

Linux 的 QT 编程技术 在卫星激光测距中的应用*

秦 思^{1,2} 张忠萍¹ 张海峰¹ 吴志波¹

(1. 中国科学院上海天文台, 上海 200030; 2. 中国科学院研究生院, 北京 100049)

提 要

Linux 操作系统具有良好的稳定性和安全性,在系统控制中得到广泛应用。QT 是 Linux 操作系统下一种优秀的可视化编程工具,具有良好的图形界面和可移植性。该文介绍了在 Linux 下利用 QT 编程技术,采用面向对象编程方法,开发的基于 QT 的卫星激光测距控制软件,重点论述了用 QT 编程对事件计时器、距离门产生器、系统实时时钟模块以及所涉及的计算机并口和串口等硬件设备的控制方法。目前该控制软件已在卫星激光测距工作中成功应用。

主题词: Linux 操作系统 — QT 编程工具 — 卫星激光测距 — 控制软件

分类号: P228.5, TP314

1 引 言

Linux 操作系统是基于 Unix 发展起来的,由于其众多优秀的技术特点,受到广大用户的青睐,在各行业中得到广泛的应用。

Linux 具有许多特点和优势,尤其表现在它良好的安全性和稳定性上。Linux 采取了许多安全技术措施,包括读/写权限控制、带保护的子系统、审计跟踪、核心授权等^[1]。最重要的是 Linux 开放的源代码,保证了任何系统漏洞都能被及时发现和改正,大大降低了病毒入侵以及黑客攻击的概率,提高了系统安全性。经相关实验验证,Linux 操作系统同大型操作系统一样具有相同的可靠性。此外,Linux 具有良好的图形用户界面,能够为用户呈现一个直观、易操作、交互性强、友好的图形化界面。

卫星激光测距(Satellite Laser Ranging, SLR)控制软件是激光测距系统的枢纽,负责控制和连接外部设备,使得各个设备按一定顺序进行工作,其主要的控制和连接对象包括激光器、事件计时器、距离门控制电路以及系统时钟模块等。Linux 操作系统能够有效地提高卫星激光测距控制软件系统的安全性、稳定性,还为用户提供友好的图形界面,能满足激光测距对操作系统的要求。而且随着桌面应用的发展,Linux 图形用户界面将更友好,为激光测距控制软件的发展提供了广阔的空间。本文拟介绍 Linux 操作系统下的 QT 编程语言以及在该平台下设计和开发的卫星激光测距控制软件,其中对事件计时器、距离门产生器、系统实时时钟模块以及

所涉及的计算机并口和串口等硬件设备的控制方法进行了重点介绍。

2 QT 编程语言

Troll Technologies 公司开发的 QT 库是一个跨平台的 C++ 类库,具有丰富的控件资源和良好的可移植性。尤其是 Qt Designer 图形界面设计工具为软件开发者提供了图形用户界面工具箱,大大简化了图形用户界面的开发和维护任务。本文以 Redhat9.0 操作系统为开发环境,使用 QT3.0 开发库,采用面向对象编程模式,结合 QT 特有的信号与槽机制进行相关功能模块开发。

QT 的信号和槽是一种高级接口,它应用于对象之间的通信。信号和槽能携带多种类型的参数,当某个对象的状态改变时,信号就由该对象发出并由相应的槽(功能函数)接收,进行相应的操作处理。信号和槽可与多个对象连接,且可封装成一个整体模块,这种信息的封装,能确保多个对象被当作一个真正的多功能组件来使用。

信号与槽的关联可以通过调用 QObject 对象的 connect 函数来实现^[2]。其定义形式如下:

```
bool QObject::connect(QObject:l = sender,Signal,QObject:l = receiver,Slot);
```

其中,Signal 为信号;Slot 为槽,即定义的功能函数。当信号到来时,就会调用 Slot 中的函数,实现某种操作。

3 QT 编程技术在卫星激光测距中的应用

QT 编程语言高效、灵活的执行代码适合于对速度和应用程序之间协同性要求较高的场合,卫星激光测距控制软件需定时采集、处理、数据输出及图形实时显示,测距实时性的要求相对较高。QT 开发工具提供了类库开发框架以及很多控件,采用面向对象编程思想,模块化操作,既简化整个开发过程又提高了软件的可读性、可移植性,适合于激光测距控制软件的开发。

3.1 卫星激光测距控制软件的整体功能描述

控制软件作为测距系统枢纽,负责控制激光器、事件计时器、系统实时时钟、距离门产生器等设备,使它们按照一定的顺序开展工作,同时实时显示测距情况,以判断是否测量到卫星。图 1 为上海天文台 SLR 控制模块图,控制计算机是测距系统的核心;时钟信号 10MHz 和秒信号由 GPS 接收机提供;系统实时时钟用于维持测距系统时间和激光点火信号的产生;事件计时器将主波信号和回波信号看成事件,将其发生时刻精确记录下来^[3];距离门发生器用于产生门控信号,使探测器打开,接收回波信号。

