

中图分类号:

单位代号: 10280

密 级:

学 号:

上海大学



专业学位硕士学位论文

题 目	面向 LTE-V2X 基带芯片的 关键技术研究 与 FPGA 实现
--------	--------------------------------------

作 者 XXX

学科专业 XXX

导 师 XXX

完成日期 2023 年 5 月

姓 名：XXX

学号：XXX

论文题目：面向 LTE-V2X 基带芯片的关键技术与 FPGA 实现

上海大学

本论文经答辩委员会全体委员审查，确认符合上海大学硕士学位论文质量要求。

答辩委员会签名：

主任：

委员：

导 师：

答辩日期：

姓 名：XXX

学号：XXX

论文题目：面向 LTE-V2X 基带芯片的关键技术与 FPGA 实现

原创性声明

本人声明：所呈交的论文是本人在导师指导下进行的研究工作。除了文中特别加以标注和致谢的地方外，论文中不包含其他人已发表或撰写过的研究成果。参与同一工作的其他同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

签 名：_____日 期：_____

本论文使用授权说明

本人完全了解上海大学有关保留、使用学位论文的规定，即：学校有权保留论文及送交论文复印件，允许论文被查阅和借阅；学校可以公布论文的全部或部分内容。

（保密的论文在解密后应遵守此规定）

签 名：_____导师签名：_____日期：_____

上海大学工程硕士学位论文

面向 **LTE-V2X** 基带芯片的
关键技术与 **FPGA** 实现

姓 名：XXX

导 师：XX

学科专业：电子信息工程

上海大学通信与信息工程学院

2023 年 05 月

A Dissertation Submitted to Shanghai University for the Degree
of Master in Engineering

FPGA-Based Realization of Pivotal Technologies for LTE-V2X Baseband Chip

MA Candidate: XXX

Supervisor: XX

Major: XXX

**School of Communication and Information Engineering,
Shanghai University**

May, 2023

摘 要

随着道路车辆日益增多,交通拥堵和交通事故也逐渐增多,给交通系统的运行和安全带来了巨大的挑战。为应对这些挑战,车联网技术得到了广泛的重视与发展。作为车联网通信技术的一种解决方案,车联网长期演进(Long-Term Evolution Vehicle-to-Everything, LTE-V2X)技术凭借其能满足低延迟、高速和高可靠性需求的优势,被越来越多的国家采用。借助 LTE-V2X 技术,车辆与车辆、车辆与道路设施的互连通信得以实现,从而有效缓解交通拥堵情况、降低事故发生频率,从而提高交通的安全性。本文针对面向 LTE-V2X 基带芯片的关键技术和现场可编程门阵列(Field-Programmable Gate Array, FPGA)实现方案进行研究,主要研究内容和创新点如下:

首先,本文针对车联网通信过程中发射机与接收机之间时间频率不同步的问题,提出了一种硬件友好的接收机同步算法并进行了硬件实现。该算法基于 LTE-V2X 物理层协议,采用改进的互相关算法,同时选取两段信号进行计算,有效提高了接收机的抗频偏性能。同时,该方法结合了 LTE-V2X 无线帧的特点,使用局部遍历的方式筛选互相关结果的峰值,相比于传统的全部遍历方式在硬件实现了减少 44%的块存储器资源消耗。基于 FPGA 对该方法进行硬件实现,并在课题组自研的开发板上进行测试。FPGA 实现的结果表明,本文设计的同步模块支持 307.2MHz 的工作频率,消耗的查找表资源为 9628,块存储器资源为 14;同步的建立时间少于 1 秒,维持的时间大于 24 小时;完成同步后,收端的解包成功率为 99.74%。

其次,为提高通信系统在不同信道配置下的传输效率和性能,本文根据基带中不同信道的特点分别实现了两种不同的解速率匹配硬件设计方案,设计了并行处理和流水线的结构,因此,相比于 LTE-V2X 工程中的早期方案,本文实现的方案在处理速率上提升了一倍,并且支持 368 种不同的信道配置。接着,为满足发射机和接收机中解调参考信号估计和相位补偿的需求,本文分别设计了正余弦计算模式和相位旋转模式的坐标旋转数字计算机(Coordinate Rotation Digital Computer, CORDIC)模块,并针对 LTE-V2X 的具体配置参数进行定制化设计,资源消耗降低了 52.1%。然后,为实现调制和解调过程中信号时域和频域之间的

转换，本文设计了流水结构的快速傅里叶变换（Fast Fourier Transform, FFT）和快速傅里叶逆变换（Inverse Fast Fourier Transform, IFFT）模块。经过仿真和上板测试，本文设计的 FFT 和 IFFT 模块支持 2048 点输入序列的流水计算，可综合的最高频率为 400MHz，计算延时为 20.57 微秒。

最后，本文将上述模块接入 LTE-V2X 发射机和接收机中，完成了系统集成工作。此时，LTE-V2X 收发机的工作频率为 307.2MHz，最大传输速率为 31.65Mbps，可支持 368 种不同的信道配置，解包成功率为 99.74%。同时，该收发机可以与商用大唐 VU3004 C-V2X 车载终端芯片实现互连，并支持 720P 分辨率的无线视频传输。

关键词：LTE-V2X, FPGA, 同步, 解速率匹配, CORDIC, FFT

ABSTRACT

With the increasing number of road vehicles, traffic congestion and accidents are also gradually increasing, posing significant challenges to the operation and safety of the transportation system. In order to address these challenges, the Internet of Vehicles technology has received widespread attention and development. As a solution in the field of Internet of Vehicles communication technology, Long-Term Evolution Vehicle-to-Everything (LTE-V2X) technology has been adopted by an increasing number of countries due to its advantages in meeting low-latency, high-speed, and high-reliability requirements. With the help of LTE-V2X technology, interconnectivity and communication between vehicles and road infrastructure can be achieved, effectively alleviating traffic congestion and reducing accident frequencies, thereby improving traffic safety. This thesis focuses on the key technologies for LTE-V2X baseband chip and explores the implementation scheme using Field-Programmable Gate Array (FPGA). The main research content and innovations are as follows:

Firstly, this thesis proposes a hardware friendly receiver synchronization algorithm and implements it to address the issue of time and frequency asynchrony between transmitters and receivers in the process of vehicle networking communication. This algorithm is based on the LTE-V2X physical layer protocol, adopts an improved cross correlation algorithm, and selects two segments of signals for calculation, effectively improving the frequency offset resistance performance of the receiver. At the same time, this method combines the characteristics of LTE-V2X wireless frames and uses local traversal to filter the peaks of cross-correlation results. Compared to the traditional full traversal method, it reduces block memory resource consumption by 44% in hardware implementation. Hardware implementation of this method based on FPGA and testing on a self-developed development board by the research group. The results of FPGA implementation indicate that the synchronization module designed in this thesis supports a working frequency of 307.2MHz, consumes 9628 lookup table resources, and 14 block memory resources; The establishment time

of synchronization is less than 1 second, and the maintenance time is greater than 24 hours; After completing the synchronization, the success rate of unpacking at the receiving end is 99.74%.

Secondly, in order to improve the transmission efficiency and performance of communication systems in different channel configurations, this thesis implements two different de-rate matching hardware design schemes based on the characteristics of different channels in the baseband, and designs parallel processing and pipeline structures. Therefore, compared to early schemes in LTE-V2X engineering, the scheme implemented in this thesis has doubled the processing rate and supports 368 different channel configurations. Then, in order to meet the requirements of demodulation reference signal estimation and phase compensation in the transmitter and receiver, this thesis designs the Coordinate Rotation Digital Computer (CORDIC) module in the sine cosine calculation mode and the phase rotation mode respectively, and customized the specific configuration parameters of LTE-V2X, reducing the resource consumption by 52.1%. Then, in order to achieve the conversion between the time-domain and frequency-domain of the signal in the modulation and demodulation process, this thesis designs pipeline structured Fast Fourier Transform (FFT) and Inverse Fast Fourier Transform (IFFT) modules. After simulation and board testing, the FFT and IFFT modules designed in this thesis support pipeline calculation of 2048 input sequences, with a maximum frequency of 400MHz that can be synthesized and a calculation delay of 20.57 microseconds.

Finally, this thesis integrates the aforementioned modules into the LTE-V2X transmitter and receiver, completing the system integration work. At this stage, the LTE-V2X transceiver operates at a frequency of 307.2MHz with a maximum transmission rate of 31.65Mbps. It supports 368 different channel configurations and achieves a successful packet decoding rate of 99.74%. Additionally, the transceiver can establish interconnection with the commercial VU3004 C-V2X chip from Datang and supports wireless video transmission with a resolution of 720p.

Keywords: LTE-V2X, FPGA, Synchronization, De-rate matching, CORDIC, FFT

目 录

摘 要	V
ABSTRACT.....	VII
目 录	IX
专业术语注释表	XII
第一章 绪论	1
1.1 课题研究背景和意义.....	1
1.2 研究现状.....	2
1.2.1 V2X 通信技术的现状.....	2
1.2.2 基带芯片的研究现状.....	3
1.3 现有研究的不足.....	5
1.4 研究内容和论文结构.....	6
1.4.1 主要的研究内容.....	6
1.4.2 论文结构安排.....	7
第二章 LTE-V2X 系统概述	9
2.1 LTE-V2X 物理层协议.....	9
2.1.1 LTE-V2X 物理层无线帧结构.....	9
2.1.2 LTE-V2X 的系统带宽和时隙架构.....	10
2.2 LTE-V2X 系统框架.....	11
2.3 发射机系统结构.....	12
2.3.1 发端比特级处理模块.....	12
2.3.2 发端符号级处理模块.....	13
2.4 接收机系统结构.....	13
2.4.1 收端符号级处理模块.....	13
2.4.2 收端比特级处理模块.....	15
2.5 开发环境简介.....	15
2.5.1 开发流程.....	15
2.5.2 实际测试的硬件平台	17
2.6 本章小结.....	18

第三章 LTE-V2X 收端同步算法的设计与实现	19
3.1 引言.....	19
3.1.1 同步技术的背景.....	19
3.1.2 传统的同步算法.....	20
3.2 基于 LTE-V2X 的同步算法.....	21
3.2.1 同步信号.....	21
3.2.2 提出的同步算法.....	22
3.3 同步算法的建模仿真.....	24
3.4 FPGA 硬件实现方案	26
3.4.1 总体硬件设计方案.....	26
3.4.2 同步相关模块的设计	26
3.4.3 频偏估计和补偿模块的设计	29
3.4.4 定时标记模块的设计	30
3.4.5 硬件资源消耗分析.....	32
3.5 本章小结.....	32
第四章 解速率匹配技术的实现	33
4.1 引言.....	33
4.2 解速率匹配算法的研究与建模.....	34
4.2.1 解速率匹配整体实现结构.....	34
4.2.2 虚拟循环缓冲器的实现.....	34
4.2.3 解速率匹配其他步骤的实现.....	36
4.3 控制信道解速率匹配模块的实现.....	37
4.3.1 总体设计框图.....	37
4.3.2 控制信道解速率匹配各个模块的设计	38
4.3.3 测试结果.....	41
4.4 共享信道解速率匹配模块的实现.....	42
4.4.1 总体设计框图.....	42
4.4.2 共享信道解速率匹配各个模块的设计	43
4.4.3 测试结果.....	46

4.5 本章小结.....	48
第五章 CORDIC 和 FFT 计算模块的设计	50
5.1 引言.....	50
5.2 CORDIC 计算模块的设计.....	51
5.2.1 CORDIC 算法原理.....	51
5.2.2 设计需求.....	53
5.2.3 正余弦计算模式 CORDIC 模块的实现.....	53
5.2.4 相位旋转模式 CORDIC 模块的实现.....	55
5.2.5 测试结果.....	56
5.3 FFT 与 IFFT 计算模块的设计.....	59
5.3.1 算法原理.....	59
5.3.2 设计需求.....	60
5.3.3 硬件设计.....	61
5.3.4 测试结果.....	63
5.4 本章小结.....	65
第六章 LTE-V2X 系统集成与测试	66
6.1 LTE-V2X 系统集成.....	66
6.2 整体时序和资源分析.....	66
6.3 固定长度数据包传输测试.....	68
6.4 无线视频传输测试.....	70
6.5 本章小结.....	73
第七章 结论与展望	74
7.1 论文工作总结.....	74
7.2 未来工作展望.....	75
参考文献	76
作者在攻读硕士学位期间公开发表的论文	83
作者在攻读硕士学位期间参与的项目	84
致 谢	85

专业术语注释表

缩略词说明:

ADC	Analog-to-Digital Converter	模数转换器
ARM	Advanced RISC Machines	先进精简指令集计算机
BRAM	Block Random Access Memory	块存储器
CORDIC	Coordinate Rotation Digital Computer	坐标旋转数字计算机
CRC	Cyclic Redundancy Check	循环冗余校验
DFT	Discrete Fourier Transform	离散傅里叶变换
DMRS	Downlink Mapping Reference Signal	下行链路解调参考信号
DSP	Digital Signal Processor	数字信号处理器
DSRC	Dedicated Short Range Communications	专用短距离通信
FDD	Frequency Division Duplex	频分双工
FF	Flip-Flop	触发器
FFT	Fast Fourier Transform	快速傅里叶变换
FIFO	First In First Out	先进先出
FPGA	Field-Programmable Gate Array	可编程门阵列
IDFT	Inverse Discrete Fourier Transform	离散傅里叶逆变换
IFFT	Inverse Fast Fourier Transform	快速傅里叶逆变换
IO	Input/Output	输入/输出
IP	Intellectual Property	知识产权
LTE-V2X	Long-Term Evolution for Vehicular networks	车联网长期演进技术
LUT	Look-Up Table	查找表
MMCM	Mixed-Mode Clock Manager	多时钟管理器
OFDM	Orthogonal Frequency Division Multiplexing	正交频分复用
OFDMA	Orthogonal Frequency Division Multiple Access	正交频分多址
PLL	Phase-Locked Loop	锁相环

PSS	Primary Synchronization Signal	主同步信号
RTL	Register Transfer Level	寄存器传输级
SC-FDMA	Single Carrier Frequency Division Multiple Access	单载波频分多址
SCI	Sidelink Control Information	侧链控制信息
SMA	SubMiniature version A	微型同轴射频接口 A 型
SSS	Secondary synchronization signal	辅同步信号
TDD	Time Division Duplex	时分双工
TNS	Total Negative Slack	总负总裕量
V2X	Vehicle To Everything	车对外界的信息交换
WHS	Worst Hold Slack	最差保持时间余量
WNS	Worst Negative Slack	最差负总裕量

第一章 绪论

1.1 课题研究背景和意义

随着经济的发展和人们生活品质的提升，道路上的车辆数量在稳步增长，这也导致了交通拥堵和交通事故的增加^[1]。为应对这些问题，人们对智能交通系统的需求日益增长。车联网作为智能交通领域的关键技术之一^[2-4]，可以为智能交通提供丰富的数据支持。传统的无线局域网通信系统和通用移动通信系统无法满足车联网高速场景下的低延迟和高可靠性要求^[5]，因此，专门为车联网场景优化的车对外界的信息交换（Vehicle To Everything, V2X）技术应运而生，并得到了广泛关注和快速发展^[6-8]。

如图 1.1 所示，V2X 技术可以实现车辆和其他各种对象的通信，包括车辆与车辆、车辆与行人、车辆与网络、车辆与云服务器和道路基础设施的通信。这种实时信息交流能使车辆更准确地感知周围环境，从而提高交通的安全性^[9-13]。

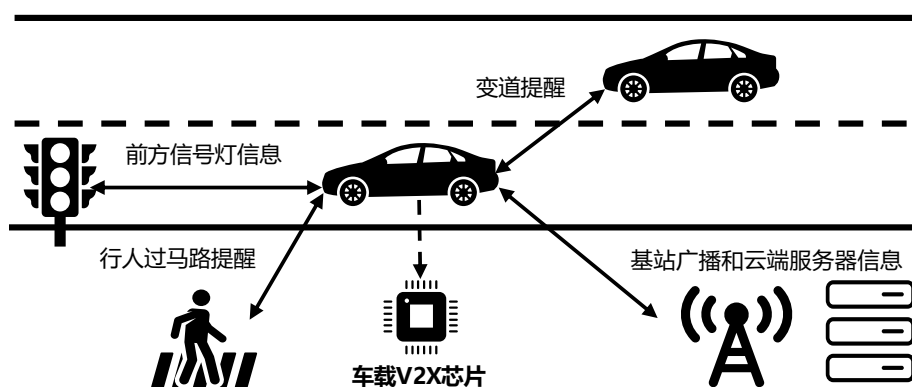


图 1.1 V2X 应用场景示意图

实现 V2X 技术的核心组件是基带芯片。基带芯片将多个功能集成在一个芯片上，从而减少电路板面积和系统成本，同时提高系统的可靠性和稳定性。目前各家厂商都在推出基于蜂窝网络尤其是 5G 技术的车联网产品，其中高通、华为、大唐等公司推出了各具特色的 V2X 芯片组和参考设计，为车辆提供精准的定位和通信能力，并支持直接通信和基于网络的通信。然而，目前已有的 V2X 基带芯片支持的通信配置较少，定制化能力和灵活性还有待提高。此外，V2X 部分新

