

引用格式:李鹏利,李兴成.基于GNU Radio的雷达信号处理系统设计实现[J].电光与控制,2021,28(5):75-78,102. LI P L, LI X C. Design and implementation of radar signal processing system based on GNU radio[J]. Electronics Optics & Control, 2021, 28(5):75-78, 102.

# 基于 GNU Radio 的雷达信号处理系统设计实现

李鹏利, 李兴成  
(空军工程大学,西安 710051)

**摘要:**针对现有雷达仿真系统的单线程运行模式在处理较大数据量或多任务执行时会造成CPU负担加重、导致系统运行速度缓慢与数据丢失,进而影响仿真测试的实效性等问题,提出了基于GNU Radio软件平台利用其多线程技术与模块化思想,同时设计并使用传输内存地址的方法来搭建雷达信号系统,提升计算机的资源利用率。多线程模式下的系统运行不仅能够对大量数据进行处理,而且能提升其处理数据的速度,同时兼具美观性与操作性。最后的测试数据表明了雷达信号处理系统的有效性,达到了预期的设计目标,满足了实际应用需求。

**关键词:** 雷达信号处理; GNU Radio 软件平台; 多线程; 模块化; 传输内存地址

中图分类号: TN95; TP311 文献标志码: A doi:10.3969/j.issn.1671-637X.2021.05.017

## Design and Implementation of Radar Signal Processing System Based on GNU Radio

LI Pengli, LI Xingcheng  
(Air Force Engineering University, Xi'an 710051, China)

**Abstract:** When the single-threaded operating mode of the existing radar simulation system is used to process a large amount of data or execute multiple tasks, it will increase the burden of the CPU, result in slow system operation and data loss, and affect the effectiveness of the simulation test. To solve the problems, a radar signal system is proposed based on the GNU Radio software platform by using its multi-threading technology and modular ideas, and the method of transmitting memory addresses is designed and used to establish the radar signal system, and improve the resource utilization of the computer. System operation in multi-threaded mode can not only process a large amount of data, but also increase the speed of data processing. The final test data shows the effectiveness of the radar signal processing system, which achieves the expected design goals, and meets the actual application requirements.

**Key words:** radar signal processing; GNU Radio software platform; multi-threading; modularization; transmission memory address

### 0 引言

随着雷达愈发的多功能化,雷达系统的设计与实现愈加复杂,由于实际雷达测试与研发的局限性,软件化雷达开始受到重视,其仿真测试的灵活性与简便性的优势也开始显现出来。雷达信号处理技术<sup>[1]</sup>作为雷达系统的核心技术,通过对雷达信号处理过程的分析,能够更好地了解雷达系统。

目前现有的雷达仿真系统在数据处理时采用单一

顺序处理的单线程模式,且不同参数输入往往在一个布局中实现,导致仿真界面布局杂乱。其虽在时序控制方面具有一定优势,但在处理较大的数据量或多任务执行时,会对实时仿真造成一定的影响,文献[2]指出了单线程系统存在的一些问题。本文基于GNU Radio<sup>[3-4]</sup>软件平台来设计开发雷达信号处理系统,利用其底层的多线程技术,并设计传输二级栈内存地址的方式来弥补单线程雷达仿真系统存在的主要缺陷,同时有效减少了对栈内存的占用率并实现了对内存地址的有效管理。多线程模式使多核计算机的数据处理更具效率性与可靠性,文献[5]指出了多线程系统在实时性方面的优势,系统同时使用面向对象的编程思想<sup>[6]</sup>与软件模块化思想<sup>[7]</sup>,即让每个模块都是一个完整的系统对象,使系统布局更

收稿日期:2020-03-05 修回日期:2021-04-19

基金项目:国家自然科学基金(61971438)