3.2 控制软件流程

图 2 为 SLR 控制软件工作流程。测距开始时,控制软件首先获取当前时刻,根据预报数据计算卫星的方位和高度,并传递给伺服控制部分,使望远镜指向所选择的卫星;控制软件获取激光发射时刻(由事件计时器记录),由预报数据获得卫星距离预报值,计算距离门控值,控制光子探测设备,接收回波信号。待获取激光回波时刻后,根据主回波匹配算法获得卫星距离的测量值,然后计算出距离残差值($O - C$)。在滤除部分噪声后将测距结果保存,同时实时显示残差值可让操作人员判断是否打中卫星以便对望远镜的指向进行微调。上述操作过程直到观测结束而终止。

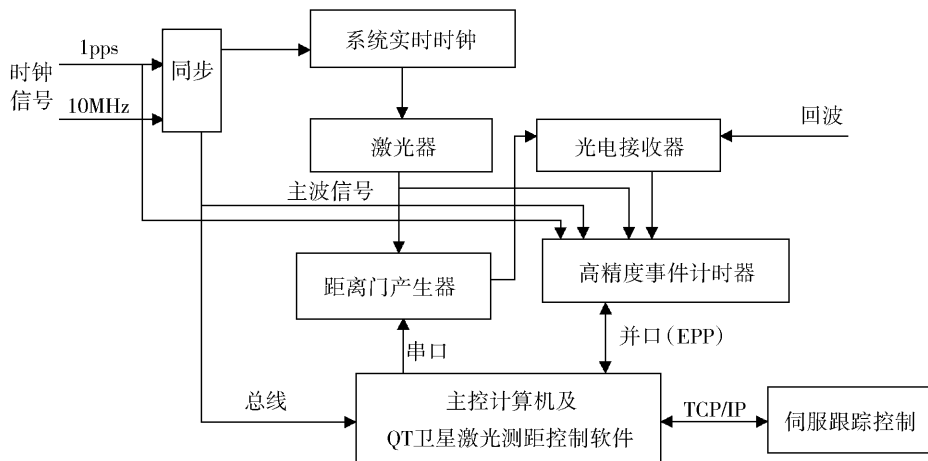


图 1 卫星激光测距控制模块

Fig.1 The control module of SLR

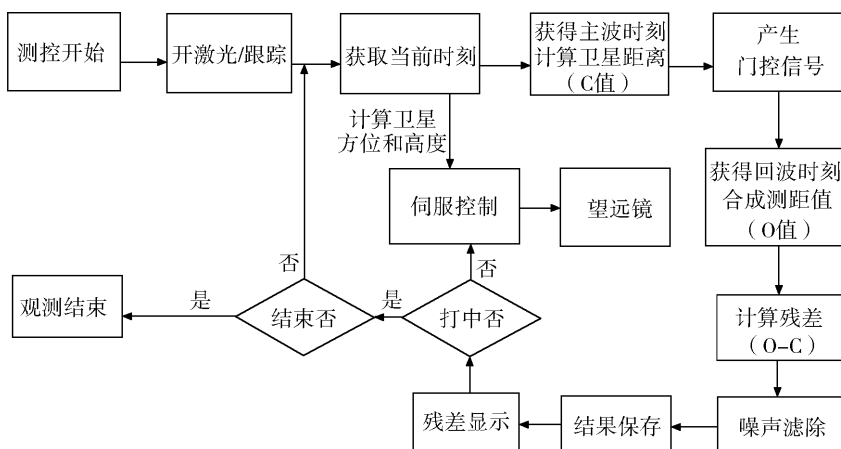


图 2 激光测距控制软件工作流程图

Fig.2 The controlling chart of SLR control software

3.3 系统实时时钟

在卫星激光测距中,时钟系统起着至关重要的作用,它不仅维持整个测距系统的时间,还要实现激光点火信号控制及控制软件的时序控制。本系统采用了 4 个 16 位的减法计数器,利用外部提供的 10MHz 时钟信号进行工作,为系统提供精确的实时时钟,4 个计数器级联方式如图 3 所示。

计数器 1 的时钟信号为 10MHz,计数器 2,3,4 的时钟信号分别为上一级计数器输出的时钟信号,其中计数器 2 的输出时钟信号还作为激光点火信号。以 20Hz 激光测距为例,控制软件对计数器 1,2,3,4 的初始值分别设置为 10,50000,40,50000,计数器 1,2 负责记录 50ms 以内的时间,计数器 3,4 负责记录整 50ms 的时间,两者结合来实现系统时间的维持。按照上述计数器初始值的设置,其能够维持的最长时间约为 28h。控制软件通过读取各个计数器的值