技术还停留在算法仿真阶段，还未针对硬件实现的特点进行优化，也没有在硬件平台上进行大规模的实际测试和验证。

针对以上背景，本文选择 V2X 技术作为车联网的无线通信方案，并根据第三代合作伙伴计划社区（3rd Generation Partnership Project, 3GPP）发布的第 14 代协议标准^[14]设计物理层基带芯片。同时，本文根据不同信道配置需求设计了 LTE-V2X 系统中的部分关键模块，并基于现场可编程门阵列（Field-Programmable Gate Array, FPGA）对其中的算法进行了硬件实现。

1.2 研究现状

1.2.1 V2X 通信技术的研究现状

V2X 技术主要包括专用短距离通信（Dedicated Short Range Communications, DSRC）技术和蜂窝车联网（Cellular-V2X, C-V2X）技术，其中 C-V2X 技术又包括长期演进车辆对一切的通信（Long-Term Evolution Vehicle-to-Everything, LTE-V2X）技术和基于第五代移动通信技术的车辆对一切的通信（5G Vehicle-to-Everything, 5G-V2X）技术，5G-V2X 是 LTE-V2X 的未来演进方向。国内外各大公司已经推出了支持 V2X 的产品，例如高通公司可支持 C-V2X 的 9150 芯片、大唐公司可支持 LTE-V2X 的 DMD31 通信模组、Autotalks 公司可支持 DSRC 和 C-V2X 双模式的芯片组。DSRC 技术是为车辆定制的一种通信技术，支持单向和双向、短距离或者中距离的通信^[15-17]。LTE-V2X 技术作为一项新型的车联网技术，目前主要由国内企业进行主导，比如大唐电信针对车联网场景进行特定的优化，并获得自主知识产权，技术相对成熟。和 DSRC 技术相比，LTE-V2X 技术在高速场景下可靠性更好、通信范围更远^[18-21]，适用于更广阔的地域范围，比如城市、乡村和高速公路等。并且，LTE-V2X 技术允许车辆与车辆之间直接进行通信，无需道路基础设施的支持，这将推动无线连接和自动驾驶技术的发展^[22-25]。

由于道路上车辆的高速移动性和车辆拥堵时的密集性，车联网的通信场景相比于传统的无线通信场景更加复杂^[26]。因此，C-V2X 技术需要选择合适的无线传输技术。当无线通信系统中的数据传输速率很高时，信道的频率选择性会产生严重的符号间干扰，导致信号传输性能下降^[27]。使用正交频分复用（Orthogonal

Frequency Division Multiplexing, OFDM) 调制技术可以有效减小符号间干扰^[28], 提高频谱效率。单载波频分多址 (Single Carrier Frequency Division Multiple Access, SC-FDMA) 和正交频分多址 (Orthogonal Frequency Division Multiple Access, OFDMA) 都是基于 OFDM 的多址接入技术^[29,30], 该技术可以将有限的通信资源划分为多个子载波, 从而满足多个用户同时通信的需求。相比于 OFMDA, SC-FDMA 具有较低的峰均功率比和更强的时延敏感性^[31, 32]。较低的峰均功率比可以减少功率放大器的非线性失真, 从而降低设备的能耗, 在车联网通信场景下可以延长电池寿命并降低设备成本^[33], 而更强的时延敏感性可以提高安全性^[34]。基于上述原因, C-V2X 技术选择 SC-FDMA 作为无线传输的解决方案。

1.2.2 基带芯片的研究现状

在基带芯片的设计中, 降低算法复杂度、减少延时和功耗、提高数据的吞吐率是研究人员关注的热点。针对这些问题, 研究人员提出了各种解决方案, 包括使用 FPGA 作为硬件加速器、优化算法、采用并行计算等方式。文献^[35]设计了适用于长期演进技术 (Long-Term Evolution, LTE) 的低功耗和低延迟基带处理器, 其中采用了一种高效的并行数据处理方法提高了计算的性能。文献^[36]提出一种 LTE 物理下行共享信道的 FPGA 加速架构设计和实现, 通过结合通用处理器和硬件加速器实现了传输信号的高速实时处理。

此外, 基于 LTE-V2X 技术的基带芯片需要实现传统 LTE 技术中的多种无线通信算法, 包括同步算法、速率匹配算法、快速傅里叶变换算法等。对于这些关键算法的优化和硬件实现, 许多学者进行了相关的研究, 具体内容如下:

1) 同步算法的实现

同步算法在无线通信系统中的应用是实现时间和频率的同步。文献^[37]提出了一种 LTE 系统在多径衰落信道中确定时间偏移和分数载波频率偏移的同步方法。Nassralla 等使用主同步信号, 提出了一种基于分布式算法的无乘法同步算法^[38]。由于同步算法中有大量重复的计算, 因此部分研究人员选择基于 FPGA 对同步算法进行实现。Kim 等提出了一种基于 FPGA 的同步方法, 使用匹配滤波器检测同步信号^[39]。Chung 等提出了一种基于长期演进的全双工正交频分复用系统

的鲁棒定时同步方法并进行了 FPGA 实现^[40]。在 LTE-V2X 系统中，由于车联网环境的复杂性，同步技术的抗噪声和抗频偏性能十分关键。同时，由于最终要在 FPGA 开发板上进行实现，需要考虑硬件实现的资源消耗和功耗，为此要在精度和资源方面做出平衡。

2) 解速率匹配算法的实现

在无线通信系统中，解速率匹配算法通过对信道承载能力的适配，可以根据当前信道条件动态调整数据的传输速率，进而提高通信的效率和可靠性^[41,42]。对于该算法的实现，降低算法的复杂度并提高吞吐量和能量效率是研究的热点。例如，Guanghui 等^[43]中提出了一种基于通用处理器技术的解速率匹配方法，该方法使用并行处理的结构获得了更高的吞吐量和更低的延时。Bukris 等提出的硬件实现结构利用分组和分割子块的方法降低复杂度^[44]。此外还有优化流水结构^[45]、优化子块交织结构^[46]等方法。在本文研究的 LTE-V2X 系统中，需要支持多种不同的信道配置，因此对解速率匹配模块在不同信道配置下的兼容性和可靠性提出较高的要求。

3) CORDIC 与 FFT/IFFT 算法的实现

坐标旋转数字计算机 (Coordinate Rotation Digital Computer, CORDIC) 算法基于简单的二维几何原理，通过坐标轴的旋转进行计算。CORDIC 的迭代公式由 Volder 提出^[47,48]，用于计算三角函数、乘法和除法。该算法将复杂的计算分解为简单的移位-累加迭代过程，通过在线性、双曲和圆形轨迹中选择旋转模式或矢量模式来执行多个计算任务^[49]。在通信系统中，CORDIC 算法可以完成对数字信号的各种处理，包括计算信号的正弦和余弦值、对信号进行相位补偿等。由于 CORDIC 算法可以充分发挥硬件电路可以快速计算大量简单算法的特点，关于 CORDIC 算法的硬件实现被广泛研究。Aggarwal 等提出了一种基于可重构 CORDIC 的新型算法架构并给出了该算法的硬件实现细节和性能分析^[50]，通过使用这种算法架构可以实现更高的计算效率和灵活性。在数字设计中，通常使用赛灵思公司的 CORDIC 知识产权 (Intellectual Property, IP) 核心进行计算，但该 IP 核针对各种不同的计算场景设计了多种复杂的结构，其中部分功能是本文研究的 LTE-V2X 系统所不需要的。

4) FFT 与 IFFT 算法的实现

快速傅里叶变换 (Fast Fourier Transform, FFT) 是一种快速计算离散傅里叶变换的算法, 通过将傅里叶计算的矩阵进行分解实现快速变换^[51]。该算法最早由 Cooley 和 Tukey 提出^[52], 在此基础上, 之后的研究提出了基于蝶形的计算方法。在硬件实现时, 对于较大点数, 如 2048 点, 实现 FFT 计算的难点在于延时和资源利用率这两个方面。目前主流的硬件实现方案主要有流水和非流水^[53]、基 2 和基 4 等结构^[54, 55]。虽然赛灵思公司的 FFT IP 核可以满足工程的需求, 但和 CORDIC IP 核一样都是通用 IP 核, 没有针对 LTE-V2X 工程进行定制化设计, 在硬件设计复杂度方面还有改进的空间。其他现有的开源设计尽管可以满足计算功能和延时的需求, 但资源开销过大, 且不支持 307.2MHz 的系统工作频率。

根据以上调研, 本文的研究基于 LTE-V2X 协议, 使用 SC-FDMA 技术作为基带芯片的解决方案。对无线通信系统中的同步算法、解速率匹配算法、CORDIC 算法、FFT 和 IFFT 算法进行研究, 并且基于 FPGA 设计实现了 LTE-V2X 发送机和接收机中对应的关键模块。

1.3 现有研究的不足

现有研究的不足主要可以归纳为以下两个方面:

一方面, V2X 技术需要实现较为复杂的信号处理算法, 这些算法增加了基带芯片的计算负担, 导致处理延时和功耗的增加。目前, 已有无线通信算法的实现大多基于传统 LTE 技术, 没有针对 LTE-V2X 的应用场景进行特定的优化, 在算法性能和硬件实现复杂度方面还有优化的空间。这为本文第三章和第四章研究的同步算法改进和硬件实现结构的优化提供了研究方向。

另一方面, 对于关键的数字信号处理模块, 目前在无线通信系统设计中一般使用赛灵思等公司的 IP 核。虽然这些封装好的计算模块可以自定义配置, 但 LTE-V2X 系统通常只需要特定的配置, 可配置性的设计可能会成为冗余的部分, 导致硬件结构复杂度和资源开销的增加。此外, 现有的开源设计无法满足本文研究的 LTE-V2X 无线通信系统对于工作频率、延时和硬件资源开销的需求。这为本文第五章 CORDIC 和 FFT 计算模块的设计提供了研究动机。

1.4 研究内容和论文结构

1.4.1 主要的研究内容

本论文是以作者攻读硕士学位期间的无线芯片设计工作为基础,主要研究包括基于 LTE-V2X 收端同步算法的实现、解速率匹配技术的实现、CORDIC 和 2048 点 FFT/IFFT 计算模块的设计以及 LTE-V2X 的系统集成工作。图 1.2 展示了本文的研究内容结构和整体章节之间的联系。

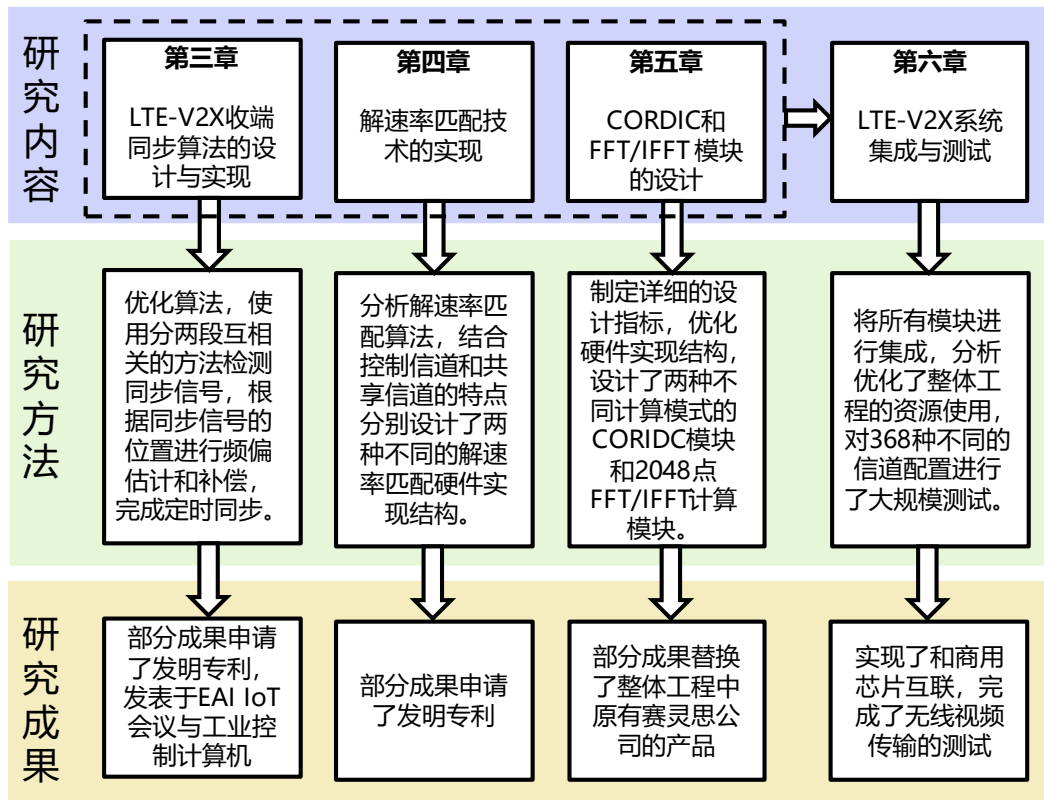


图 1.2 本文研究内容结构和整体章节的关系图

主要研究内容和创新点如下:

首先, 针对接收机的同步问题, 本文在传统算法的基础上进行改进, 提出了一种硬件友好的同步算法。提出的同步算法使用两段信号进行互相关的算法检测同步信号, 根据同步信号的位置完成接收信号子帧头的标记与频偏补偿。然后, 本文基于 FPGA 对提出的算法进行了硬件实现并测试, 测试结果表明设计的同步模块可以准确地完成 LTE-V2X 接收机的定时同步。

其次, 针对不同信道配置环境下发端和收端传输速率不匹配的问题, 本文结

合控制信道和共享信道的特点分别设计了两种不同的解速率匹配模块的硬件实现结构。设计的结构能够有效降低解速率匹配算法硬件实现的复杂度，可以在接收机中对信号进行解速率匹配，便于后级解码器的处理。

接着，为满足发射机和接收机中解调参考信号估计和对信号进行相位补偿的需求，本文分别设计并实现了支持正余弦计算模式和相位旋转模式的 CORDIC 模块。此外，为实现调制和解调过程中信号在时域和频域之间的转换，本文设计了流水结构的快速傅里叶变换和快速傅里叶逆变换模块。经过实际对比测试，设计的信号处理模块可以替换工程中原有赛灵思公司的产品，在可综合时钟频率、资源和精度方面均可以满足 LTE-V2X 工程的要求。

最后，本文将设计的模块整合到 LTE-V2X 收发机中，完成了系统的集成与测试。这部分的工作优化了整体工程的延时和资源消耗，进行了大规模的仿真和实际测试，实现了和商用芯片的互连测试，支持无线视频的传输，验证了整体设计的可靠性。

1.4.2 论文结构安排

本文总共分为七章，每一章的研究内容如下所述：

第一章是本文的绪论，首先介绍了研究课题的背景和应用意义，然后介绍了车联网通信技术的研究现状和本文的研究动机，最后介绍了本文的主要研究内容和章节的结构。

第二章是 LTE-V2X 系统的概述。依次介绍了 LTE-V2X 协议中物理层的帧结构、时间和频率资源的分配、发射机和接收机的整体设计方案和本文研究模块的功能细节，同时介绍了开发环境和开发的具体流程。

第三章介绍了接收机中同步模块的设计。首先提出了一种便于硬件实现的同步算法，然后介绍了同步模块中各个子模块涉及的算法和具体的硬件实现结构，最后对设计的同步模块进行了仿真和实际测试并分析了测试结果。

第四章介绍了接收机中解速率匹配模块的设计。首先分析了解速率匹配算法，对算法进行了建模仿真，然后针对不同信道的需求设计了两种不同的结构，有效降低硬件资源开销和算法复杂度。

第五章介绍了 CORDIC 与 FFT 计算模块的数字设计。首先研究了 CORDIC

和 FFT 算法的原理和系统的具体需求，然后根据预先制定的指标完成算法的建模仿真和基于 FPGA 的实现，并和赛灵思公司同类产品硬件资源消耗、延时和精度方面进行了对比。

第六章介绍了 LTE-V2X 系统的集成与测试。首先介绍了整体的系统集成结构和具体的测试方案，然后对系统工程的时序和资源使用进行了分析。接着分别完成了固定长度数据包和无线视频传输测试，并实现了和商用大唐芯片的互连。

第七章对全文内容进行总结，并提出了不足和改进方向。

第二章 LTE-V2X 系统概述

本章是 LTE-V2X 系统概述。首先，本章介绍了 LTE-V2X 物理层协议，包括无线帧结构、系统带宽和时隙架构。其次，本章介绍了系统的整体设计，随后，依次介绍了发射机和接收机的设计方案。最后，本章介绍了系统的开发环境和具体流程，以及实际测试的硬件平台。

2.1 LTE-V2X 物理层协议

2.1.1 LTE-V2X 物理层无线帧结构

与传统的 LTE 系统的帧结构不同，在 LTE-V2X 物理层协议中的帧结构针对车联网的特点做了优化，例如，LTE-V2X 可以根据特定的需求灵活分配时频资源，简化了帧结构、减少了传输控制信息的量，从而提高了数据传输的效率和吞吐量。此外，这种简化的帧结构还能够降低通信延迟、提高系统的实时性能。

根据 3GPP 第 14 代协议的标准，LTE 的模式有时分双工（Time Division Duplexing, TDD）模式和频分双工（Frequency Division Duplexing, FDD）模式^[56]。这两种模式的帧结构有所不同，TDD 的发送信道和接收信道在时间上进行划分，FDD 模式的发送信道和接收信道则是在频段上进行划分。虽然 TDD 模式频段配置更加灵活，但由于 TDD 模式在时域上分离发送信道和接收信道，所以对同步的要求较高^[57]。另外，为避免串扰，TDD 模式需要使用较长的保护间隔，导致频谱资源的消耗增加^[58]。考虑到车联网高机动性、低延迟的应用场景，本文中研究的 LTE-V2X 物理层协议采用 FDD 模式的无线帧结构，如图 2.1 所示。