作者简介:李鹏利(1994—),男,山西阳泉人,硕士生,研究方向为雷达信号处理与信息处理技术。

具美观性。最后对系统进行测试,所得到的结果表明系统设计解决了存在的问题。

### 1 GNU Radio 软件平台

#### 1.1 GNU Radio 软件平台架构

GNU Radio 软件平台内的编程是基于 Python 脚本语言<sup>[8]</sup>和 C++ 混合方式<sup>[9]</sup>进行的。利用 C++ 静态语言具有的高效运行特性,来编写各种雷达信号处理底层模块,如脉冲压缩/匹配滤波器、MTI 和 MTD 等典型模块,运行速度快,且用户不需要理解其内部原理,只需输入对应的系统参数。在 GNU Radio 软件平台中把这种能发挥特定作用的模块称为 Block。Python 作为一种语法简便的动态“胶水”语言,强大的第三方库与快速的开发效率是其显著特点,通过它连接搭建雷达信号处理系统所需的 Block,组成完整系统,使该系统具有美观性与操作简便性,在 GNU Radio 软件平台中把完整的信号处理系统称为流图(flow graph)。

由于 Block 使用 C++ 语言编写,而流图则使用 Python 语言编写,所以在 GNU Radio 软件平台中的“桥梁”— SWIG (Simplified Wrapper and Interface Generator)<sup>[10]</sup>就起到了连接两种语言接口的作用,使在 Python 中能够直接导入由 C++ 编写的 Block 并调用相关 Block 来完成系统搭建。图 1 所示为 GNU Radio 软件平台架构<sup>[11]</sup>。

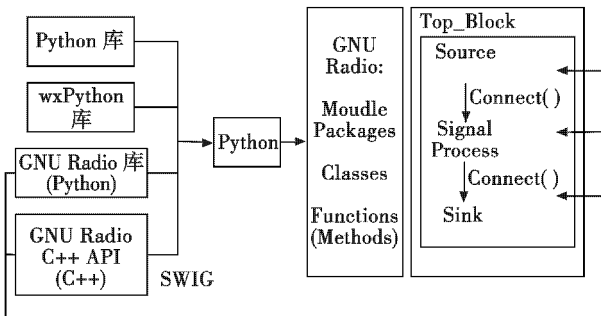


图 1 GNU Radio 软件平台架构图

Fig. 1 Architecture diagram of GNU Radio software platform

#### 1.2 模块(Block)结构

当编写相应的雷达信号处理模块时,主要编写 3 个文件,即 .h, \_impl.h 与 \_impl.cc 文件。其中 .h 文件作用为定义数据传输、系统变量、调度器等的相关接口与指针;\_impl.h 与 \_impl.cc 文件用来对信号处理算法中的相关变量与函数进行定义与实现,并连接相关接口、指针,使其能执行相应的信号处理的功能。

在编写 Block 的 \_impl.cc 文件时需注意两个最主要的问题。

1) 根据数据是基于项目进行传输,所以在模块内需定义流经数据接口的最小和最大的项目数以及每个

项目的字节数。数据流结构为

`io_signature::make(1,1,sizeof(gr_complex))`//输入流  
`io_signature::make(1,1,sizeof(gr_complex))`//输出流  
其中:数据接口标注为输入与输出的最小与最大项目数均为 1,即每次流经该接口的数量为 1 个项目;`gr_complex` 为 GNU Radio 软件平台中专门定义的数据类型,与 C++ 中的复数类型相同(`complex <float >`)。

2) 另一个为编写文件内的 `general_work()` 函数,作为 Block 设计的核心函数,是一个纯虚函数,在编写 Block 时必须对其进行重写才能使其发挥作用。纯虚函数结构为

```
general_work(int noutput_items,
             gr_vector_int & ninput_items,
             gr_vector_const_void_star & input_items,
             gr_vector_void_star & output_items)
```

其中:`noutput_items` 表示输出的项目个数;`ninput_items` 表示输入的项目个数;`input_items` 与 `output_items` 分别表示指向输入流与输出流的指针向量。