来获得当前时刻,进而开展图 2 中所示的一系列工作。

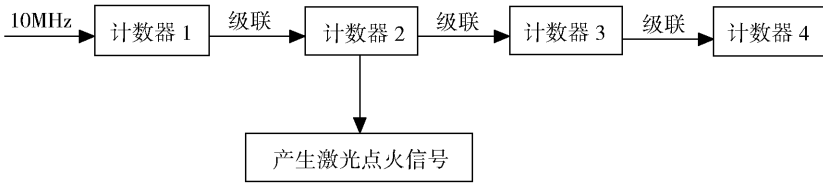


图 3 计数器连接示意图

Fig. 3 The sketch map of counters connecting

3.4 事件计时器控制

事件计时器作为激光测距系统中的一个重要设备,负责记录主波与回波的时刻。控制软件根据卫星预报轨道所推算出的激光脉冲往返时间间隔,可以识别出相关的主波和回波各自的时刻,其时刻差即为激光往返时间,进而获得卫星观测距离值^[4]。

事件计时器与主控计算机通过增强型并口(Enhanced Parallel Port, EPP)进行数据交互。操作之前首先进行计算机并口初始化,再经过内部校准、时间同步检测后方可开始测量。在有主回波到达时,就可精确记录主回波事件的发生时刻,之后进行数据采集,这一系列工作均由控制软件对并口进行读写来完成。

Linux 操作系统对底层端口操作采取了保护措施,并设定了访问级别,对于一般用户的 I/O 端口访问级别定义为 0。为了实现计算机并口的操作,需要提高控制软件的 I/O 访问权限级别。采用了如下功能函数实现控制软件对端口访问权限级别的修改:

```
int iopl(int level)
```

其中参数 level 代表新的 I/O 访问权限,范围是 1 ~ 3。其中 3 代表最高访问权限。在本软件中,level 取为 3。

控制软件中对事件计时器的所有操作封装成一个 Qt 类对象,如下:

```
class EPP : public EPPDlg{
public:
    EPP( QWidget* parent = 0, const char* name = 0 );
    ~EPP();
    int StartSync ( double *time);           // 时间同步
    int ReadET2();                          // 读取 ET 采集到的数
    int CheckBuf();                          // 检查 ET 的 buffer 是否为空
    void Initial();                          // 初始化 ET
    QString Calibration();                   // 系统校准
    void Start_Measure();                    // 开始测量
    ..... }
```

在对 I/O 端口进行读写时, Linux 提供了 inb()、inw()、outb()、outw()、inb_p()、inw_p()、outb_p()、outw_p() 等功能函数,根据所写入和读取的数据长度采用不同的函数,这些函数的参数为物理端口或读写数据。例如在事件计时器类对象的 Initial() 函数中使用了以下代

码进行并口初始化:

```
s = inb(PORT + 0x402); // PORT 为计算机并口基地址,通常为 0x378
s = s & 0x9F;
s = s | 0x80;
outb(s,PORT + 0x402); // 向端口 PORT + 0x402 写数据 s
s = inb(PORT + 4); // 从端口 PORT + 4 读取数据,赋予 s
```

上述类对象实现了在 Linux 下对事件计时器的操作,包括初始化、内部校正、时间同步及数据采集,且具有较好的移植性,可在其他 QT 应用软件中直接应用。

3.5 距离门控制

在卫星激光测距中,距离门控技术是一种常用且有效的抑制噪声的方法。测距时,控制软件根据卫星预报距离,推算出激光信号往返时间(即距离门控长度),然后将此数据置入到距离门控电路中的计数器。待主波信号到达后计数器开始递减,当递减到设定值后产生信号给探测器,使探测器打开。

本文所使用的距离门控制电路数据接口为计算机串口,为此我们在 Linux 下开发了串口通讯程序。采用了 QT 类封装机制将串口操作封装成独立的类对象,包括对串口的打开、波特率设置、读写数据,关闭等操作。

串口操作类对象结构如下:

```
class CommPort {
public:
CommPort();
~CommPort();
void setNewOptions(int m_fd,int baudrate, int databits, const QString& parity, const QString&
stop, bool softwareHandshake, bool hardwareHandshake); // 设置串口参数
int portOpen( QString strComm,QString paraSetting ); // 打开串口
int com_Write(int fd,unsigned char ch[ ],int lens); // 将数据写入串口
int com_Write(int fd,QString strWrite); // 将数据写入串口
void closePort(int fd); // 关闭串口
.....}
```