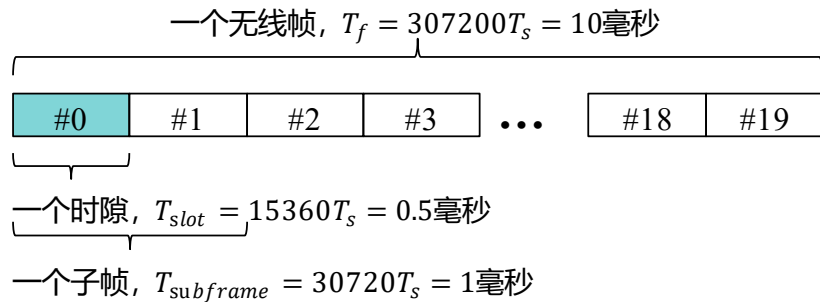


图 2.1 LTE-V2X 无线帧结构

其中, T_s 是无线信号的采样周期, 一个 LTE-V2X 无线帧的长度 T_f 是 10 毫秒, 在设计的 LTE-V2X 系统中共有 307200 个采样周期, 分为 20 个时隙, 每个时隙长度为 0.5 毫秒, 每两个时隙构成一个子帧, 其中每个子帧的长度为 1 毫秒, 共由 14 个 SC-FDMA 符号组成。

2.1.2 LTE-V2X 的系统带宽和时隙架构

LTE-V2X 物理层协议使用 SC-FDMA 技术进行信号的发送和接收, 并支持 10MHz 和 20MHz 的信道带宽。合适的时频资源分配策略可以有效提高 LTE-V2X 系统的通信性能并满足用户的特定需求^[59], 在本文研究的 LTE-V2X 系统中, 选择的配置如表 2.1 所示。

表 2.1 LTE-V2X 物理层系统参数配置

系统带宽	循环前缀种类	N_{RB}	N_{symbol}^{SL}	N_{SC}^{RB}
20MHz	正常循环前缀	100	7	12

其中系统带宽选择了 20MHz, 循环前缀选择的是正常循环前缀, N_{RB} 、 N_{symbol}^{SL} 和 N_{SC}^{RB} 是系统参数, 其值依次表示 LTE-V2X 系统传输信息的大小有 100 个资源块 (Resource Block, RB)、一个时隙在时域上有 7 个 SC-FDMA 符号、每个 SC-FDMA 符号在频域上有 12 个子载波。一个资源单位 (Resource Element, RE) 由一个 SC-FDMA 符号和一个子载波构成, 一个资源块 (Resource Block, RB) 由时域上连续 7 个 SC-FDMA 符号和频域上 12 个子载波构成, 如图 2.2 所示。

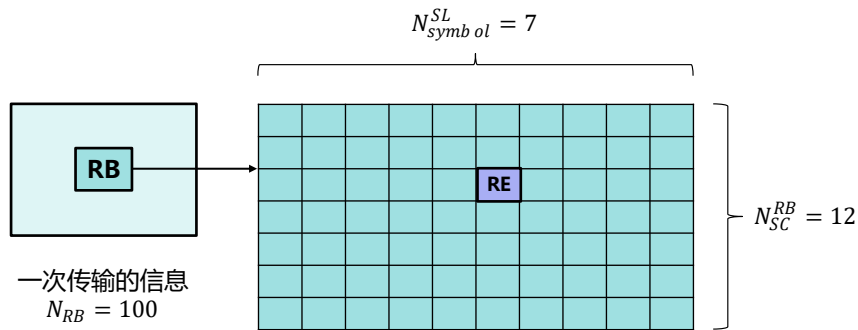


图 2.2 传输信息的格式

目前 LTE-V2X 系统实现中主要有两个信道, 包括控制信道和共享信道。控制信道的调制方式为四相移键控调制^[60], 用于处理 LTE-V2X 系统中的控制信息,

计算相关的配置参数，得出控制信息后将这些信息传递给共享信道^[61]。共享信道负责对信息数据进行处理，支持的调制方式有三种，分别为四相移键控调制、16进制正交幅度调制和 64 进制正交幅度调制^[62]。

2.2 LTE-V2X 系统框架

LTE-V2X 整系统框架如图 2.3 所示，主要包括上位机、先进精简指令集计算机（Advanced RISC Machines, ARM）^[63]、发射机和接收机这四个部分。

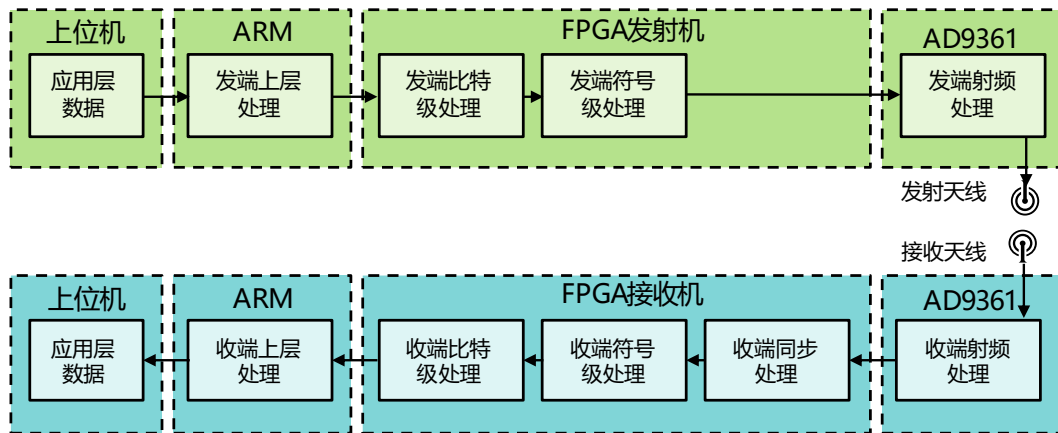


图 2.3 LTE-V2X 系统框架示意图

其中各个模块的具体描述如下。上位机负责发送需要传输的应用层数据，在目前的 LTE-V2X 系统中是个人电脑，未来在车联网中将使用车载电脑。ARM 代表 ARM 处理器，负责对上位机传输来的信息进行编码转换等操作，可以高效完成复杂的运算。FPGA 发射机是 LTE-V2X 的发端，负责对发送信息进行编码和调制，然后转换成无线信号进行发射。FPGA 接收机是 LTE-V2X 的收端，负责对接收到的信号进行解调和解码，恢复出发送的原始信息。比特级处理指的是原始二进制信号在比特层面进行处理，包括循环冗余校验（Cyclic Redundancy Check, CRC）^[64]、速率匹配^[65]、加扰^[66]等。符号级处理指的是在 SC-FDMA 符号层面进行的处理，包括调制、资源映射等操作。AD9361 指的是目前 FPGA 开发板上使用的 ADC 芯片，负责对芯片端的数字信号和天线端的模拟信号进行相互转换。

LTE-V2X 系统对传输数据的处理过程如下：

- 1) 首先，上位机选择要发送的数据，例如视频音频信息，通过 ARM 进行上

层处理，包括压缩、加密等操作。接着 ARM 将信息传输给发射机，在发射机中根据 LTE-V2X 的协议依次进行比特级处理和符号级处理，处理完成后由发射机将数字信号通过模数转换（Analog-to-Digital Converter, ADC）芯片转换为模拟信号，并通过天线进行发送。

2) 接收机通过天线接收信号，并通过 ADC 芯片将模拟信号转换为数字信号。收端同步模块负责对接收到的数字信号进行定时同步，估计频偏并进行相应的频偏补偿，这部分也是本文的主要研究内容之一。对信号完成同步后依次进行符号级处理和比特级处理，接着传输给 ARM 进行上层处理。最终，ARM 将解码的信息发送给上位机，还原出原始的视频音频信息。

2.3 发射机系统结构

2.3.1 发端比特级处理模块

发射机中比特级处理模块位于发端上层处理模块和发端符号级处理模块之间，其内部具体结构如图 2.4 所示。

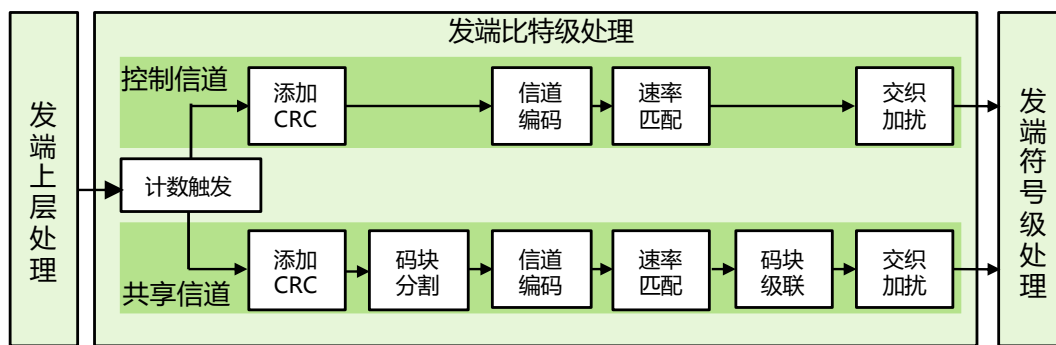


图 2.4 发端比特级处理模块

首先，发端比特级处理模块接收 ARM 上层处理后的数据。然后，发端的计数触发器根据计数结果将数据分配给控制信道和共享信道，在这两个信道中分别对传输的控制信息和数据信息进行处理。对于控制信道，信号将依次进行添加 CRC、信道编码、速率匹配、交织和加扰处理。对于共享信道，信号将依次完成添加 CRC、码块分割、信道编码、速率匹配、码块级联、交织和加扰处理。经过控制信道和共享信道处理后的信号会传输到后级的发端符号级处理模块进行后

续的符号级处理。

2.3.2 发端符号级处理模块

发端符号级处理模块负责对比特级模块处理过的数据进行在 SC-FDMA 符号级的处理。如图 2.5 所示，首先，信号分别进入控制信道和共享信道，两个信道均对信号完成调制和离散傅里叶变换（Discrete Fourier Transform, DFT）处理。处理后的信号根据下行链路解调参考信号（Downlink Mapping Reference Signal, DMRS）序列生成器生成的参考信号完成资源映射操作。为提高发射信号的质量，映射后的信号还需要依次进行补零与偏移、IFFT 计算、添加循环前缀、多址处理和频偏预补偿操作。最后，处理后信号在发端射频处理模块中使用 AD9361 芯片实现数字信号向模拟信号的转换，随后通过天线进行发送。

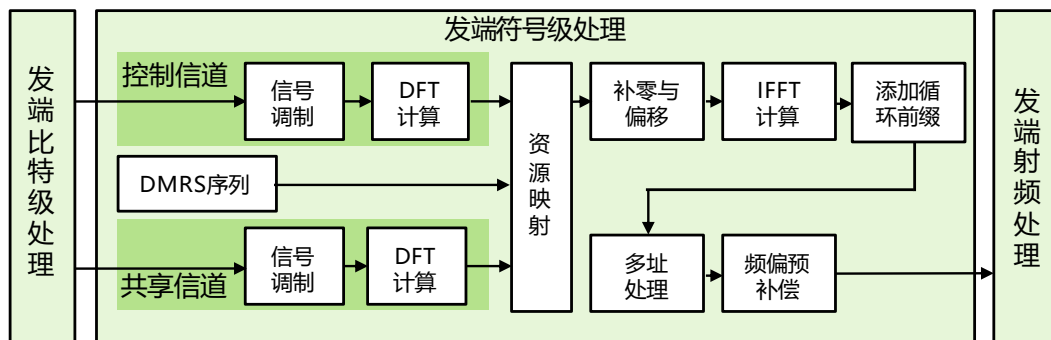


图 2.5 发端符号级处理模块

本文部分研究内容和发端的符号级处理模块相关，设计的 IFFT 模块用于完成其中的 2048 点 IFFT 变换，实现调制后的信号从频域向时域的转换。根据具体功能的需求，本文设计了两种计算模式的 CORDIC 模块，设计的正余弦计算模式 CORDIC 模块负责完成 DMRS 序列生成器中正弦值和余弦值的计算，相位旋转模式 CORDIC 模块则负责完成发端信号的频偏预补偿。

2.4 接收机系统结构

2.4.1 收端符号级处理模块

收端符号级处理模块的具体流程如图 2.6 所示。首先，收端的天线接收信号

并在收端射频处理模块中将模拟信号转换为数字信号,接着在收端同步模块中完成接收信号的定时同步。同步后的数据在计数触发器的控制下依次进行自动增益控制、双天线处理、多址处理、去循环前缀和 FFT 计算以及偏移和去零操作。解映射模块可分离出原始控制信道信息和共享信道信息,分离出的信息根据 DMRS 参考序列分别进行控制信道解码和共享信道解码。

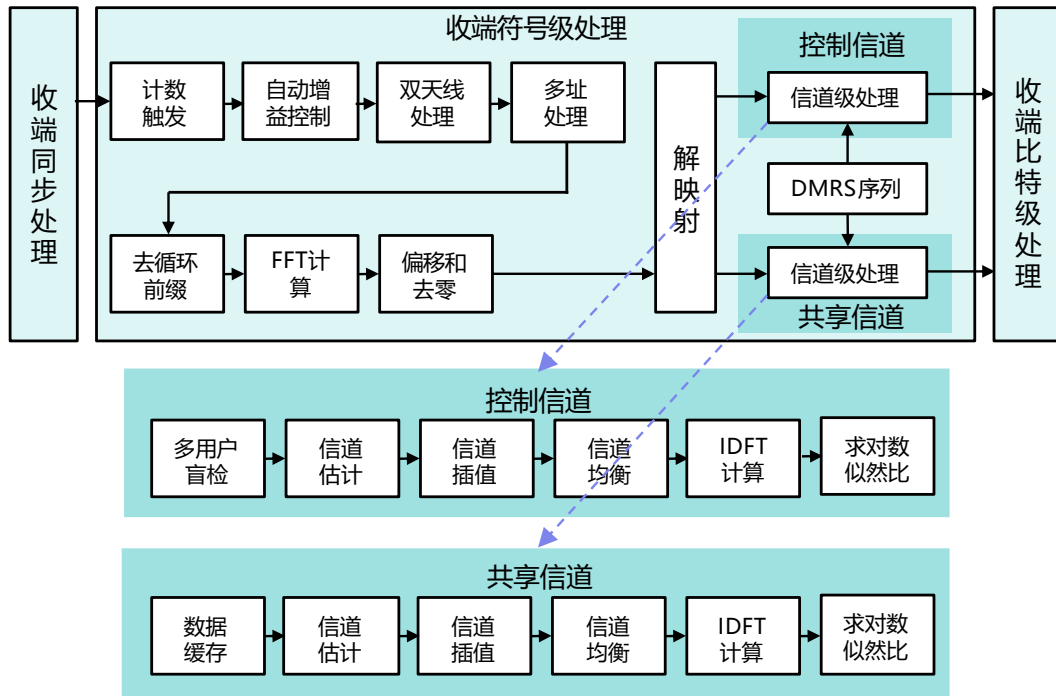


图 2.6 收端符号级处理模块

在控制信道解码过程中,信号需要依次完成多用户盲检、信道估计、信道插值、信道均衡、离散傅里叶逆变换(Inverse Discrete Fourier Transform, IDFT)计算和求对数似然比计算。在共享信道解码中,信号将依次进行数据缓存、信道估计、信道插值、信道均衡、IDFT 计算和求对数似然比计算。控制信道和共享信道全部处理完成后,信号被传输给下一级的收端比特级处理模块。

本文研究了图 2.6 中最左侧的收端同步处理模块。在同步模块中需要使用相位旋转模式的 CORDIC 模块对接收到的信号进行频偏补偿。此外本文还设计实现了图中的 FFT 计算模块,在这里的作用是将从前级模块中提取出的信号进行时域到频域的变换,便于后级解映射模块在频域上分离控制信道和共享信道。

2.4.2 收端比特级处理模块

收端的比特级处理模块如图 2.7 所示。首先，从收端符号级处理模块传来的信息分别进入对应的控制信道和共享信道。在控制信道中处理的是控制信息，信号依次进行解扰、解交织、解速率匹配、信道解码和 CRC，数据校验成功后在侧链控制信息（Sidelink Control Information, SCI）参数计算模块中计算出和控制信息相关的参数，将这些参数传递给上一级的收端符号级处理模块。在共享信道中处理的是数据信息，信号依次完成解扰、解交织、码块分割、解速率匹配、信道解码和 CRC，校验成功后得到正确的数据信息。最后，将解出的数据信息发送给收端上层处理模块。