### 2 软件设计

#### 2.1 数据传输方式设计

在 GNU Radio 软件平台中,当数据基于项目的方式在模块间以数据流的方式进行传输时,一个模块的输出即为一个项目。上一个模块产生的数据,由 GNU Radio 软件平台底层设计进行调度并传输到信号处理模块的缓存区,再对其数据量进行判断,当输入口的缓存区内的数据量满足模块的信号处理量时,模块对数据进行相应的处理,并传入输出口的缓存区。这个过程通过一个写指针 `W_ptr` 来输出数据,它的下游模块会使用读指针 `R_ptr` 读取数据来实现,指针所写入与读取的数据即为项目。数据传输方式如图 2 所示。

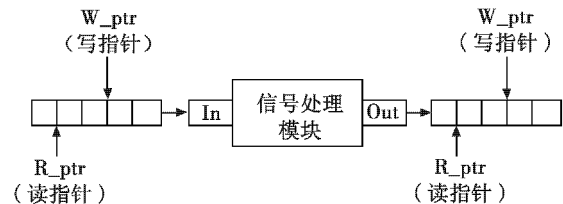


图 2 数据传输方式

Fig. 2 Data transmission mode

在数据传输过程中,处理的数据量较大,存在几千甚至上万个采样点,若依旧采用 GNU Radio 软件平台内传统的数据传输方式,长时间运行则可能会造成内存泄漏,从而导致程序崩溃。为了达到能完整传输数据并减小栈内存占用率的目的,设计了传输保存有数据的内存地址的方法。通过传输内存地址,并在模块内部设计读取机制来达到传输数据的目的。数据传输

设计如图3所示。

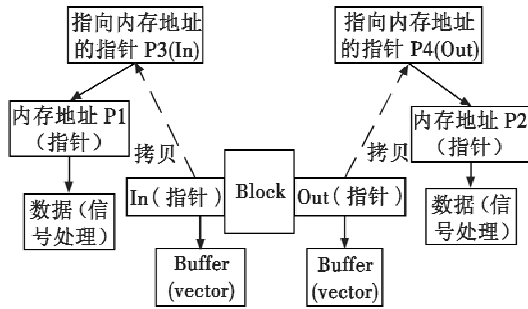


图3 数据传输设计图

Fig. 3 Data transmission design

图中,信号处理模块的输入(In)与输出(Out)是指向缓冲区(Buffer)的指针,指针(P1)指向信号处理所产生的数据,内存地址(P3)保存内存地址(P1),通过把指针(In)拷贝到指针(P3)中,使输入(In)指向内存地址(P1),而不是再指向缓冲区(Buffer),来达到传输内存地址的目的,在输出(Out)处与输入(In)设计相同。通过传输二级内存地址来减小对栈内存的占用,不仅能够有效提升对计算机资源的利用率来提升处理数据的速度,同时方便了对数据内存的管理。在数据传输过程中,每一个模块都是一个线程,每个模块的运行都是随机的,这就导致系统运行时信号处理顺序变得难以控制,需对

系统进行时序控制,使其输出正确的数据,通过对每个模块读取的数据长度判断程序运行顺序是否正确。雷达信号处理整体设计如图4所示。

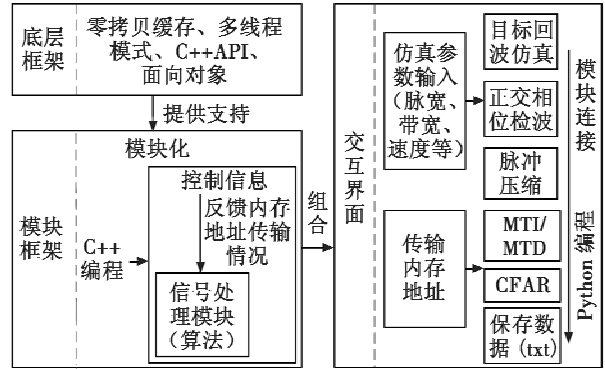


图4 雷达信号处理整体设计图

Fig. 4 Overall design of radar signal processing

通过多线程<sup>[12]</sup>模式、零拷贝缓存<sup>[13]</sup>模式的底层设计,在尽可能保证数据准确传输的同时,保证计算机的运行速度,并通过加入控制层,反馈内存地址传输情况,来保证时序的准确性。