此串口类对象独立性较强,其类成员函数中采用的是 Linux 操作系统内部函数,可以应用在 Linux 系统下其他 QT 或非 QT 应用程序中,使用灵活性较高。

3.6 控制软件及测量结果

2008 年 12 月上海天文台完成了 Linux 操作系统下卫星激光测距控制软件的设计和开发,并成功开展了对卫星的实际测量。图 4 为控制软件界面及实时距离残差测量图,图中弥散点为噪声,比较密集的点为有效回波信号,中间的一条虚长直线为距离残差零基准线。

从图 4 可以看出,本控制软件所采用的 Qt Designer 工具可设计出友好、直观的操作界面,且易于操作。实验结果表明,Linux 操作系统下基于 QT 编程语言开发的控制软件可以实现卫星激光测距观测任务,在随后的运行过程中也表现出了良好的稳定性,证明了本文所采用的控制软件设计方法是可行的。

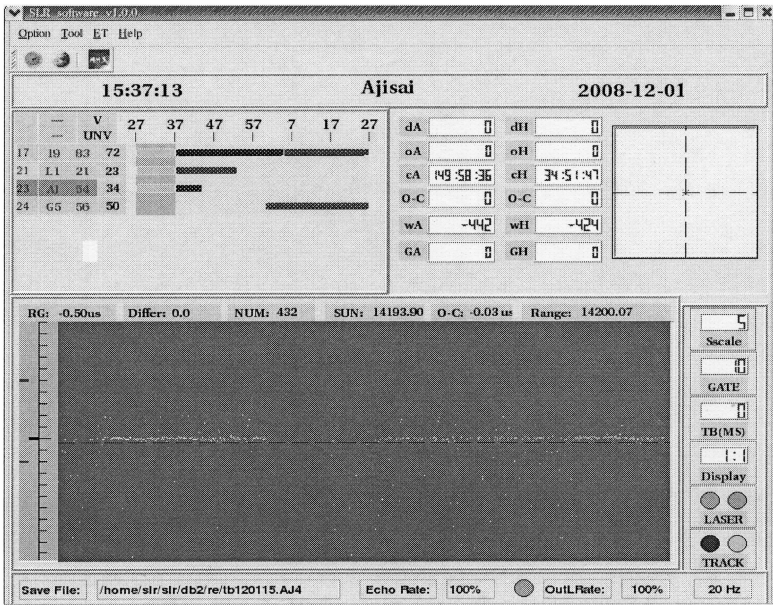


图 4 Linux 下卫星激光测距控制软件界面

Fig. 4 The interface of SLR control software developed by QT in Linux OS

4 总 结

QT 作为 Linux 操作系统下可视化图像开发工具,它为用户提供了优秀的图形化控件以及功能齐全的操作模块,大大简化了开发过程,提高了效率。本文设计和开发的控制软件采用了面向对象编程模式,使得软件的优化和升级都较易实现,且可以运行在通用的、包含 QT 软件开发库的 Linux 操作系统平台上。卫星测量实验结果表明,该控制软件可以实现卫星激光测距的测量任务,而且系统的稳定性、安全性也有了一定的提高。QT 编程技术在激光测距中的成功应用,为在 Linux 操作系统下开发其他图形化操作软件及底层端口操作提供了经验。

参 考 文 献

- [1] 李永权. Linux 操作系统应用前景, Intelligence, 2008, 02: 162
- [2] Blanchette J, Summerfield M 著. 齐亮译. C++ GUI Qt3 编程, 北京: 北京航空航天大学出版社, 2006
- [3] 吴志波. 高重复率人卫激光测距的研究(硕士学位论文), 上海: 中国科学院上海天文台, 2006, 37
- [4] 吴志波, 张忠萍, 孙宝山等. 中国科学院上海天文台年刊, 2008, 29: 179

THE APPLICATION OF QT PROGRAMMING TECHNOLOGY UNDER LINUX IN SATELLITE LASER RANGING

QIN Si^{1,2} ZHANG Zhong-ping¹ ZHANG Hai-feng¹ WU Zhi-bo¹

(1. Shanghai Astronomical Observatory, Chinese Academy of Sciences, Shanghai 200030;

2. Graduate School of the Chinese Academy of Sciences, Beijing 100049)

Abstract

Linux Operating System is widely used in system control because of its stability and security. QT is one excellent programming toolkit under Linux OS with good Graphical User Interface and naturalization. This paper introduces QT programming technology under Linux OS and the SLR control software developed by QT toolkit with object oriented method. The operating methods of event timer controlling, range gate generator controlling, system real-time clock, computer parallel port and serial port are mainly discussed. By now this control software is successfully applied to SLR.

Key words Linux Operating System — QT toolkit — Satellite Laser Ranging — control software