本文针对收端比特级处理模块中的解速率匹配模块进行研究。根据控制信道和共享信道各自的特点，本文实现了两种不同的解速率匹配硬件设计方案。

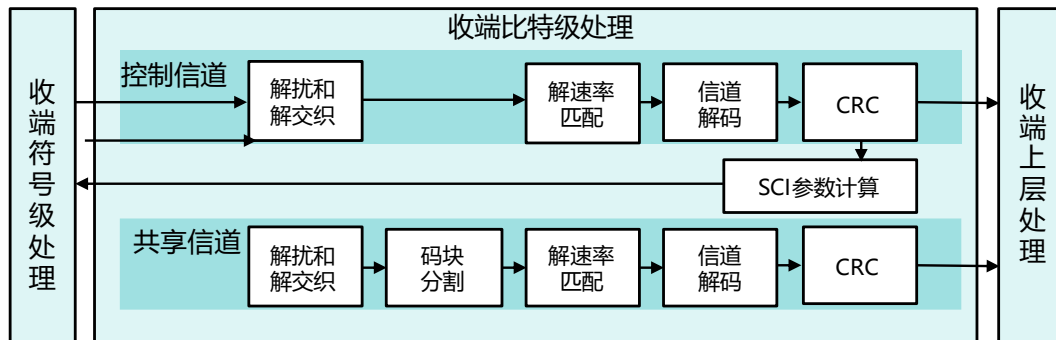


图 2.7 收端比特级处理模块

2.5 开发环境简介

2.5.1 开发流程

本文研究的 LTE-V2X 关键模块开发流程主要分为三部分，分别是算法开发、数字设计和开发板调试，如图 2.8 所示。

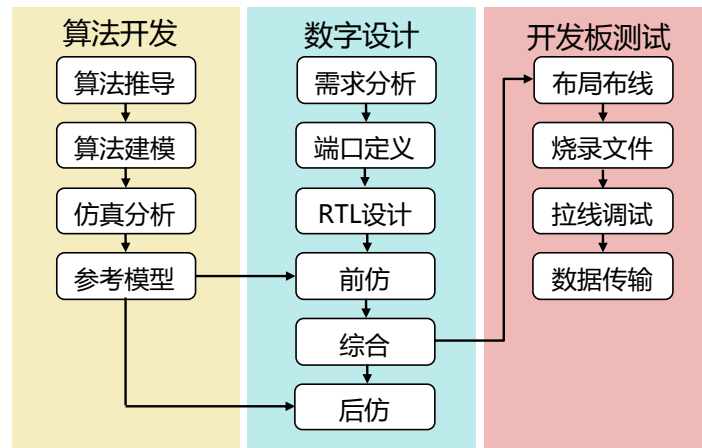


图 2.8 总体开发流程图

第一部分是算法开发流程。在算法开发过程中，依次进行算法推导、编写 MATLAB 代码完成算法建模、仿真分析和参考模型的搭建，便于后续验证模块功能的正确性^[67]。

第二部分是数字设计过程。首先需要设计需求分析，其中包括定义量化位宽、系统允许的计算延时、最高综合频率、可使用的资源等。进行详尽且具体的需求分析能显著提升后期设计的效率，并避免因需求更改导致的重复劳动。完成需求分析后在 VIVADO 软件中使用 Verilog 编程语言开始具体的设计^[68,69]。设计过程中先定义好每个模块的输入输出端口，包括端口定义和位宽，接着根据 MATLAB 参考模型编写寄存器传输级（Register-Transfer Level, RTL）代码实现对应的功能，并利用 SPYGLASS 软件对代码进行检查。接着依次对设计的模块进行前仿、综合以及后仿。其中，前仿指综合前的功能仿真，主要使用 VIVADO 软件自带的仿真器和服务器上的 VCS 仿真器完成仿真过程，通过将仿真结果和 MATLAB 的仿真结果进行对比验证功能的准确性。综合主要使用 VIVADO 软件和 DC 综合工具，主要功能是将设计的 RTL 电路映射为与工艺库相关的网表，根据综合的结果检查时序和资源，进行时序违例的修复和资源的优化。后仿是对综合后生成的网表文件进行仿真分析，检查功能和时序是否满足要求。

第三部分是开发板调试。在完成充分的仿真验证后，对综合后的文件进行布局布线，生成比特流文件。将比特流文件烧录到 FPGA 开发板中进行实际的板级测试，根据预留的测试信号和串口数据验证实际测试的正确性和稳定性，最后进行数据传输的测试，其中数据包括固定长度的数据包和视频编码后的数据包。

为保证设计的可靠性，需要对所设计的模块进行充分的验证，本文采用的功能验证方法如图 2.9 所示。

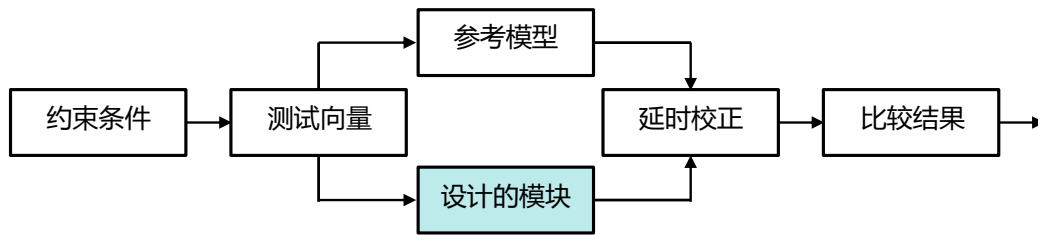


图 2.9 本文使用的验证方法

其中，测试向量使用 MATLAB 产生的参考数据、测试脚本文件内部产生的随机数和开发板实际测试过程中采集的实时数据，参考模型使用 MATLAB 编写的程序和赛灵思公司的 IP 核。开始验证时，将测试向量作为输入数据分别输入到设计的模块和对应的参考模型中，用不同平台完成相同的功能，将两者的输出结果进行延时校正后，比较各自的输出结果，若两者的结果一致，则说明该功能通过验证。

2.5.2 实际测试的硬件平台

硬件测试平台使用实验室课题组自研的 LTE-V2X 通信系统板卡，搭载的芯片型号为 ZYNQXC7Z045-FFG900ABX21，集成了 ARM Cortex-A9 双核处理器，可以实现高度可编程。如图 2.10 所示，开发板左侧的三个接口依次为以太网接口、迷你通用串行总线接口和电源接口。以太网接口负责传输网络层的信号，迷你通用串行总线接口集成了烧录和串口功能，不仅可以作为 FPGA 编译比特流的烧录线，还可以使用串口协议和用户电脑进行通信，将信号传输并显示在电脑上，电源接口负责对开发板供电。

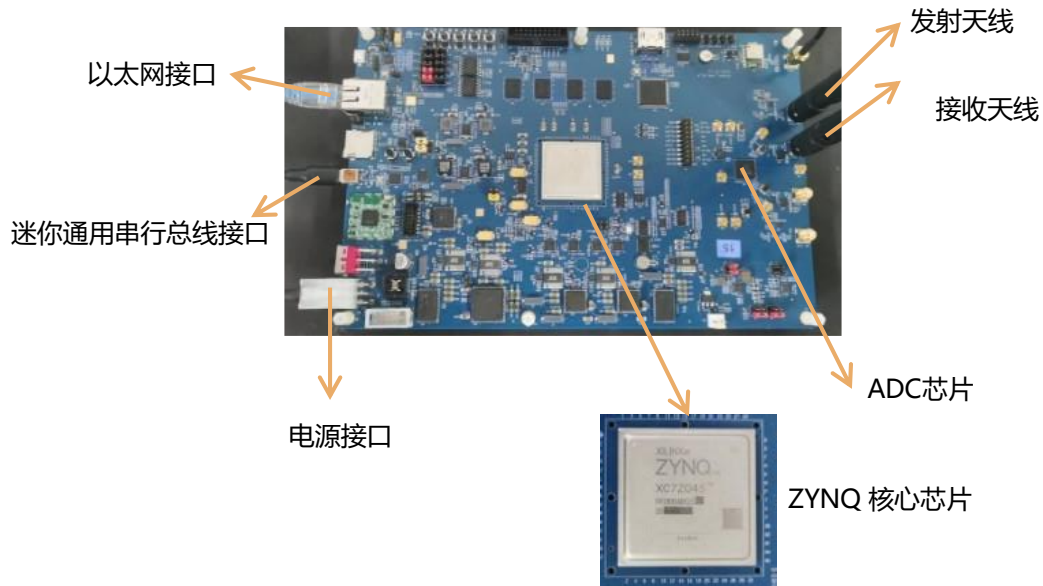


图 2.10 定制的 FPGA 开发板配置情况

自研的开发板集成了收发的天线和 AD9361 型号的高速 ADC 芯片，通过天线和 ADC 芯片可以完成收端和发端射频处理过程中模拟信号到数字信号的转换，因此本文只需研究数字信号处理的部分。

2.6 本章小结

本章首先介绍了 LTE-V2X 的物理层协议，包括无线帧结构、系统带宽和时隙架构。接着，本章简单介绍了 LTE-V2X 物理层的整体系统设计。随后，本章对发射机和接收机的整体设计结构进行介绍，其中包括符号级和比特级处理的具体流程，引出本文研究内容在整个 LTE-V2X 系统设计中的位置和功能。最后本章描述了具体的开发环境，包括算法开发、数字开发和开发板测试三部分，并介绍了验证方案和使用的具体硬件平台。

第三章 LTE-V2X 收端同步算法的设计与实现

本章提出并实现了一种面向 LTE-V2X 系统的收端同步算法。首先，本章介绍了同步技术的背景、传统同步算法的原理和局限性，说明了本章的研究动机。然后，本章对接收机的同步算法进行了研究，根据 LTE-V2X 物理层协议提出了一种硬件友好的同步算法，对算法进行建模和 FPGA 实现。最后本章对设计的同步模块进行了仿真验证和实际测试并对结果进行了分析。

3.1 引言

3.1.1 同步技术的背景

在车联网通信场景下，由于车辆与车辆、车辆与道路基础设施之间相对高速移动，并且道路环境复杂，存在噪声、多径效应干扰和多普勒效应干扰，因此通信的可靠性和稳定性对于安全驾驶尤其重要。在无线通信系统中，噪声主要是来自于电气装置的电磁辐射和其他干扰信号产生的环境噪声。多径效应指的是电磁波通过不同的传输路径后，不能同时到达接收端，进而发生相位偏移叠加导致的原始信号失真现象^[70]。多普勒效应指的是由于发端和收端的相对运动导致信号频率发生变化的现象^[71]。上述两种效应都会导致传输信号的频谱发生变化。

本文研究的 LTE-V2X 系统基于 SC-FDMA 技术，该技术使用单个连续载波来传输不同用户的数据信息，需要保证每个用户的信息在时间和频率上是严格正交的，从而避免干扰。为保证信号时间和频率的正交性，LTE-V2X 接收机需要设计合适的同步模块，对天线接收到的信号进行时间和频率上的同步。同步过程中，时间同步过程需要保证接收机正确检测到接收信号的起始时间，从而进行后续的解调解码等操作。频率同步过程需要对接收信号的相位偏移进行估计并进行相应的相位补偿，消除传输过程中由于噪声、多径效应和多普勒效应造成的频率偏移。如果不对频偏进行补偿，后续模块将无法正确地解调出信号。

针对上述这些需求，本文设计了一种针对 LTE-V2X 系统的收端同步模块，提出了一种硬件友好的同步算法并基于 FPGA 进行了硬件实现。

3.1.2 传统的同步算法

在无线通信系统中，对于同步技术的研究，主要关注点在于同步算法的设计和硬件实现的优化。为实现同步，往往在发端发射的信号中插入一段特定的同步信号序列，在收端根据同步信号序列的特点进行检测从而得到接收信号中同步信号的位置，进而根据通信系统的物理层协议推算出无线帧的起始位置。同步信号的检查可以通过比较两个信号之间的相似性来实现，即采用相关的计算方式。传统的同步算法中涉及三种计算相关的算法，分别是互相关算法、自相关算法和部分互相关算法。首先，对接收到的信号进行相关计算，然后，找到相关计算结果的最大值，凭借最大值确定是否检测到同步信号以及同步信号的位置。三种相关算法计算公式分别如式（3.1）、（3.2）和（3.3）所示。

$$R = \max_n \left[\left| \sum_{i=0}^{N_{FFT}-1} Y(i) X^*(i+n) \right|^2 \right] \quad (3.1)$$

$$R = \max_n \left[\left| \sum_{i=0}^{\frac{N_{FFT}-1}{2}} Y(i+n) Y^* \left[(N_{FFT}-i)+n \right] \right|^2 \right] \quad (3.2)$$

$$R = \max_n \left[\sum_{m=0}^{N_{PB}-1} \left| \sum_{i=m \left(\frac{N_{FFT}}{N_{PB}} \right)}^{\frac{(m+1)N_{FFT}-1}{N_{PB}}} Y(i) X^*(i+n) \right|^2 \right] \quad (3.3)$$

式中， R 表示相关峰值结果， $X(i)$ 是本地参考主同步信号（Primary Synchronization Signal, PSS）， $Y(i)$ 是包含 PSS 信息的接收信号， i 是输入信号的索引， n 是相关结果的索引， N_{FFT} 表示选取信号样本的大小， N_{PB} 表示分段的总数， m 表示当前运算的序列编号。式（3.1）是互相关算法的计算公式，将接收到的信号与本地参考主同步序列进行相关计算，遍历所有结果找到最大值。式（3.2）是自相关算法的计算公式，将接收到的信号和自身进行相关计算，然后遍历找到最大值。由于未采用标准的参考信号，因此自相关算法的抗频偏和抗噪声能力最弱。式（3.3）是部分互相关算法的计算公式，该算法与第一种算法类似，待测信号是接收到的信号，参考信号是本地存储的主同步序列。部分互相关算法计算时先将接收信号和参考信号按相同的方式进行分段，然后分别计算每一段的相关结果再进行累加，最后遍历所有结果找到最大值。由于部分互相关算法中使用了分段的方式，

减少了随机噪声对于计算结果的影响，因此这种算法的抗噪声干扰能力较强。

虽然上述三种算法都可以计算出主同步信号的位置，但如果按照对应算法直接进行硬件实现，需要大量的存储和逻辑运算资源，不利于 FPGA 的实现。为解决上述问题，本文提出了一种分段互相关和局部遍历相结合的同步算法。该方法利用分两段进行互相关的方式抵抗频偏和噪声的干扰，同时利用 PSS 互相关峰值的特点在时域上只选取部分信号的计算结果进行遍历确定峰值，降低了硬件设计的复杂度，减少了 FPGA 资源和面积的开销。

3.2 基于 LTE-V2X 的同步算法

3.2.1 同步信号

为了使 LTE-V2X 接收机可以确定每个无线帧的起始和结束位置，需要使用同步技术确定传输过程中每个无线帧的起始位置。根据 LTE-V2X 无线帧结构的特点，只需要确定子帧 0 的位置即可推断出其他无线帧的位置。同步信号分为主同步信号和辅同步信号（Secondary Synchronization Signal, SSS）。这两种同步信号在无线帧中的位置如图 3.1 所示，其中，PSS 位于子帧 0 的第二和第三个符号位置处，可用于检测子帧 0 的位置和估算接收信号小数倍的频率偏移量。SSS 用于估算整数倍的频率偏移量。由于目前的 LTE-V2X 项目中未涉及到整数倍频率偏移，因此本章的同步技术只关注 PSS 的检测。

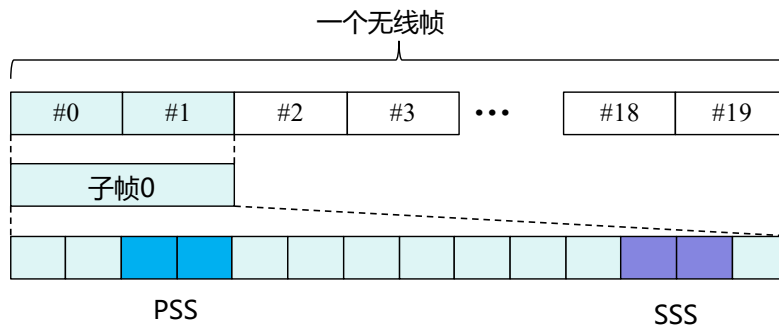


图 3.1 同步信号在无线帧中的位置

PSS 序列使用 Zadoff-Chu 序列。Zadoff-Chu 序列是由 D.C Chu 提出的一种 Frank-Zadoff 序列^[72, 73]，数值为复数，该序列具有良好的互相干特性，能够抵挡

一定的噪声干扰和频率偏移。在 LTE-V2X 物理层中, PSS 在频域中生成, 表达式如式 (3.4) 所示。

$$d_u(n) = \begin{cases} e^{-j\frac{\pi un(n+1)}{63}} & n = 0, 1, 2, \dots, 30 \\ e^{-j\frac{\pi un(n+1)(n+1)}{63}} & n = 31, 32, 33, \dots, 61 \end{cases} \quad (3.4)$$

式中, $d_u(n)$ 是产生的 PSS 序列, u 是产生一段特定序列的根索引, 改变 u 的值可以得到不同的 PSS 序列。本文研究同步技术使用的两个 PSS 序列根索引分别是 26 和 37。

3.2.2 提出的同步算法

1) 相关算法

参考传统的同步算法, 本文采用了分两段的方式进行互相关计算。对接收信号进行预处理后, 在时域上选取两段长度相同但延时相差 137 个采样周期的信号, 将选取的两段信号和 PSS 参考序列分别进行互相关计算, 并将计算结果进行叠加, 如式 (3.5) 所示, 以此可以减少噪声和频率偏移对同步性能的影响。

$$R_{pss128}(n) = \left| \sum_{i=0}^{127} rx_{128}(n+i) \cdot S_{pss128}^*(i) \right|^2 + \left| \sum_{i=0}^{127} rx_{128}\left(n + \frac{2048+144}{16} + i\right) \cdot S_{pss128}^*(i) \right|^2 \quad (3.5)$$