### 2.2 人机交互界面

基于 GNU Radio 软件平台的人机交互界面被称为 GRC(GNU Radio Companion),如图5所示。

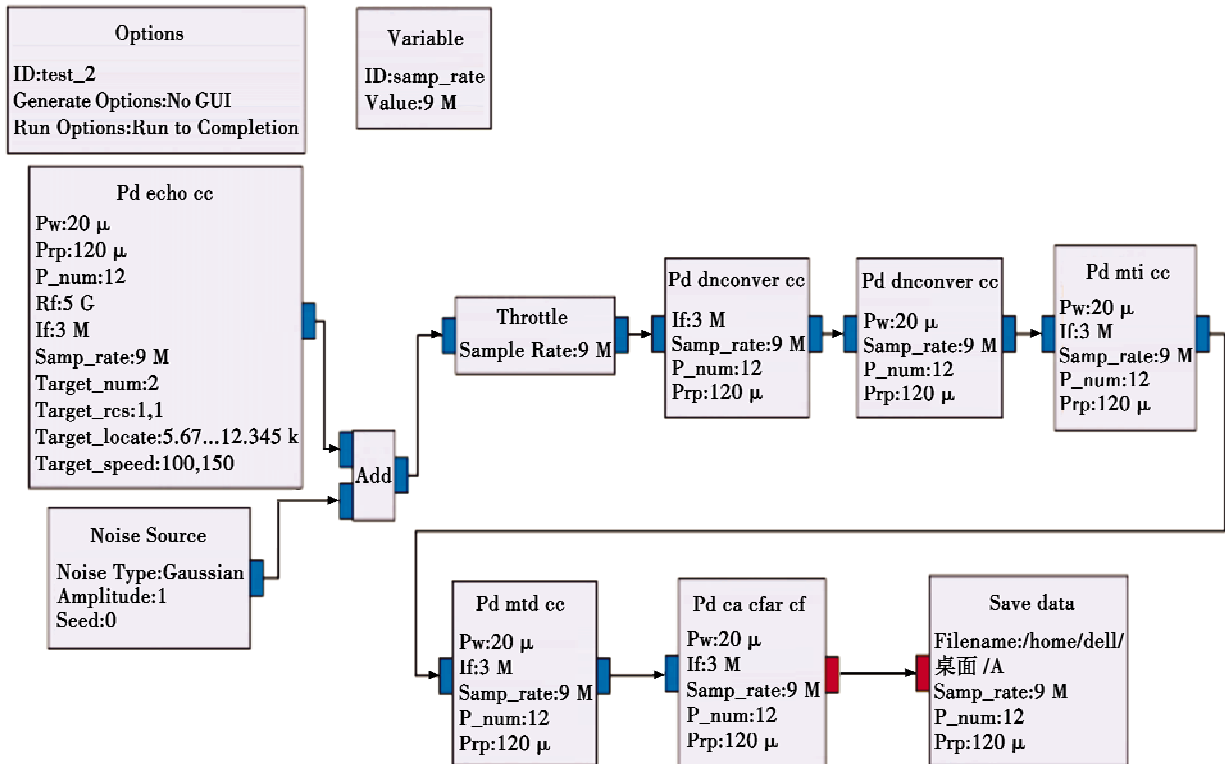


图5 雷达信号处理人机交互界面

Fig. 5 Human-computer interaction interface of radar signal processing

本文中,雷达系统以相参脉冲串波形的 PD 雷达为背景,用来对目标进行测速仿真。模块具体功能如

表1所示。

表1 模块及功能

Table 1 Modules and their functions

模块名	功能作用
Options	定义流程图名
Variable	定义采样频率
Pd echo cc	信源(生成目标)
Noise source	信源(生成噪声)
Add	数据相加
Throttle	节流阀
Pd dnconver cc	下变频
Pd compress cc	脉冲压缩
Pd mti cc	动目标显示
Pd mtd cc	动目标检测
Pd ca cfar cf	单元平均恒虚警
Save data	信宿(保存数据)

