listen(QHostAddress(addr),port);}
    void serverClose(){
        socket.close();
        close();}
    void dataSend(QString str);
    void dataRecv();
};
```

在上述类声明中Q_OBJECT为Qt元对象宏,添加此宏才能使用信号、槽、对象树等Qt特有的功能。serverInit槽用于创建服务器,监听指定的端口。serverClose槽用于断开套接字并关闭服务器。派生类中重新实现了incomingConnection(qintptr socketDescriptor)虚函数,当有新的客户端连接服务器时,系统会自动调用此函数建立套接字。由于实际项目中仅有1个UR

机器人作为客户端与上位机通信,因此类的声明中只声明了1个套接字,即服务器仅能同时连接1个客户端。当然开发者可以声明多个套接字用来同时连接多个客户端。由于篇幅所限,上述声明中并没有对收发函数进行实现,其实现过程与串口收发类似。另外类中还声明了一个信号用来与UI界面通信,将机器人回传的信息显示在UI上。

在构建好服务器后,新建一个子线程。以QThread为基类派生一个子类serverThread。派生类的目的是重构QThread基类中的虚函数run()。以下为派生类的声明。

```
class serverThread: public QThread
{
    Q_OBJECT
public:
    serverThread(QObject *parent = 0):QThread(parent){}
    ~serverThread(){
        requestInterruption();
        quit();
        wait();}
protected:
    void run(){
        robotServer server;
        connect((Widget *)this->parent(), SIGNAL(serverOpen
(QString,uint)),&server,SLOT(serverInit(QString,uint)));
        while(!isInterruptionRequested()){
            QCoreApplication::processEvents(QEventLoop::All
Events, 100);//保持线程的事件循环
            msleep(1);}
        };
```

虚函数run()相当于线程的主函数,重新实现此函数相当于对线程的主函数进行编程。另外在Qt中,任何子线程都禁止访问UI界面中的对象,但可以通过信号和槽的方式与UI通信,间接操作UI界面中的对象。析构函数中的requestInterruption()与线程主函数中的isInterruptionRequested()配合使用,可以保证线程安全退出。

完成上述两个对象的派生后,在Widget对象的声明里添加一个子线程对象,代码如下:

```
serverThread *robotThread = new serverThread(this);
在Widget对象的构造函数里启动线程,代码如下:
robotThread->start();
```

Qt5中的信号与槽也支持跨线程的连接方式,例如上述上文run()函数中的连接方式。这种连接方式是利用了Qt对象树的特性,由于Widget对象(即UI界面)创建robotThread对象时,使用this指针声明了父子关系,因此在子线程内,用this->parent()返回的指针即为Widget的对象指针。

在实际项目中,使用UR机器人作为客户端成功与本文构建的服务器相连接。通过发送特定的控制指令实现了对UR机器人的远程控制。

3.3 网口通信客户端的调试

客户端的调试与服务器的调试过程类似,只需从QTcpSocket派生一个子类即可,创建线程的方法与创建服务器线程完全一致,此处不再赘述。(下转第26页)

4.3 先进的综合评价方法

目前,主要的评价方法还是多层次分析方法和模糊综合评价方法,这些方法虽然可以综合考虑图书的多指标对图书进行综合评价。但从可查文献看,都只是利用一些简单的综合评价方法,如模糊综合评价方法中,评价模糊都是极其简单,且权值都是设定为常值,这样并利于图书的综合评价。因此,还需更深入地挖掘和开发模糊信息处理在语言处理、综合决策等方面的优势,研究更加先进的模糊综合评价方法。

5 结论

本文对当前国内外图书评价的研究现状进行了总结。首先,对当前图书的评价指标体系进行总结;再次,对当前的图书评价方法进行总结,包括图书评论、单指标评价方法和多指标综合评价方法;最后,对科学评价图书提出三点建议,以期推进我国图书评价向前发展,对图书进行客观公正的评价,更好地为读者服务。

参考文献:

- [1] 何峻. 我国图书评价现状分析[J]. 大学图书馆学报, 2012(3): 107-111.
- [2] 代根兴, 周晓燕. 出版社与图书馆图书评价标准比较研究[J]. 大学图书馆学报, 2013(5):38-42.