式中, $R_{pss128}(i)$ 是互相关的计算结果, n 是互相关计算结果的索引, i 是在接收信号中选取 128 个采样点的索引, $rx_{128}(i)$ 指接收机接收信号依次经过有限脉冲响应 (Finite Impulse Response, FIR) 低通滤波和 16 倍降采样处理后产生的信号, $S_{pss128}^*(i)$ 是根据 PSS 序列生成公式产生后存储在本地的参考 PSS 序列。接收到的信号经过预处理后和本地的参考 PSS 序列进行互相关计算, 根据计算结果即可判断对应的原始信号中是否存在 PSS。由于设计物理层协议的时候选取了两条根索引不同的 PSS 参考序列, 因此本文在具体实现时使用两条不同的 PSS 参考序列进行并行计算。如果计算结果中出现了互相关的峰值, 即可确定匹配的 PSS 序列。

在车联网无线通信场景中, 发射机和接收机之间的距离变化、发射功率和接收功率的调整、障碍物的出现导致信号在传输过程中的衰减等因素都会导致接收

机接收信号的幅值发生变化。由式（3.5）可以得出，若 $rx_{128}(i)$ 的大小发生变化，那互相关结果 $R_{pss128}(i)$ 也会随之改变。因此，仅根据互相关计算结果不能确定检测到的是同步信号还是幅值较大的噪声信号。考虑到上述情况，此处引入能量计算的概念。信号能量由信号的幅值决定，因此，在完成互相关计算的同时计算其对应信号的能量。通过比较互相关计算结果和能量计算结果的相对大小对上述情况进行判断，即可消除信号幅值变化对结果的影响。接收信号的能量计算表达式如式（3.6）所示。

$$E_{rx}(n) = \sum_{i=0}^{128-1} |rx_{128}(n+i)|^2 + \sum_{i=0}^{128-1} \left| rx_{128} \left(n + \frac{(2048+144)}{16} + i \right) \right|^2 \quad (3.6)$$

式中， n 是每个能量点的计算结果，为了和互相关计算结果进行比较，这里也选取了相同的两段 128 个采样点信号进行计算，对接收的信号 $rx_{128}(i)$ 进行自相关计算后就可以得出信号能量的值。

互相关计算和能量计算是并行的，具体实现时采用流水的设计，支持连续输入和输出。实时比较互相关结果和能量计算结果，如果满足式（3.7），就认为检测到了同步信号的峰值。其中 $R_{pss128}(i)$ 是互相关计算的结果， $E_{rx}(n)$ 是能量计算的结果， $R_{peak}(r)$ 是满足条件的峰值， r 是峰值在时域上的位置索引， ε 是能量阈值系数，该值会根据实际测试场景下的计算结果进行微调。由于同步序列的特点，实际计算时可能会产生多个满足式（3.7）的峰值，因此需要根据式（3.8）筛选出最大值。

$$R_{pss128}(n) > \varepsilon \cdot E_{rx}(n) \quad (3.7)$$

$$result = \max [R_{peak}(r)] \quad (3.8)$$

同步信号所产生的互相关峰值集中在同步信号时域上的半个子帧长度内，因此，不必对所有的计算结果进行遍历，选取时域上半个子帧长度（降采样后 960 个信号采样周期）内最大的峰值即可得到同步信号在时域上的位置。得到同步信号的位置后，根据 LTE-V2X 无线帧结构即可确定其他子帧的起始和结束位置。

2) 频偏估计和补偿算法

在实际传输过程中接收到的信号会发生频率偏移，为避免频率偏移对后续的解码造成影响，需要在完成同步信号的检测之后估算接收信号的频率偏移值并进

行适当的补偿。检测到同步信号之后可以确定接收信号中同步信号的位置，从该位置提取出接收信号中的同步信号序列，和本地存储的 PSS 参考序列进行相乘和累加计算，根据计算结果的相位角度就可以算出接收信号中同步信号相对于本地 PSS 参考序列的相位偏移，计算公式如式（3.9）和（3.10）所示。

$$C_{pss} = \sum_{i=0}^{63} \left(rx_{new128}(i) \cdot S_{pss128}^*(i) \right)^* \cdot \left(rx_{new128}(i+64) \cdot S_{pss128}^*(i+64) \right) \quad (3.9)$$

$$FFO_1 = \frac{1}{\pi} \angle(C_{pss}) \quad (3.10)$$

式中， $rx_{new128}(i)$ 是输入信号经过滤波和 16 倍降采样后的信号，起始位置是主同步信号所在的位置， $S_{pss128}^*(i)$ 是本地存储在只读存储器（Read-Only Memory, ROM）中主同步序列的共轭， C_{pss} 是频偏计算的累加值，对 C_{pss} 计算反正切即可得到接收信号小数倍的频率偏移值 FFO_1 。

完成频偏估计后，根据频偏估计值对原始的信号进行频偏补偿，补偿的公式如式（3.11）所示。

$$rx_{out}(n) = rx_{in}(n) e^{-j2\pi\epsilon \frac{n}{N}} \quad (3.11)$$

式中， $rx_{in}(n)$ 是频偏补偿的输入信号， $rx_{out}(n)$ 是补偿后的输出信号， ϵ 表示归一化的频率偏移量。 N 是一段信号的总长度，这里的值为 2048， n 是信号时域上的索引值。

3.3 同步算法的建模仿真

同步算法的建模仿真总共分成四步，分别是预处理、互相关计算、频偏估计和补偿以及定时同步，可以通过 MATLAB 进行实现。

在算法开始之前，先根据 LTE-V2X 物理层的帧结构预先生成一段包括 PSS 的数据信号并存储在文本文件中作为输入的测试向量，后续通过同步算法检测其中同步信号的位置。

建模仿真过程中，首先在 MATLAB 中读取预先生成好的测试向量，然后开始预处理。预处理包括 FIR 低通滤波和 16 倍降采样两部分。低通滤波的目的是滤除和同步信号无关的高频噪声，降采样则是为了减少需要处理的数据量，降低后续硬件实现的资源开销。FIR 低通滤波使用 MATLAB 内置的函数，设定滤波

器的阶数为 128 阶，截止频率为 0.5025MHz，采样频率为 30.72MHz，然后计算出滤波器的抽头系数，计算出的抽头系数也用于后续 FPGA 开发时的滤波器配置。由于在 FPGA 设计中无法直接使用小数进行计算，因此需要用定点的方案规定信号整数位和小数位的比特位宽大小，这里抽头系数也进行了定点处理。降采样的实现较为简单，对滤波后的信号每 16 个点选取一个点即可。

预处理结束后开始互相关计算。首先参考式 (3.4) 的 PSS 序列生成公式生成根索引分别为 26 和 37 的两段 PSS 参考序列，然后参考式 (3.5) 完成互相关计算。互相关计算结果如图 3.2 所示，左侧仿真结果中有一个主峰和两个次峰，说明检测到了同步信号，右侧是未检测到同步信号的结果。

检测到同步信号之后，为验证同步检测的结果，需要对同一段信号进行能量计算，计算后根据式 (3.7) 和 (3.8) 进行判断，确定是否检测到同步信号，并得出同步信号在原始信号中的位置索引值。

频偏估计和补偿的实现参考式 (3.9) 和 (3.10)。从检测到的同步信号索引位置处提取一段固定长度序列进行频偏估计的计算，然后根据频偏估计值对预处理前的接收信号进行频偏补偿，从而校正信号在传输过程中发生的频率偏移。

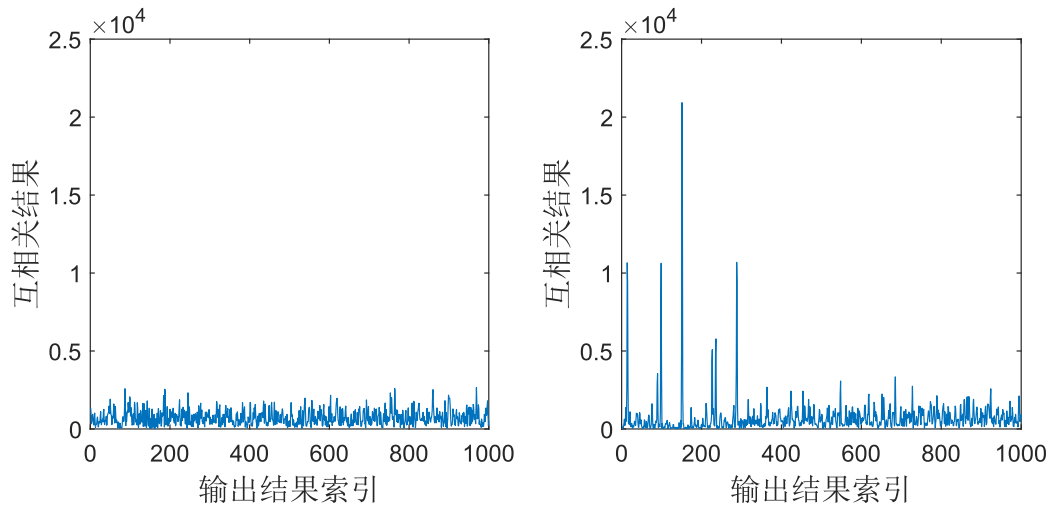


图 3.2 检测到 PSS (左) 与未检测到时 (右) 的互相关结果

3.4 FPGA 硬件实现方案

3.4.1 总体硬件设计方案

根据提出的同步算法进行硬件实现，总体硬件实现的框图如图 3.3 所示。根据主要功能划分为同步相关模块、频偏校正模块和定时标记模块。在同步相关模块中完成信号的预处理、互相关计算、能量计算、峰值检测和峰值局部遍历操作。在频偏校正模块中根据同步相关模块确定的主同步信号位置提取一段固定长度的信号进行频偏估计，随后对原始信号进行频偏补偿。最后输出补偿后的信号，同时标记输出信号的子帧头，便于后续的处理。

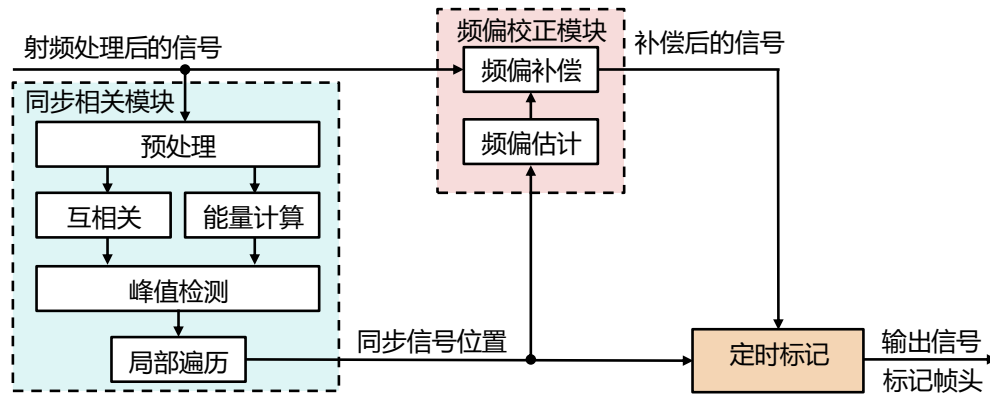


图 3.3 总体硬件实现框图

3.4.2 同步相关模块的设计

在硬件实现时，同步相关模块接收的信号首先需要进行预处理。预处理的具体过程包括 FIR 滤波器、最高有效位截位处理和 16 倍降采样三部分。首先使用赛灵思公司的 FIR Compiler 7.2 IP 核实现 FIR 滤波器，配置 IP 核的参数时抽头系数使用 MATLAB 仿真时生成的数据，滤波器类型配置为单一采样率模式。接着，使用最高有效位截位处理模块在不改变信号相对大小的情况下将滤波器输出数据的位宽从 36 比特截取为 16 比特。然后，通过计数器标记的方法完成 16 倍降采样的处理。

FIR 的计算方式与互相关的计算方式非常相似，都是需要进行流水的滑动计算。FIR 的计算公式如式（3.12）所示，其中 N 是滤波器的阶数， k 是滤波器输

出结果的索引， n 是输入数据的索引， $y(k)$ 是输出结果， $x(k-n)$ 代表输入信号， $a(n)$ 是抽头系数。图 3.4 是 FIR 的硬件计算结构。由于 FIR IP 核比较可靠且资源配置更加合理，本文直接利用 FIR 滤波器完成两段信号的互相关计算，其中，将 PSS 参考信号作为滤波器的抽头系数。

$$y(k) = \sum_{n=0}^{N-1} a(n)x(k-n) \quad k = 0, 1, \dots \quad (3.12)$$

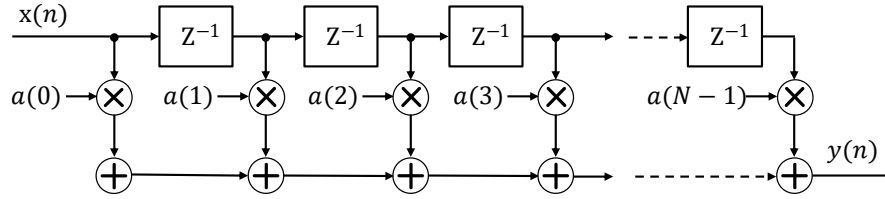


图 3.4 FIR 硬件计算的结构

由于输入数据为复数，此处需要对原算法进行处理，便于后续在 FPGA 上的实现。处理过程中，根据乘法结合律对每个点的复数乘法进行拆分，分别计算出复数相乘结果的实部和虚部，如式 (3.13) 所示。

$$\begin{aligned} Y(k) \cdot X^*(k) &= [Y_r(k) + Y_i(k) \cdot j] \cdot [X_r(k) - X_i(k) \cdot j] \\ &= [Y_r(k) \cdot X_i(k) + Y_i(k) \cdot X_r(k)] \\ &\quad + [Y_i(k) \cdot X_r(k) - Y_r(k) \cdot X_i(k)] \cdot j \end{aligned} \quad (3.13)$$

式中， $X(k)$ 与 $Y(k)$ 都是复数， $X_r(k)$ 和 $X_i(k)$ 分别表示 $X(k)$ 的实部和虚部， $Y_r(k)$ 和 $Y_i(k)$ 分别表示 $Y(k)$ 的实部和虚部。将本地参考 PSS 序列的实部和虚部分别作为 FIR 的抽头系数导入两个 FIR IP 核，每个 FIR IP 核配置为两个通道，分别计算出最终互相关结果的实部和虚部，如图 3.5 所示。将 FIR IP 核计算结果的实部和虚部分别进行平方后相加，得到信号模值的平方，即为实数形式的互相关结果。能量计算结果同样为模值的平方形式，因此，此处只需要计算实部平方和虚部平方的和，不需要进行开根号求模值。

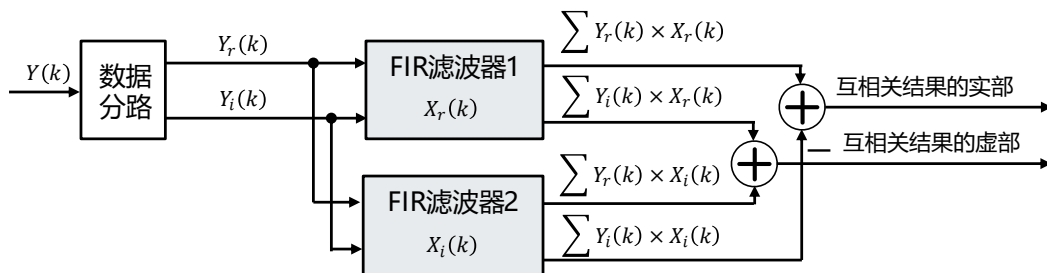


图 3.5 使用 FIR IP 核的互相关的实现结构

根据式 (3.6)，在时域上选取两段信号同时计算能量结果并相加。其中一段信号能量计算的实现结构如图 3.6 所示，其中 $rx_{128}(i)$ 是前级预处理后的数据， $d(i)$ 是输入数据的平方， $E(i)$ 是能量计算的结果，“S” 模块是移位计算模块。首先，为实现流水的能量计算，需要实时计算 128 个数据采样点的能量结果。由于每次能量计算只更新第一个数据 $d(i)$ 和最后一个数据 $d(i+128)$ ，因此可以使用式 (3.14) 进行计算，简化了硬件实现的复杂度。接着，将能量计算的结果乘上一个阈值系数再和互相关结果进行比较，就可以得到同步信号的位置。经过实际测试，这里选择的系数值为 0.21875。硬件设计时使用“S” 模块对二进制数据进行移位即可完成除以 2 的计算，从而完成乘 0.21875 的操作，如式 (3.15) 所示。通过这种拆分计算可以避免使用乘法器，节省了硬件资源开销。

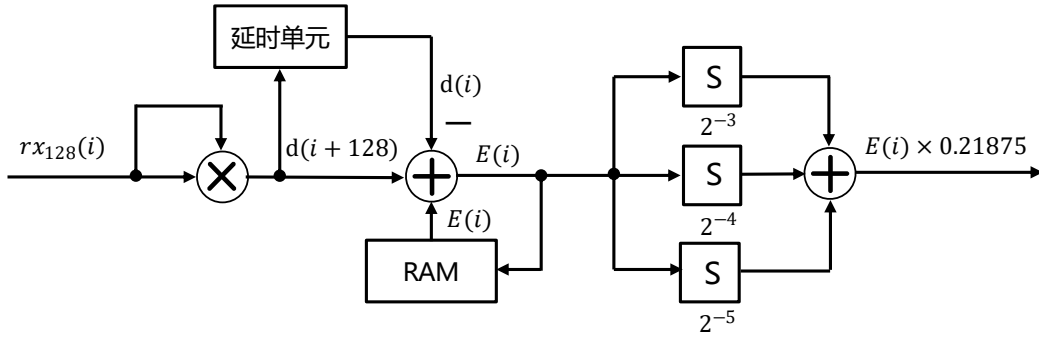


图 3.6 能量计算的实现结构

$$E(i) = E(i) + d(i+128) - d(i) \quad (3.14)$$

$$E(i) \times 0.21875 = E(i) \times (2^{-3} + 2^{-4} + 2^{-5}) \quad (3.15)$$

如图 3.2 所示，互相关计算结果中会出现多个峰值，包括主峰和其他次峰，这些峰值信号的点也满足式 (3.7) 的条件，因此需要进一步计算来确定主峰的位置。传统方法中，通过对所有互相关结果进行遍历来确定最大的峰值，该方法虽然可以确定主峰位置，但是硬件实现过程中需要缓存所有信号，消耗过多的存储资源。本文根据同步信号的特点，即同步信号只位于子帧 0 的前半个子帧长度范围内，选择首次出现峰值位置之后半个子帧长度的信号作为遍历对象，即可确定主峰位置。局部遍历的实现流程如图 3.7 所示，根据 LTE-V2X 无线帧结构，半个子帧长度在时域上是 15360 个采样点，由于互相关计算的输入信号经过了 16 倍降采样，因此缓存 RAM 的深度设置为 960 即可。

