

基于FPGA的测距雷达信号处理线路设计与实现

赵瑞龙 于可欣 魏彤彤

陕西长岭电子科技有限责任公司 陕西 宝鸡 721006

摘 要: 本文提出了一种基于现场可编程门阵列 (FPGA) 的测距雷达信号处理线路设计与实现方案。该方案利用FPGA的高速并行处理能力, 实现了对雷达回波信号的高效处理, 提高了测距雷达的精度和实时性。文章详细阐述了系统的硬件设计、软件实现以及关键信号处理算法, 为基于FPGA的测距雷达信号处理提供了一种可行的解决方案。

关键词: FPGA; 测距雷达; 信号处理; 线路设计; 实现

引言

测距雷达作为现代探测技术的重要组成部分, 广泛应用于军事、民用等多个领域。随着科技的发展, 对测距雷达的精度和实时性要求越来越高。传统基于数字信号处理器 (DSP) 的信号处理方案, 在处理速度和灵活性上存在一定的局限性。而FPGA以其高速并行处理能力和高度可配置性, 成为实现高精度、实时测距雷达信号处理的理想选择。

1 系统总体设计

1.1 系统架构

基于FPGA的测距雷达信号处理系统, 主要由以下几个核心部分组成, 共同协作以实现精准的测距功能。雷达前端作为系统的“眼睛”, 承担着发射与接收雷达信号的重任。它利用高频电磁波的特性, 向目标物体发射信号, 并接收由物体反射回来的回波信号。这一环节是雷达测距的基础, 其性能直接影响到后续信号处理的准确性和可靠性。紧接着, ADC (模数转换器) 发挥着桥梁的作用。它将雷达前端接收到的模拟回波信号转换为数字信号, 以便后续的数字信号处理单元能够进行处理。这一转换过程需要高度的精确性和稳定性, 以确保信号在转换过程中不失真。FPGA处理单元是整个系统的“大脑”。它接收来自ADC的数字信号, 并通过一系列复杂的算法和逻辑运算, 对信号进行实时处理。这些处理包括信号滤波、目标检测、距离计算等, 旨在从原始信号中提取出有用的测距信息。处理完毕后, DAC (数模转换器) 将FPGA输出的数字信号转换回模拟信号, 以便进行后续的输出或显示。这一步骤是信号处理的最终环节, 也是将处理结果呈现给用户的关键。最后, 上位机接口作为系统与外部世界的“窗口”, 负责与上位机进行通信和数据传输。通过这一接口, 用户可以方便地获取测距结果, 并对系统进行配置和监控。

1.2 设计目标

基于FPGA的测距雷达信号处理系统的设计目标, 旨在实现高精度与实时性的完美结合。致力于通过精细的算法设计与FPGA的高效并行处理能力, 确保雷达信号处理的每一个环节都能达到极高的精度, 从而准确测量目标物体的距离。同时, 利用FPGA的并行处理特性, 将大幅提升信号处理的速度, 实现实时测距, 满足各种动态应用场景的需求。此外, 系统的设计还充分考虑了灵活性与可配置性^[1]。在硬件层面, 采用模块化的设计理念, 使得各个组件之间既相互独立又易于扩展。在软件层面, 设计可配置的参数与算法接口, 使得用户可以根据实际需求对系统进行定制化的调整与优化。这样的设计不仅便于后续的升级与维护, 也大大增强了系统的适应性与生命力。

2 硬件设计

2.1 FPGA选型

在选型时, 需综合考虑多个因素, 包括逻辑资源量、I/O接口速度、DSP处理能力以及功耗等。对于测距雷达信号处理系统而言, Xilinx的Virtex系列和Intel的Stratix系列都是极佳的选择。这两大系列的FPGA芯片均以其卓越的性能和丰富的资源著称。以Xilinx的Virtex-7系列为例, 它提供了高达200万个逻辑单元、高达960个DSP切片以及高速的GTX/GTH串行收发器, 能够轻松应对复杂的信号处理任务。而Intel的Stratix 10系列则以其先进的14nm工艺、高达10Tb/s的串行带宽以及强大的硬件加速功能, 为高性能信号处理提供了有力支持。在选择具体型号时, 需根据系统的实际需求进行权衡。若系统对处理速度和实时性要求较高, 可选择具有更高逻辑资源和DSP切片的型号; 若对功耗有严格限制, 则可考虑采用低功耗版本的FPGA。此外, 还需考虑FPGA的封装形式、引脚数量以及价格等因素, 以确保选型的合理性与经济性。