表1中的各模块内均需输入关于雷达信号处理的参数,如脉宽、带宽、采样频率及目标参数速度、位置等信息才能在连接后成功运行,其中,Pd echo cc与Pd dnconver cc模块与系统自带模块有连接,其内部接口已进行相应转换,这种模块化思想使系统具有简洁美观、操作简便的特点。

### 3 软件测试

#### 3.1 软件运行结果测试

雷达信号处理系统中,每个模块产生的数据均可自定义保存,为实现对完整的雷达信号处理系统进行测试,本文对图5中信号经恒虚警检测(CA-CFAR)处理后的数据进行保存,并使用Python对数据进行读取与可视化处理得到如图6所示的结果。

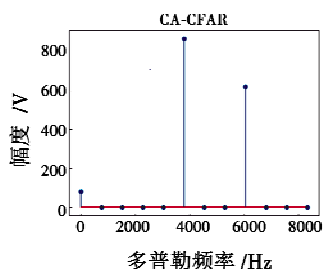


图6 数据可视化图

Fig. 6 Data visualization

由输入的参数与最后的输出结果比较,可得在系统存在一定虚警概率的情况下,所产生的数据是正确的,说明系统运行正常。

#### 3.2 软件运行速度测试

为了对单线程模式与多线程模式进行对比,使用C++语言编写了一个简单的雷达信号处理程序作为单线程模式系统,由于基于GNU Radio软件平台的雷达信号处理系统多线程模式下运行时具有较大的随机性,所以通过测量多组数据取平均值,单线程模式运行

稳定,取较少数据即可。系统在不同线程模式下运行时间如表2所示。

表2 不同线程模式下系统运行时间

Table 2 System running time in different thread modes

运行次数	运行时间/ms		运行次数	运行时间/ms	
	单线程	多线程		单线程	多线程
1	70 682	48 601	5	-	29 686
2	63 995	30 110	6	-	21 458
3	65 133	37 262	7	-	22 262
4	-	21 638	8	-	24 203

由图5中数据可知,系统要处理数据点数为12 960,在数据量较大的情况下,由表2中的数据可得,单线程模式下平均运行一次的时间为66 603.33 ms,而基于GNU Radio软件平台的多线程模式下,雷达信号处理系统平均运行一次的时间为29 408.13 ms,比单线程模式运行时间提高了55.85%,证明了在多线程模式下系统运行速度明显有所提升。

### 4 结语

基于GNU Radio软件平台的雷达信号处理系统界面布局简约美观、运行流畅稳定。相比于单线程模式系统,在对大数据量进行处理时,通过传输内存地址,在保证数据完整性的同时兼具提升计算机性能的优点,使计算机运行速度受影响较小,解决了目前雷达信号处理单线程仿真系统存在的主要缺陷。GNU Radio软件平台所具有的开源特性,使其在系统建模、信号处理方面均具有广阔的应用前景。

#### 参考文献

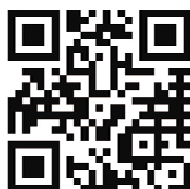
- [1] 严明. 脉冲多普勒雷达信号处理 MATLAB 仿真研究 [D]. 淮南:安徽理工大学,2016.
- [2] 熊邦毛,熊叶明,张再萍. 用多线程技术解决单线程冲突问题[J]. 计算机与数字工程,2007,35(1):115-118.
- [3] 曹瀚文,王文博. GNU Radio: 开放的软件无线电平台 [J]. 电信快报,2007(4):31-34.
- [4] BLOSSOM E. GNU Radio: tools for exploring the radio frequency spectrum [J]. Linux Journal, 2004, 2004(122): 76-81.
- [5] 王新国,郭锁利,李爱华,等. 多线程的光陀螺信号采集系统设计[J]. 电光与控制,2005,12(5):50-53.
- [6] 丁岚,范开勇,冯振颖. 面向对象程序设计方法研究 [J]. 数字通信世界,2019(1):73-74.
- [7] 刘晓荣. 软件模块化思想在变频器系统设计中的应用 [J]. 无线互联科技,2019,16(17):43-44.
- [8] MATTHES E. Python 编程:从入门到实践 [M]. 2版. 袁