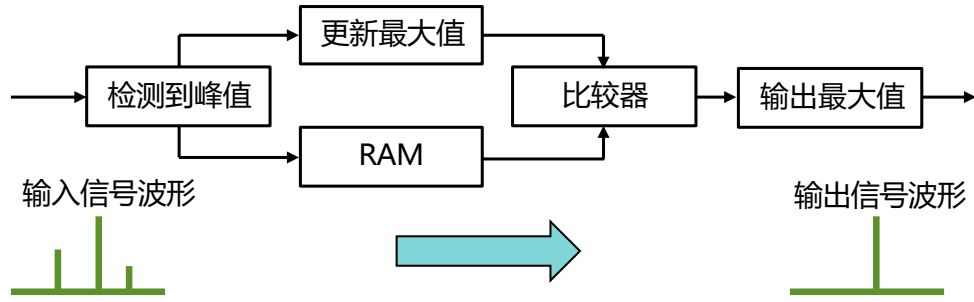


图 3.7 局部遍历的实现流程

3.4.3 频偏估计和补偿模块的设计

1) 频偏估计模块

根据式 (3.9) 和 (3.10) 对信号进行频偏估计，由于信号是连续输入的，因此需要使用 RAM 缓存预处理之后的信号。当完成同步信号的检测后根据同步信号的索引位置从缓存 RAM 中提取 128 个采样点的数据，并将提取的采样数据和存储在本地 ROM 中的 PSS 参考信号进行相乘和累加。然后使用 CORDIC IP 核的反正切模式计算出反正切值，得到频偏估计的值，实现结构如图 3.8 所示。

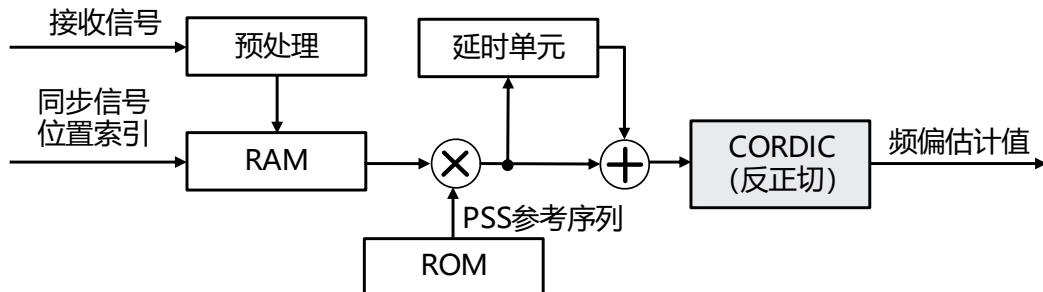


图 3.8 频偏估计的硬件实现结构

2) 频偏补偿模块

图 3.9 是频偏补偿的实现结构，该结构根据频偏估计值对原始接收信号进行频偏补偿。这里使用的 CORDIC IP 核的相位旋转模式，根据 CORDIC IP 核手册，输入相位的范围必须在 $-\pi$ 到 π 之间，因此硬件设计中需要增加一个相位校正模块，根据三角函数的周期性，对输入的相位值进行调整。

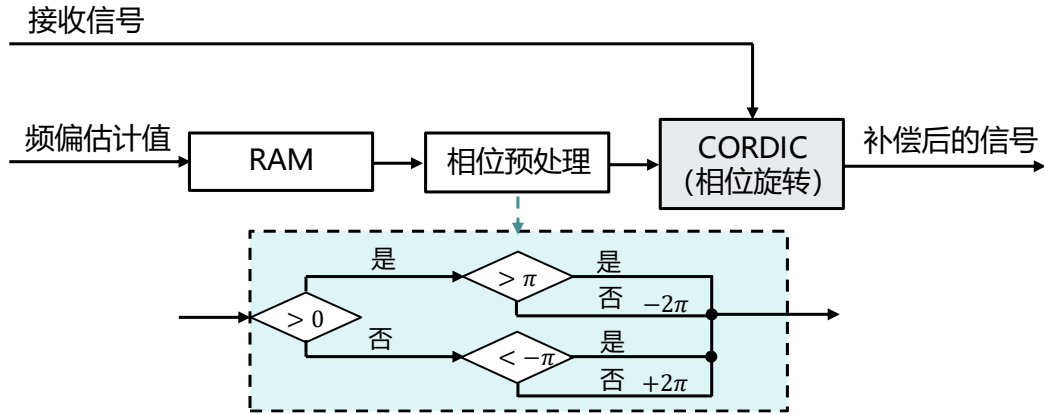


图 3.9 频偏补偿的实现结构

3.4.4 定时标记模块的设计

检测到 PSS 位置之后，根据 LTE-V2X 的帧结构和同步信号的位置就可以推出下一个子帧的帧头位置，定时标记的方法如图 3.10 所示。图 3.10 中，A 点是 PSS 位置，B 点是检测到同步信号的时刻，将这个时刻延迟一段固定的时间间隔就可以标记下一个子帧（子帧 1）的帧头，即 C 点位置。A、B、C 三点相对位置是固定的，通过仿真可以确定 B 点和 C 点之间的固定时间间隔。

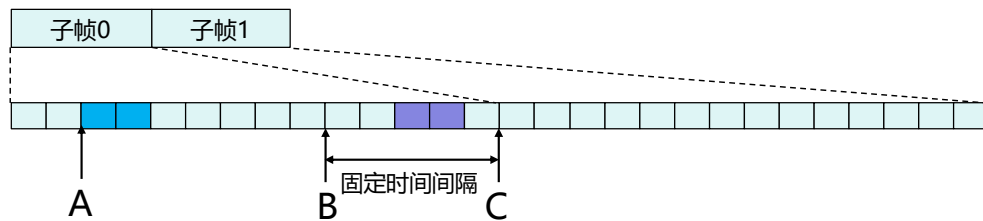


图 3.10 定时标记示意图

根据硬件实现的设计结果编写 RTL 代码，使用 VIVADO 软件进行仿真和综合实现，仿真结果如图 3.11 所示。图 3.11 中，输出的是互相关结果，该结果和 MATLAB 仿真结果保持一致。对照 MATLAB 参考模型对每个模块的输出结果进行验证，确保模块功能正确。在个人电脑的 VIVADO 中进行单模块仿真，然后接入整体 LTE-V2X 工程，并在实验室服务器上使用 VCS 软件进行仿真。仿真验证完成后进行综合，设定约束的时钟频率为 307.2MHz，检测综合后建立时间和保持时间的时间余量是否满足要求，如果不满足则检测对应的时序路径，修改相应的代码。

最终在赛灵思 ZYNQ-7000 FPGA 开发板上进行测试，先提前标记代码中的待测信号，然后使用 VIVADO 软件中的信号波形查看工具实时观察开发板上的信号变化。如图 3.12 所示，其中 1589 是收端解出数据包的数量，发端发送的数据包数量为 1589，测试结果表明接收信号的子帧头被正确标记，收端此次解出数据包的成功率为 100%。

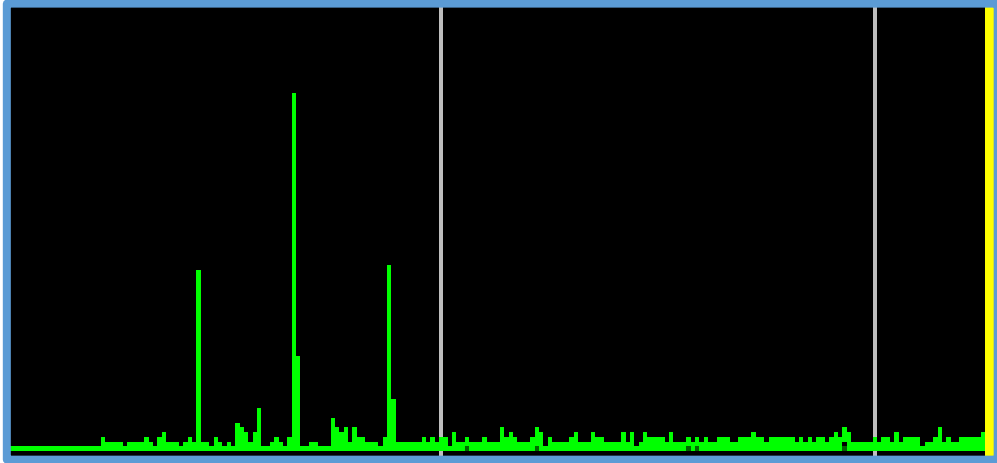


图 3.11 VIVADO 软件中互相关模块的仿真结果

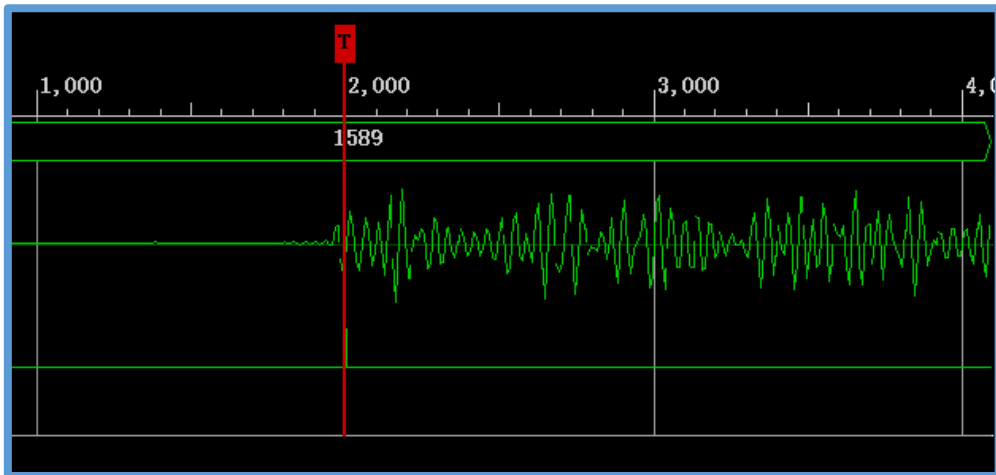


图 3.12 实际上板测试时子帧头标记结果

后续将同步模块放入 LTE-V2X 工程中进行整体调试，并在多种场景下进行了测试，包括单块开发板通过屏蔽线连接发射端口和接收端口的自发自收模式，单块开发板使用两根天线的空口自发自收模式，两块开发板通过屏蔽线之间的双板通信模块和两块开发板使用天线空口双板通信的模式。根据控制信道和共享信道正确解码的数量，可以判断出两块板子之间完成了同步。经过多次长时间的测

试得出，本文设计的同步模块同步建立时间少于 1 秒，同步维持时间大于 24 小时，完成同步后收端的解包成功率为 99.74%。

3.4.5 硬件资源消耗分析

对提出的同步算法进行 FPGA 实现后，具体的硬件资源使用情况如表 3.1 所示。在 3.4.2 节中提到了局部遍历寻找互相关峰值的方法，经过实际测试，如果使用全部遍历的方法，整体工程需要消耗 25 个块随机存储器，使用局部遍历的方法减少了 44%的块存储器资源消耗。

表 3.1 同步模块硬件资源使用情况

资源种类	查找表	片寄存器	块随机存储器	数字信号处理器单元
单元数量	9628	11902	14	52

3.5 本章小结

本章首先介绍了同步技术的应用背景，介绍了几种常用的同步算法。然后，本章提出了一种有利于硬件实现的算法。首先在 MATLAB 中建模仿真完成了硬件设计参考模型的搭建，然后根据参考模型开始进行 FPGA 的设计，介绍了各个模块的具体实现方式，最终连入整个 LTE-V2X 系统进行了实际的测试。设计的同步模块使用了一种改进的互相关算法，同时在筛选峰值时采用局部遍历的方式，相比于全部遍历方式减少了 44%的块存储器资源消耗。测试结果表明，在不同场景下该同步模块都可以完成发射机与接收机的定时同步，支持 307.2MHz 的工作频率，同步建立时间少于 1 秒，维持时间大于 24 小时，完成同步后收端的解包成功率为 99.74%。

第四章 解速率匹配技术的实现

本章介绍了 LTE-V2X 系统接收机中控制信道和共享信道解速率匹配算法的实现。首先,本章介绍了解速率匹配的应用背景和具体的实现算法,然后,根据控制信道和共享信道特点分别实现两种硬件设计方案,最后,利用 FPGA 开发板测试上述方案在 368 种不同信道配置下的实现效果,验证了方案设计的稳定性和通用性。

4.1 引言

随着智能交通的快速发展,车联网技术受到了广泛关注。近年来,需要通过车联网无线通信进行交互的信息量越来越多,如何在保证信息传输可靠性的前提下提高数据吞吐量成为研究者们关注的重点。为满足这种高吞吐量和低能耗的需求^[74],LTE 的协议标准中引入了速率匹配和解速率匹配技术。

速率匹配主要是对发端原始数据进行处理。速率匹配的原理是根据实际信道的传输能力,将要发送的数据分组并重新排列,进而实现打孔和重发这两种操作。其中,打孔指的是在发端将一些数据去除,被舍弃的数据往往是一些冗余数据和在收端可以根据已有信息恢复出的数据,通过打孔操作可以减小数据码块的大小,提高单位时间内有效数据的传输量。重发则是为了提高数据的可靠性,对部分数据进行重复发送,增大有效数据码块,提高传输的稳定性。具体实现时主要根据信道的实际配置选择打孔和重发操作。根据物理层的配置,如果输入数据比输出数据多就进行打孔,反之则进行重发。解速率匹配主要应用于收端。解速率匹配的处理过程和速率匹配相反,根据提前解出的信道配置信息确定原始数据码块的大小,恢复出在发端被舍弃的比特或者去除重复的比特。实际应用过程中,由于速率匹配和解速率匹配算法复杂度较高,往往需要消耗大量的硬件计算资源和能耗,不利于提高系统的整体效率。

LTE-V2X 是在传统 LTE 技术的基础上发展而来,因此,该技术中也引入了速率匹配和解速率匹配的处理过程,主要用于支持不同的传输速率。目前,LTE-V2X 物理层系统支持 368 种不同的信道配置,在不同配置下,无线传输时的码

率不同，具体表现在码块大小和码块数量的差异上。LTE-V2X 发端的控制信道和共享信道需要针对不同配置完成速率匹配，收端的控制信道和共享信道则需要完成相应的解速率匹配操作。针对以上调研和系统需求，本文提出并实现了两种不同的解速率匹配硬件实现结构，分别针对 LTE-V2X 收端控制信道和共享信道的实际应用场景进行优化，在满足解速率匹配功能正确性的前提下有效降低了硬件实现的复杂度，降低了硬件资源和能耗的开销。

4.2 解速率匹配算法的研究与建模

4.2.1 解速率匹配整体实现结构

解速率匹配是速率匹配算法的逆过程，整体实现流程如图 4.1 所示，主要包括四个部分，分别为构建虚拟循环缓冲器、比特重排和填充、比特分流、子码块解交织器。图 4.1 中首先构建一个虚拟循环缓冲器，输入的比特 e_k 在虚拟循环缓冲器中进行重排，根据具体信道配置填充或者舍弃比特得到 w_k ，然后进入比特分流模块，共分出三路不同的比特数据 V_k^0 、 V_k^1 、 V_k^2 ，分别通过子码块解交织器进行解交织，最终恢复出在发端进行速率匹配前的原始数据 D_k^0 、 D_k^1 、 D_k^2 ，便于后续模块进行解码和校验等处理。