2.2 ADC与DAC设计

对于ADC而言,选择具有高采样率和高分辨率的芯片至关重要。高采样率能够确保捕捉到更多的信号细节,提高测量的精度;而高分辨率则能够提供更精细的量化级别,减少量化误差。例如,AD9625是一款具有12位分辨率、最高采样率可达250MSPS的ADC芯片,适用于对带宽和精度要求较高的雷达信号处理系统。而AD9680则是一款14位、1GSPS的高性能ADC,能够提供更高的动态范围和信噪比,适用于对信号质量要求极高的场合。在DAC的选择方面,同样需要关注其分辨率、采样率以及噪声性能等指标。低噪声和高精度的DAC能够确保输出信号的纯净度和稳定性。例如,AD9739是一款14位、1.2GSPS的DAC芯片,具有极低的噪声和出色的线性度,适用于高精度信号生成和重建。而AD9783则是一款16位、250MSPS的DAC,其高分辨率和优异的噪声性能使其成为高精度模拟信号输出的理想选择。除了芯片本身的性能外,还需考虑ADC和DAC与FPGA之间的接口设计。为了确保高速、稳定的数据传输,可采用高速串行接口(如LVDS或JESD204B)进行连接。LVDS(Low-Voltage Differential Signaling)接口具有低功耗、高速率和抗干扰能力强的特点,适用于短距离、高速数据传输。而JESD204B则是一种用于高速ADC和DAC与FPGA之间数据传输的串行接口标准,它支持更高的数据速率和更长的传输距离,同时降低了接口设计的复杂性和成本^[2]。在实际设计中,还需根据系统的具体需求选择合适的接口类型和芯片封装形式。例如,对于需要紧凑设计的应用场景,可选择BGA(Ball Grid Array)封装的芯片;而对于需要方便调试和测试的场景,则可选择QFP(Quad Flat Package)等易于操作的封装形式。

2.3 接口设计

在FPGA与ADC、DAC之间的接口设计中,需考虑数据格式、时钟同步、数据传输速率以及接口电路的布局和布线等因素。为了确保数据的准确传输和时钟的同步性,可采用专用的时钟管理芯片或FPGA内部的时钟管理模块进行时钟分配和同步。同时,还需根据ADC和DAC的数据手册选择合适的接口电路和电平转换芯片,以确保信号的电平和格式与FPGA相匹配。在FPGA与上位机之间的接口设计中,需考虑通信协议、数据传输速率、接口类型以及接口电路的保护和隔离等措施。常用的通信协议包括UART、SPI、I2C以及以太网等,具体选择需根据系统的实际需求进行权衡。例如,对于需要高速数据传输和远距离通信的场景,可选择以太网接口;而对于需要简单、可靠的通信方式的场景,则可选择UART或

SPI等接口。为了提高数据传输的速率和可靠性,可采用高速串行接口(如LVDS或USB3.0)进行连接。LVDS接口具有低功耗、高速率和抗干扰能力强的特点,适用于短距离、高速数据传输。而USB3.0则是一种广泛应用的高速串行接口标准,它支持更高的数据速率和更长的传输距离,同时提供了丰富的外设连接选项和电源管理功能。在接口电路的设计中,还需考虑信号完整性、电磁干扰以及电源管理等问题。例如,可采用合适的阻抗匹配网络、滤波器和去耦电容等元件来提高信号的完整性和稳定性;采用屏蔽和接地等措施来减少电磁干扰;采用电源管理芯片和稳压器等元件来确保接口电路的电源稳定性和可靠性。

3 软件实现

3.1 信号处理算法

3.1.1 数字混频

数字混频是雷达信号处理中的第一步,也是将模拟信号转换为数字信号后进行的关键处理之一。其原理是将接收到的雷达信号与本地振荡器(LO)产生的信号进行混频,从而得到下变频后的基带信号。这一过程在FPGA中通过数字混频器实现,核心组件是乘法器和累加器。在FPGA设计中,数字混频器通常采用定点运算,以提高处理速度和降低资源消耗。乘法器用于实现雷达信号与LO信号的相乘操作,而累加器则用于对乘法结果进行累加,得到混频后的信号。为了确保混频的准确性和稳定性,需要精确控制LO信号的频率和相位,以及乘法器和累加器的位宽和舍入方式^[3]。此外,数字混频器还需要考虑抗混叠滤波的问题。由于混频过程中会产生高频分量,如果不进行适当的滤波,这些高频分量会对后续处理造成干扰。因此,在数字混频器之前,通常需要加入一个抗混叠滤波器,以滤除不必要的高频成分。