- [3] 王国权, 黄海洋, 张剑. 基于模糊模式的图书质量识别方法仿真研究[J]. 计算机仿真, 2011,4(3):189-192.
- [4] 梁铨华. 基于读者反馈意见的图书评价系统的设计与实现[J]. 科技情报开发与经济, 2010,20(15):37-38.
- [5] 孙晓慧. 图书综合评价指标体系构建的研究[M]. 东北师范大学硕士学位论文, 2014.
- [6] 王绍平, 陈兆山, 陈钟鸣, 等. 图书情报词典[M]. 汉语大词典出版社, 1990.
- [7] 刘亚斐. 基于层次分析法的旅游官网评价指标体系构建研究[D]. 长春:东北师范大学, 2013.
- [8] 邱燕燕. 基于层次分析法的网络信息资源评价[J]. 情报科学, 2001,19(6):599-602.
- [9] 吕殿秋. 基于模糊层次分析法的网络信息资源评价研究[D]. 哈尔滨:黑龙江大学, 2010.
- [10] 李良群, 李小香. 研究生课程教学质量的直觉模糊综合评价[J]. 电脑知识与技术, 2016,12(6):138-141.
- [11] 滕居特, 顾幸生. 研究生教育评估的多级变权模糊综合评判[J]. 华东理工大学学报(自然科学版), 2006,32(9):1121-1125.
- [12] 李小香, 李良群. 一种基于模糊理论的图书质量评价新方法[J]. 电脑知识与技术, 2020,16(10):198-200.

【通联编辑:王力】

(上接第19页)

在实际项目中,本文构建的客户端成功与PLC相连接。通过标准的Modbus TCP协议读、写PLC的线圈状态,完成对PLC流水线的控制。

4 结论

(1)使用Qt5可以在x86计算机上开发面向工业控制的显控软件。使用Qt5自带的QSerialPort类可以完成计算机串口通信、Modbus串口通信;使用QTcpServer类、QTcpSocket类以及他们的派生类可以完成TCP网络通信、Modbus TCP网络通信;使用QThread的派生类可以进行多线程操作。使用这些Qt5标准类进行软件开发,能快速实现预期功能而不必关心具体实现方法,从而大幅提高软件开发的效率。

(2)本文使用Qt 5.6.1和C++语言编写的显控软件在实际项目中成功控制了16个电磁阀、2个直流电机、3个步进电机、2个舵机、1个UR5协作型机器人以及某型流水线。且该软件还可以继续扩展,实现对更多工业模块的复杂控制,进一步提高系统的自动化程度。

参考文献:

- [1] 张肃, 许慧, 黄蕊. 我国人工智能产业发展问题研究[J]. 长春理

工大学学报, 2018,31(5):1-6.

- [2] 钟义信. 人工智能:概念·方法·机遇[J]. 科学通报, 2017,62(22): 2473-2479.
- [3] 倪珉. 柔性制造系统在中小企业中的应用和未来的发展[J]. 机械管理开发, 2018,33(8):227-229.
- [4] 任永杰, 尹仕斌, 郝继贵. 面向现代柔性制造的工业机器人高精度控制方法[J]. 航空制造技术, 2018(5):16-21.
- [5] 王锐, 吴紫薇. 基于物联网技术的列车环境监测系统设计[J]. 长春理工大学学报(自然科学版), 2019,42(3):133-137.
- [6] 马晓波. 一种印刷电路板(PCB)AOI假缺点的修复方法[D]. 苏州: 苏州大学, 2016: 1-7.
- [7] 贺冰倩. 仓库智能AGV路径设计及调度研究[D]. 武汉: 华中科技大学, 2019: 16-19.
- [8] 吴星峰. 小包邮件自动分拣系统的设计与实现[D]. 长春: 吉林大学, 2015: 1-3.
- [9] 戴振民. 基于Qt5的MODBUS协议编程与实现[J]. 电子技术与软件工程, 2018(17):55.
- [10] 罗旋, 李永忠. Modbus TCP安全协议的研究与设计[J]. 数据采集与处理, 2019,34(6):1110-1117.

【通联编辑:梁书】