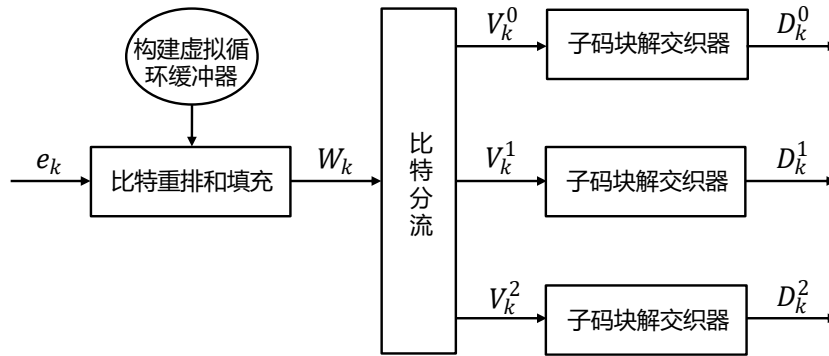


图 4.1 解速率匹配算法的整体实现流程

4.2.2 虚拟循环缓冲器的实现

虚拟循环缓冲器是解速率匹配的核心模块。建立虚拟循环缓冲器之前，首先要根据信道配置的控制信息确定循环缓冲器大小，其中包括每个比特流的个数和

矩阵的行数与列数，然后开始创建虚拟循环缓冲器，具体实现步骤如下：

1) 根据 LTE-V2X 物理层协议设置矩阵的列数 C 固定为 32，外部控制信号给出当前输出要求的码块总大小为 F_n ，根据式 (4.1)、(4.2) 和 (4.3) 即可计算出每个比特流的大小 N_t 、矩阵的行数 R 和每个子块交织器的大小 K_{rect} ，其中式 (4.2) 中的半括号表示向上取整。一个循环缓冲器包括三个大小相同的子块，分别为标记为 S、P1 和 P2，这三个子块的位置如图 4.2 所示，S 位于第一段，P1 和 P2 交错存在 S 段数据之后。

$$N_t = \frac{F_n}{3} \quad (4.1)$$

$$R = \left\lceil \frac{N_t}{C} \right\rceil, C = 32 \quad (4.2)$$

$$K_{rect} = R \cdot C \quad (4.3)$$

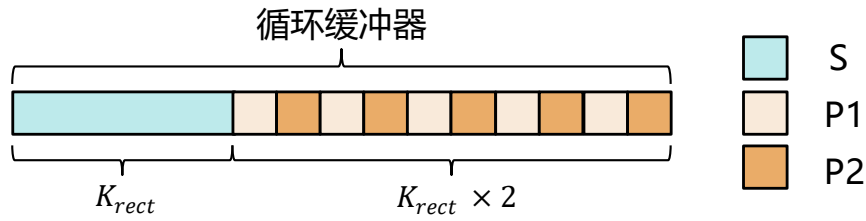


图 4.2 循环缓冲器的结构

2) 每个子块都需要填充虚比特，其中，虚比特是一种数据传输的计量单位，用来在速率匹配和解速率匹配中模拟一个数据比特的大小。在确定每个子块的大小之后根据式 (4.4) 计算需要填充的虚比特数量，并在循环缓冲器中每个子块的起始位置处按行依次进行填充，其中 D_k 表示虚比特的数量。在 MATLAB 中进行建模仿真时，虚比特用 Inf 表示，空的位置用 0 表示。

$$D_k = K_{rect} - N_t \quad (4.4)$$

3) 在每个子块的起始位置填充完虚比特之后，对每个子块进行列间交织处理。在表 4.1 中，列数和列变换映射一一对应，表示将列数中对应位置的列移动到列变换映射中对应位置的列上。例如，对于列数和列变换映射中的第一个数，表示将第一列移动到第十七列，第二个数表示将第二列移动到第一列，以此类推，完成列变换。完成列变换之后将循环缓冲器每个子块的比特按列从上到下依次排

列，得到交织后带有虚比特的子块，最后按照图 4.2 的结构将三个子块组合。

表 4.1 列变换映射规则

列数	列变换映射
<1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18, 19,20,21,22,23,24,25,26,27,28,29,30,31,32>	<17,1,25,9,21,5,29,13,19,3,27,11,23,7,31,15, 18,2,26,10,22,6,30,14,20,4,28,12,24,8,16>

以单个子块 S 为例解释上述过程，如图 4.3 所示。假设 S 是一个 6×4 大小的子块矩阵（实际子块矩阵的列数固定为 32，这里为了便于画示意图设列数为 6），每个子块比特流大小为 24，子块交织器大小为 20，根据式（4.4）即可算出虚比特数量为 4。在子块开头按行从左到右进行填充，然后根据特定规则进行列变换，接着按列从上到下进行排列，得到交织后的循环缓冲器中单个子块 S。

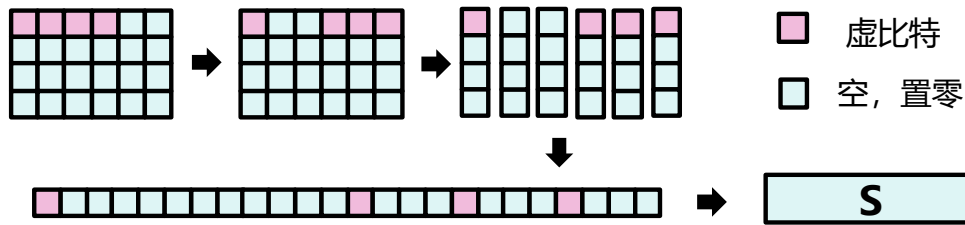


图 4.3 循环缓冲器中单个子块的构建

4.2.3 解速率匹配其他步骤的实现

1) 比特重排和填充

得到交织后带有虚比特的循环缓冲器后，按行逐一填入数据比特，遇到有虚比特的位置就跳过并存到下一个位置。如果数据长度少于循环缓冲器的大小，空余位置置零。若数据长度大于循环缓冲器的大小，则将多余的数据填充到循环缓冲器的开头位置与之前填充过的数据进行相加。每次填满就循环回到开头位置进行填充，直到填充完所有的数据比特。

2) 比特分流

输入的数据比特填充完成后按照图 4.2 的结构将所有数据分成 S、P1 和 P2 三部分，还原出每个子块。

3) 子块解交织器

分出三个带有数据比特的子块后对每个子块按照相同的规则进行解交织。首

先根据表 4.1 的列变换规则对顺序输入的地址进行交织变换，变换过程和图 4.3 相似，但是变换的对象不是数据而是对应数据的地址，如图 4.4 所示。这里使用 6×3 的子块矩阵并标记 1-18 的地址。在计算出解交织的地址后，利用这些地址读取每个子块的数据，以完成数据的解交织处理。接着，进行比特重排并移除虚拟比特，以恢复发端速率匹配前的原始数据，完成解速率匹配全过程。

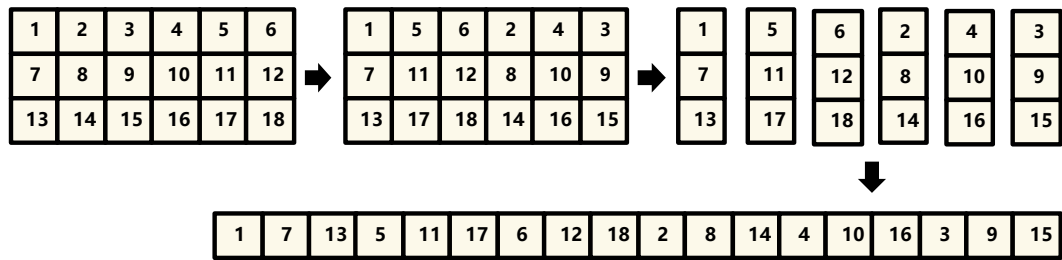


图 4.4 解交织地址计算示意图

4.3 控制信道解速率匹配模块的实现

4.3.1 总体设计框图

图 4.5 是控制信道解速率匹配的总实现框图，按主要功能可以划分为四个部分，分别是状态机控制模块、输入控制模块、RAM 模块和输出控制模块。

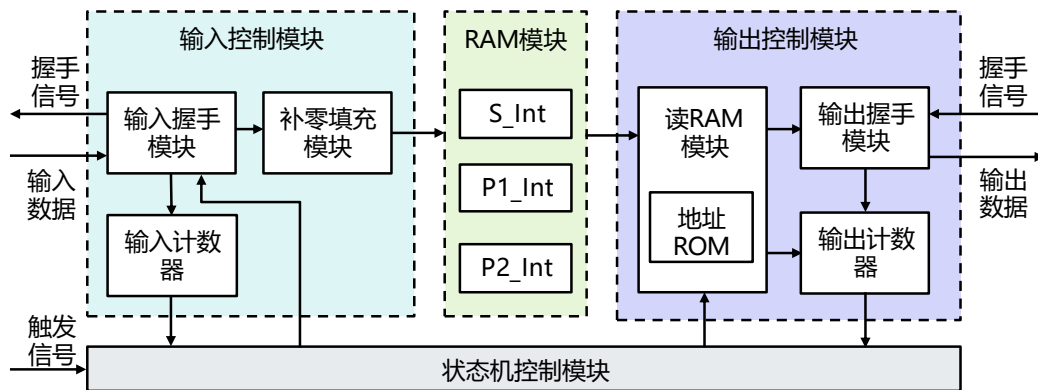


图 4.5 控制信道解速率匹配总体实现框图

状态机模块是整个解速率匹配系统的核心模块，负责接收外部输入的信道配置信息，根据其他模块计数器的计数结果进行状态的切换，产生发送给其他模块的控制信号。输入控制模块根据握手协议，接收前级模块的数据信号和写 RAM 的地址信号。RAM 模块负责所有码块的存储，通过改变读写地址完成比特重排。

输出控制模块控制读 RAM 的地址信号以及和后级模块的握手信号。

在目前的 LTE-V2X 物理层系统中, 控制信道的配置参数是固定不变的, 对于 368 种不同的信道配置, 每次传输码块的大小都相同, 这意味着解速率匹配过程中子块矩阵的大小也都是确定的, 其中包括行数和列数。因此, 控制信道中输入数据和输出数据的比特变换规则和映射关系都是固定的。本文利用了上述条件, 预先通过 MATLAB 计算出输入比特和输出比特的映射关系, 将计算结果作为地址信息存储在 FPGA 的 ROM 中, 使用查表的方法完成解速率匹配过程中的比特重排过程, 避免了交织和解交织的计算, 有效降低解速率匹配的计算复杂度和硬件资源消耗。

4.3.2 控制信道解速率匹配各个模块的设计

1) 状态机控制模块

状态机可以将复杂的任务划分为一个个子块, 便于实现复杂的控制功能, 因此经常被用于数字设计中。本文使用标准的三段式状态机进行设计, 分别是空闲状态 (IDLE)、RAM 存储器写数据状态 (S1)、RAM 存储器读数据状态 (S2), 如图 4.6 所示。

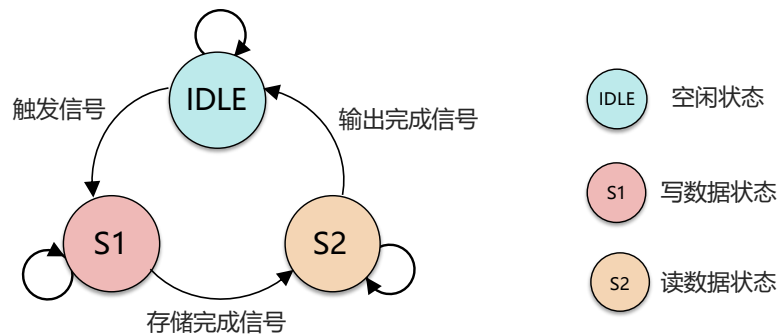


图 4.6 状态机跳转情况

IDLE: 在该状态下整体模块处于待机模式, 控制 RAM 存储器的读写信号都置零, 当外部的控制信息给出触发信号后, 解速率匹配模块开始工作, 跳转到 S1 状态。

S1: 在 S1 状态下, 系统开始接收前级模块的数据信息并控制写地址的信号将数据存储到 RAM 中。根据 LTE-V2X 物理层配置, 输入数据会扩充到 432 个

比特，将 432 个数据比特平均分成三段，每段 144 个比特，将三段数据进行叠加最终得到总长为 432 的数据子块。模块产生 0-431 的写地址信号，随后将数据依次存入 RAM 存储器中，完成输入数据的存储后产生存储完成信号，跳转到 S2 状态。

S2：该状态主要控制数据的读取，将 MATLAB 预先计算的映射关系作为读地址信号存储在 ROM 中，进入该状态后从 ROM 中读出比特映射信息并将读出的数据作为读地址，读取 RAM 中的数据比特并整理后输出，完成整个解速率匹配过程，并回到空闲状态等待下一次传输。

2) 输入控制模块

输入控制模块主要由三部分组成，分别是输入握手模块、补零填充模块和输入计数器模块。输入握手模块负责和前级模块握手，当模块处于空闲状态时，前级握手信号拉高，进入可以接收数据的状态。补零填充模块根据输入数据的计数器结果对数据进行补零填充，填满 432 个比特。此外，计数器的结果也将传递给状态机，触发状态机的状态跳转。

3) RAM 存储模块

存储模块利用 VIVADO 软件提供的 Block RAM IP 核来存储用 S_Int、P1_Int 和 P2_Int 表示的三段数据。通过向 RAM 发送读写地址信号和读写使能信号，可以读取或写入所需地址的数据。输入控制模块负责将数据写入 RAM，而输出控制模块负责从 RAM 读取数据。根据当前的 LTE-V2X 配置，有三个数据段 v0、v1 和 v2，每段长度为 48 个点，总长度为 144 个点。前级输入控制模块处理后的数据有 432 个，为提高信息的利用率和后续解码性能，这里选择使用 RAM 在存储数据的同时完成三段数据的叠加，每段数据长度均为 144 个比特。输入数据的位宽为 8 比特，三个位宽为 8 比特的数据段叠加，需要进行两次加法运算，为防止加法计算时发生数据溢出，此处需要将结果扩充 2 比特。综上考虑，RAM 配置中宽度为 10 比特，深度为 144，大小为 1440 比特。设输入数据为 $x(n)$ ，经过叠加，最终存储在 RAM 中的数据为 $y(t)$ ，两者之间的关系如式 (4.5) 所示。

$$y(t) = \sum_{M=1}^3 x[n + (M-1) \times 48] \quad (4.5)$$

式中， $t \in [1, 432]$ ， $t \in \mathbb{Z}$ ， $n \in [1, 144]$ ， $n \in \mathbb{Z}$ 。M 是总共叠加的段数。

地址寄存器在时钟的驱动下会根据输入数据的有效信号计数加 1，同时，数据被依次存储在 RAM 中。后续模块将采用查表方法来读取 S_Int、P1_Int 和 P2_Int 这三路数据。查表时可以根据 MATLAB 计算出的映射关系灵活选择这三路数据，无需用三个 RAM 模块分别存储不同数据段的数据，因此，该过程只需一个 RAM 即可实现三路数据的存储。在第一轮数据存储完成之后，依次读取数据并与新输入的数据相加，然后将第一次叠加后的数据再次顺序存入 RAM 中，覆盖原来的数据。重复两次叠加过程，即可完成三段数据的叠加与存储。

4) 输出控制模块

输出控制模块负责从 RAM 中读取数据，整理排列后输出，主要包括三个模块，分别是读 RAM 模块、输出握手模块和输出计数器模块。读 RAM 模块负责产生读取 RAM 的读地址，其中有一块 ROM，该 ROM 预先存储了 MATLAB 计算出的输入输出比特位置映射关系，通过从 ROM 中读取数据得到读 RAM 的读地址。根据读地址从 RAM 中读取数据，即可完成解速率匹配过程中的比特重排。输出握手模块负责和后级的信道解码模块进行握手，当解速率匹配模块存储完数据且后级模块准备好后产生触发信号，从 RAM 存储模块中读取数据并输出。输出计数器模块负责对输出的数据进行计数，根据计数结果判断状态机是否完成当前状态，完成后跳转回空闲状态。

具体的比特变换过程如图 4.7 所示，S_Int 地址、P1_Int 地址和 P2_Int 地址是三段数据比特的映射地址信息，将三段数据比特的地址组合后得到读 RAM 模块的读地址，读出 RAM 中存储的数据即可依次读出图 4.2 中三段数据，完成控制信道的解速率匹配过程。

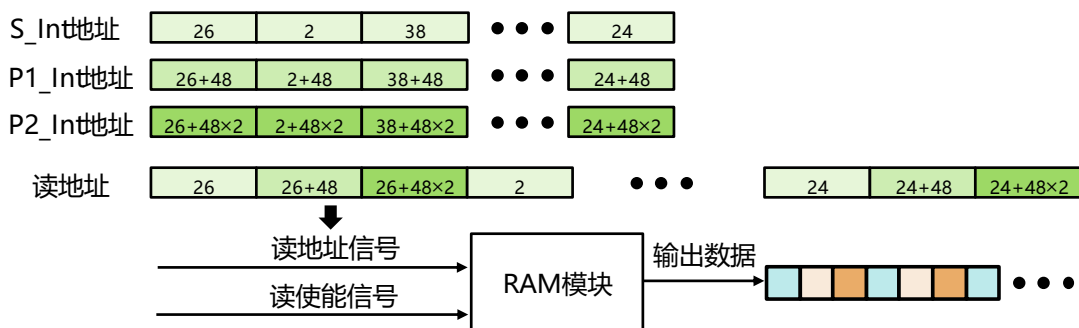


图 4.7 具体的比特变换过程

4.3.3 测试结果

按照设计的结构编写 RTL 代码，并使用 MATLAB 产生输入的测试向量，在 MATLAB 和 VIVADO 软件中分别进行仿真，并比对最终输出的数据，以此验证功能的正确性。图 4.8 是 VIVADO 软件的仿真结果，其中 state 表示状态机的状态，0、1、2 分别对应了 IDLE、S1 和 S2 三个状态，data_descr 为解速率匹配模块的输入数据，data_viterbi_o 为解速率匹配的输出数据。进行多组测试向量的仿真验证后在 VIVADO 软件中对工程添加时序约束进行综合，综合结果表明模块的工作频率可以满足 307.2MHz 的系统时钟频率要求。表 4.2 是综合后的资源使用情况，结果显示，所设计的控制信道解速率匹配模块只用 213 个查找表、241 个片寄存器和 1 个块存储器资源就实现了控制信道的解速率匹配算法，同时由于避免了交织的具体计算，该过程未用到乘法器所需的数字信号处理器（Digital Signal Processor, DSP）资源。