3.1.2 低通滤波

低通滤波器在雷达信号处理中扮演着至关重要的角色,它用于滤除混频后信号中的高频成分,保留低频有用信号。在FPGA设计中,低通滤波器通常采用有限脉冲响应(FIR)或无限脉冲响应(IIR)结构。FIR滤波器具有线性相位和稳定性的特点,适用于对相位要求较高的场合。在FPGA实现中,FIR滤波器通过系数加载和卷积运算实现滤波处理。具体地,将滤波器的系数存储在FPGA的存储器中,然后通过卷积运算将输入信号与滤波器系数进行相乘和累加,得到滤波后的信号。IIR滤波器则具有更高的滤波效率和更低的资源消耗,适用于对滤波效率要求较高的场合。在FPGA实现中,IIR滤波器通常通过递归结构实现,即利用前一时刻的输出信号和当前

时刻的输入信号进行计算,得到当前时刻的输出信号。这种结构使得IIR滤波器能够实现更高的滤波阶数,从而获得更好的滤波效果。在实际设计中,需要根据系统的具体需求和FPGA的资源情况选择合适的滤波器结构和参数。同时,还需要考虑滤波器的量化效应和舍入误差对滤波效果的影响,以确保滤波器的准确性和稳定性。

3.1.3 快速傅里叶变换(FFT)

FFT是雷达信号处理中的一项重要技术,它用于对滤波后的信号进行频谱分析,提取目标信号的频率信息。在FPGA中实现FFT算法,可以大大提高运算速度和实时性。FFT算法的核心思想是将时域信号转换为频域信号,从而得到信号的频谱信息。在FPGA实现中,通常采用流水线结构和并行处理来提高运算速度。具体地,将输入信号分为多个数据块,每个数据块分别进行FFT运算,然后将结果合并得到最终的频谱信息。为了实现高效的FFT运算,需要合理划分数据块的大小和数量,以及优化FFT运算的蝶形结构和旋转因子。此外,还需要考虑FFT运算的量化效应和舍入误差对结果的影响,以确保频谱分析的准确性。

3.1.4 目标检测与测距

目标检测与测距是雷达信号处理系统的最终目的。根据FFT结果,可以通过峰值检测算法提取目标信号的频率信息,并根据雷达方程计算目标距离。在FPGA实现中,目标检测与测距算法通常包括阈值判断和距离计算两个步骤。阈值判断用于确定是否存在目标信号,即判断FFT结果中的峰值是否超过设定的阈值。如果超过阈值,则认为存在目标信号,并进行距离计算;否则,认为没有目标信号。距离计算则是根据雷达方程和FFT结果中的峰值频率计算目标距离。具体地,将峰值频率代入雷达方程中,解算出目标距离。为了确保距离计算的准确性,需要精确测量峰值频率,并考虑雷达方程中的各种参数和误差因素。

3.2 FPGA编程

在实现了上述信号处理算法后,接下来需要通过FPGA编程将这些算法集成到系统中。FPGA编程通常使用硬件描述语言(HDL),如Verilog或VHDL。在编程过程中,需要采用模块化设计的方法,将各个处理模块封装成独立的IP核。这样做的好处是可以提高代码的可读性和可维护性,便于后续调试和升级。同时,还可以将各个模块进行复用和组合,以构建更复杂的处理系统。在具体编程时,需要关注几个关键点。首先是资源利用率的优化,即如何在有限的FPGA资源上实现高效的信号处理算法。这需要通过合理的代码结构和算法优化来实现^[4]。其次是时序约束和时钟管理,以确保各个模块之间的时序关系和时钟同步性。最后是功耗管理,即通过合理的电源规划和功耗优化来降低系统的功耗。

结语

本文提出了一种基于FPGA的测距雷达信号处理线路设计与实现方案,通过硬件和软件的综合设计,实现了高精度、实时的测距雷达信号处理。该方案充分利用了FPGA的高速并行处理能力和高度可配置性,为测距雷达信号处理提供了一种新的解决方案。未来工作将进一步优化算法设计,提高系统性能和稳定性。

参考文献

- [1]李欢,余红英.基于FPGA的高性能激光雷达测距系统设计[J].电子测量技术,2021,44(20):6-10.
- [2]曾洁.基于FPGA的多通道近距离测距雷达设计[D].成都理工大学,2021.
- [3]陈林军,涂亚庆,刘鹏,等.基于FPGA的LFMCW测距雷达调制信号源设计[J].电子设计工程,2016,24(01):115-117+120.
- [4]梁颖,李克丽,田粉仙,等.基于FPGA的LFMCW雷达测速测距优化系统研究[J].云南大学学报(自然科学版),2020,42(01):36-42.