(下转第102页)

架构,从而降低了系统功耗,提高了可靠性。目前,该系统设计已成功运用于某飞机显示控制管理系统。

### 参 考 文 献

- [1] 江娟娟. 基于 DSP 系统的机载液晶显示器图形反走样研究[D]. 西安:西安科技大学,2007.
- [2] 谷士鹏,马亚平,陈新华,等. 一种机载 XGA 视频信息采集的设计与实现[J]. 硅谷,2015(4):43,38.
- [3] 邢达波,艾波. 机载 RS422 总线采集技术应用研究[J]. 中国科技信息,2016(21):73-74.
- [4] 刘语乔,覃杨森,郭世伟. KAM500 机载采集 RS422 总线数据处理技术研究[J]. 测控技术,2017,36(3):47-49.
- [5] 张森,严小双,晏恺晨,等. 机载总线技术应用综述及其对飞机性能的影响[J]. 电子世界,2019(21):37-38.
- [6] 曾锋,冯宇. 基于 FPGA 的 ARINC429 协议设计[J]. 现代电子技术,2015,38(18):92-94.
- [7] 乔霖,李永红,岳凤英. 基于 ARINC818 机载航图单元视频辅助传输方案设计[J]. 电光与控制,2019,26(2):93-96.
- [8] 温世杰,刘康,柳邦奇,等. 基于 ARINC818 的航空视频发送接收系统设计与实现[J]. 电光与控制,2015,22(8):90-92,114.
- [9] 袁春辉,周建江,董珊珊. 基于 FPGA 的机载视频显示系统的低功耗设计[J]. 现代电子技术,2015,38(12):89-91.
- [10] 蒋中荣,甘俊杰,欧伟明. 基于 FPGA 的 LED 点阵显示控制方法[J]. 现代电子技术,2019,42(24):16-19,24.
- (上接第 45 页)
- [9] 聂建平. 基于支持向量机的离心泵故障诊断方法研究[D]. 哈尔滨:哈尔滨工业大学,2017.
- [10] 李琦,杜晓东,张洪略,等. 基于移动窗的多核最小二乘支持向量机建模算法[J]. 大连理工大学学报,2017,57(5):511-516.
- [11] 顾德英,吴成赛,侯娇. 基于补偿模糊神经网络的 BLD-CM 伺服控制[J]. 东北大学学报(自然科学版),2013,34(1):13-16.
- [12] XU Q F, ZHANG J X, JIANG C X, et al. Weighted quantile regression via support vector machine [J]. Expert Systems with Applications, 2015, 42(13):5441-5451.
- (上接第 78 页)
- 国忠,译. 北京:人民邮电出版社,2016.
- [9] LIPPMAN S B, LAJOIE J, MOO B E. C ++ Primer 中文版[M]. 5 版. 王刚,杨巨峰,译. 北京:电子工业出版社,2019.
- [10] KULSHRESHTHA K, NARAYANAN S H K, BESSAC J, et al. Efficient computation of derivatives for solving optimization problems in R and Python using SWIG-generated interfaces to ADOL-C[J]. Optimization Methods and Software, 2018, 33(4-6):1173-1191.
- [11] 白勇,胡祝华. GNU Radio 软件无线电技术[M]. 北京:科学出版社,2017.
- [12] 杨豪,胡利军,楼成武. 基于 LINQ 和多线程技术的雷达远程监控系统[J]. 计算机系统应用,2020,9(4):92-96.
- [13] HASHMI J M, CHU C H, CHAKRABORTY S, et al. FALCON-X: zero-copy MPI derived datatype processing on modern CPU and GPU architectures [J]. Journal of Parallel and Distributed Computing, 2020, 144:1-13.



请扫描二维码关注我刊