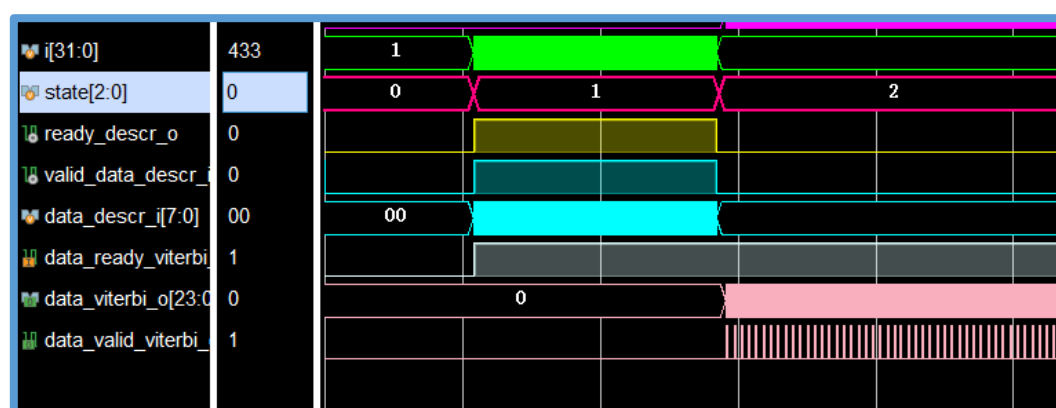


图 4.8 控制信道解速率匹配模块 VIVADO 软件仿真结果

表 4.2 控制信道解速率匹配总体硬件资源使用情况

资源种类	查找表	片寄存器	块随机存储器	数字信号处理器单元
单元数量	213	241	1	0

综合完成后将设计的控制信道解速率匹配模块接入整个 LTE-V2X 工程，对总体工程进行仿真、综合和上板测试。控制信道解速率匹配模块位于收端比特级处理模块的控制信道处理部分，如图 4.9 所示。

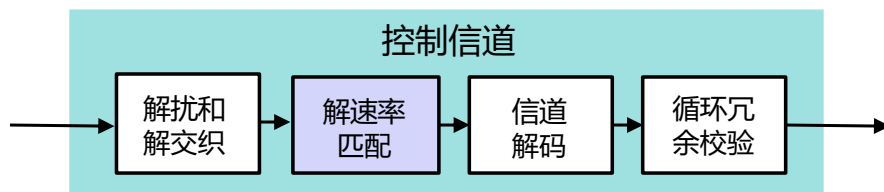


图 4.9 控制信道解速率匹配模块的位置

完成整体仿真后进行多次上板测试，根据计数器结果判断解出的数据包数量。在上位机中配置发送 1599 个数据包，上板测试的计数器信号显示解出 1599，说明控制信道的解码成功率为 100%，验证了解速率匹配模块的功能正确性。

4.4 共享信道解速率匹配模块的实现

4.4.1 总体设计框图

共享信道解速率匹配模块的整体设计框图如图 4.10 所示，包括状态机控制模块、输入控制模块、RAM 模块和输出控制模块。其中输入控制模块包括触发开始模块、参数计算模块、输入握手模块、虚比特填充模块和写数据控制模块。RAM 模块包括三个分开的 RAM，分别标记为 S_Int、P1_Int 和 P2_Int。输出控制模块包括读数据控制模块、输出握手模块和复位模块。

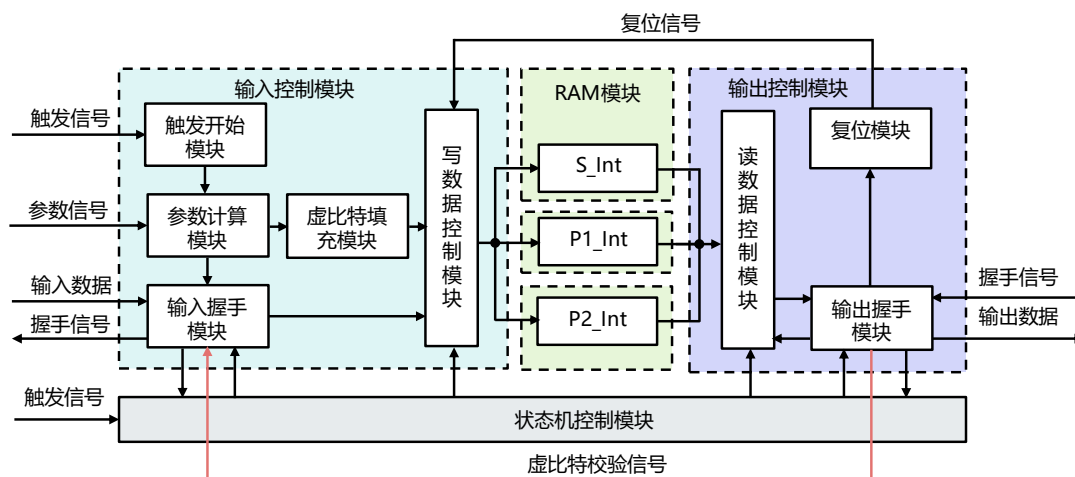


图 4.10 共享信道解速率匹配模块总体设计框图

在 LTE-V2X 收端比特级处理的共享信道中, 不同信道配置下解速率匹配的参数配置也是不同的, 接收机需要能够处理不同大小、不同数量的码块, 因此需要设计合理的交织和解交织模块完成比特重排。和控制信道的解速率匹配结构不

同，共享信道解速率匹配无法使用固定的查表方式，需要根据控制信道每一次传输的参数推导出比特重排的规则，因此总体结构设计比控制信道的解速率匹配模块更为复杂。本文提出了一种使用虚比特校验的共享信道解速率匹配模块设计。首先，计算虚比特数量和位置并将其存入 RAM 模块，缓存数据时读出当前地址的数据并判断是否为虚比特，有虚比特则拉低向前级请求数据的握手信号，使前级模块的数据等待一拍存入下一个地址的位置。通过这样的校验方式简化了硬件设计中判断逻辑的复杂度，提高了解速率匹配模块在不同码率配置下的适配性。

4.4.2 共享信道解速率匹配各个模块的设计

1) 状态机控制模块

共享信道的解速率匹配模块需要按照本章 4.2 节的算法进行实现，使用状态机来控制对码块中不同数据段的处理，如图 4.11 所示。该过程共分包含五个状态，分别是空闲状态（IDLE）、虚比特填充状态（S1）、存 S_Int 状态（S2）、存 P1_Int 和 P2_Int 状态（S3）、输出数据状态（S4）。

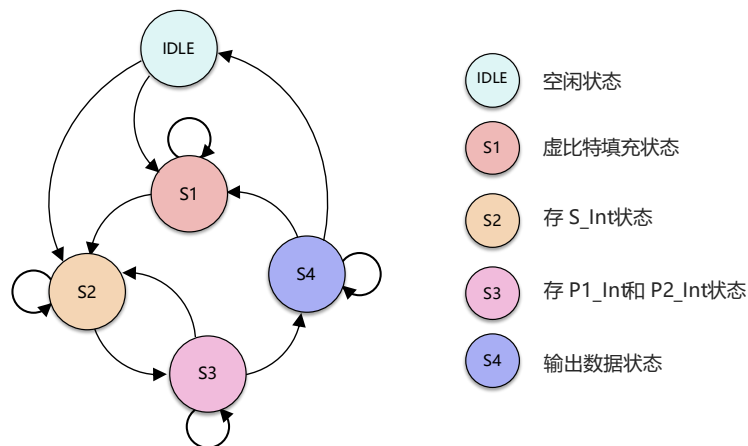


图 4.11 共享信道解速率匹配状态机示意图

IDLE: 该状态和控制信道解速率匹配模块类似，也表示待机状态，当前级触发信号到来后，进入 S1 状态。

S1: 在 S1 状态下，构建虚拟循环缓冲器并对其填充虚比特。首先根据控制信道参数计算出虚比特的数量和位置，然后添加虚比特标志位填充在 S 部分数据段的 RAM 中。根据解速率匹配的算法，三个数据段 S、P1 和 P2 中虚比特的数量和位置都是相同的，因此为了校验虚比特，只需要在 S 数据段对应的 RAM

中存入虚比特，当存 P1 和 P2 的时候也使用 S 部分的缓冲器进行虚比特校验即可。当虚比特填充完成后进入 S2 状态，等待数据的到来。

S2: 从当前状态开始向虚拟循环缓冲器中存入数据比特。首先存的是图 4.2 中 S 部分的数据段，存的时候提前三个系统时钟周期预先读取出 S 部分数据段 RAM 对应地址的数据，判断读出的数据是虚比特还是空（用 0 表示），如果为虚比特，则存到下一个位置，否则就存到当前的地址。依次存完 S 部分的数据后，进入下一个状态 S3，开始存 P1 和 P2 部分的数据。

S3: S3 状态负责存 P1 和 P2 部分的数据比特。在当前状态下从 S 部分的 RAM 中读出数据判断当前地址是否有虚比特，然后根据判断结果交替向 P1 和 P2 部分的 RAM 中存入数据。存满 P1 和 P2 的 RAM 后判断数据是否全部存完，如果没有，就继续跳转到 S2 状态开始循环。继续将数据存入 S 部分的数据段并和对应位置的数据叠加，直到存完所有的数据后进入下一个状态 S4。

S4: 在当前状态下，数据比特已经全部存入循环缓冲器。当后级模块发出数据请求的信号后，根据控制信号参数计算出解交织的地址，用计算出的地址读取 RAM 中的数据比特，在读取数据的同时完成解交织。根据后级模块的输入格式要求，将数据进行整理后输出，完成共享信道的解速率匹配过程。

2) 参数计算模块

参数计算模块首先接收并缓存控制信道解出的控制信息，包括码块数量、输出码块的大小、输入数据比特的数量。接着，根据式 (4.2) 和 (4.3) 计算出循环缓冲区中每个数据段的矩阵行数和虚比特数量。

3) 虚比特填充模块

虚比特填充模块主要负责完成循环缓冲器中的虚比特填充，模块只对 S 部分数据段对应的 RAM 进行填充。首先根据参数计算模块计算出的虚比特数量生成相应的递增数列，然后对这些数进行交织处理，计算出交织后的地址，在 S1 状态中向 RAM 中对应地址存入 10000000000。存入的数据中，最高位是虚比特标志位，1 表示有虚比特，0 则表示当前位置为空。

在虚比特填充模块中有一个计算交织地址的模块。交织指的是将多个数据重新排列。根据解速率匹配算法，交织地址的计算流程如图 4.12 所示。首先根据计算出的虚比特数量 D_k 产生 1、2、3... D_k 的顺序地址，接着根据生成循环缓冲

器的大小计算当前地址在矩阵中位于第几行第几列，然后从 ROM 中读取预先定义的列变换规则进行列变换，得到新的列。最后根据式 (4.6) 进行计算，其中 R 是预先计算好的循环缓冲器矩阵行数， $X_{address}$ 是虚比特的顺序地址， R_i 和 C_i 是填充虚比特在循环缓冲器矩阵中行和列的位置， C_o 是对 C_i 进行列变换后的列数， $Y_{address}$ 是虚比特经过交织计算后的新地址。

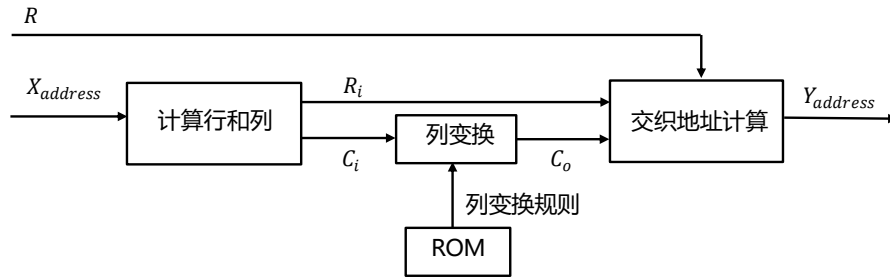


图 4.12 交织地址计算流程

$$Y_{address} = C_o \times R + R_i \quad (4.6)$$

虚比特交织计算在 VIVADO 软件中的仿真结果如图 4.13 所示，图中红框表示的是在本次传输中共有 28 个虚比特，图中绿色部分是虚比特的原始地址，蓝色部分是虚比特经过交织计算后的新地址。

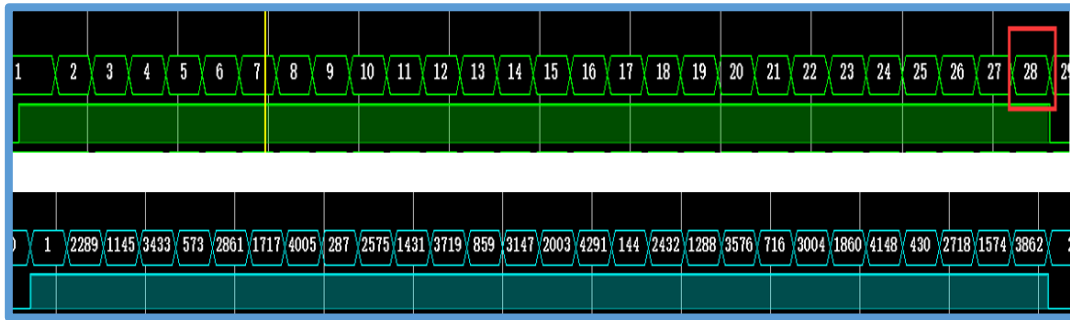


图 4.13 虚比特交织地址计算仿真结果（上面是输入地址，下面是交织计算后的输出地址）

4) 写数据控制模块

写数据控制模块负责向标记好虚比特的 RAM 中存入数据比特。在存数据比特之前提前读出数据判断当前位置是否有虚比特，即检测输出数据的最高位是否为 1。如果检测到虚比特标志位则反馈给输入信号握手模块，将向前级请求数据的信号拉低，从而使输入数据等待一个时钟周期存入到下一个没有虚比特的空位置。由于前级模块中用先进先出缓冲器（First In First Out, FIFO）存储数据，因

此可以通过拉低数据请求信号的方式让前级信号暂停输入,等待跳过虚比特的位置继续输入。通过这种方式降低了解速率匹配模块中地址计算的复杂度,提高了针对不同虚比特位置情况的适配性。根据解速率匹配算法,首先参考 S_Int RAM 中虚比特位置完成 S 段数据的存储,接着继续按照 S_Int RAM 中的虚比特位置存储 P1 和 P2 的数据。虽然此时 S_Int RAM 中已经存储了数据比特,但是本文设计的存储位宽为 11 位,最高位是虚比特标志位,低 10 位是数据位,因此 S_Int RAM 中存储的数据比特不会覆盖最高位的虚比特校验信号。按照图 4.2 的结构,存完 S 段数据后需要交替存储 P1 和 P2 的数据比特,实际硬件实现时使用了奇偶判断的方法,根据计数器结果,若为奇数,存入 P1_Int RAM,否则存入 P2_Int RAM。

5) 读数据模块和输出控制模块

读数据模块和输出控制模块负责读出循环缓冲器中的数据。首先计算出解交织的地址,使用和交织相同的算法计算出比特变换的映射关系。接着,生成输出数据比特的序号,依次标记为 1、2、3、4...K, K 是当前配置的输出码块大小。然后,根据式 (4.6) 计算比特重排后的对应地址。用计算出的地址作为读地址分别从三个数据段的 RAM 存储器中读取数据,完成解交织变换。最终将三路数据整理后输出,完成整个共享信道的解速率匹配过程。

4.4.3 测试结果

根据设计的硬件结构编写 RTL 代码,设定多种不同的配置情况进行仿真测试并和 MATLAB 参考模型的结果进行比对验证。完成功能仿真后将该模块接入整个 LTE-V2X 系统中,图 4.14 是设计的解速率模块在系统工程中的位置。对工程进行总体仿真、综合和静态时序分析并优化时序和资源,表 4.3 是设计的解速率匹配模块总体硬件资源使用情况,各项资源使用的分配情况都较为合理。

表 4.3 共享解速率匹配模块总体硬件资源使用情况

资源种类	查找表	片寄存器	块随机存储器	数字信号处理器单元
单元数量	874	840	8.5	2

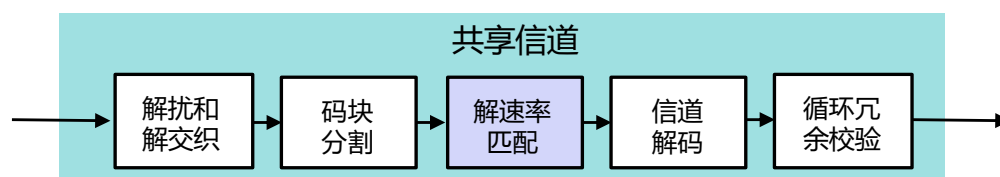


图 4.14 共享信道解速率匹配模块的位置

图 4.15 是 VIVADO 软件的仿真结果，pssch_start_i 信号是共享信道解速率匹配模块的触发模块，ready_descr_o 和 valdi_data_descr_i 分别是输入数据的请求和数据有效信号，data_descr_i 是输入的数据信号，tbs0_rm_i 是当前配置下输出码块的大小，E0_i 是输入数据的总大小，cb_num_i 是码块总个数，state 是状态机信号，图中可以看到之前描述的五五个状态，最终输出的信号为 data_deturbo_o。

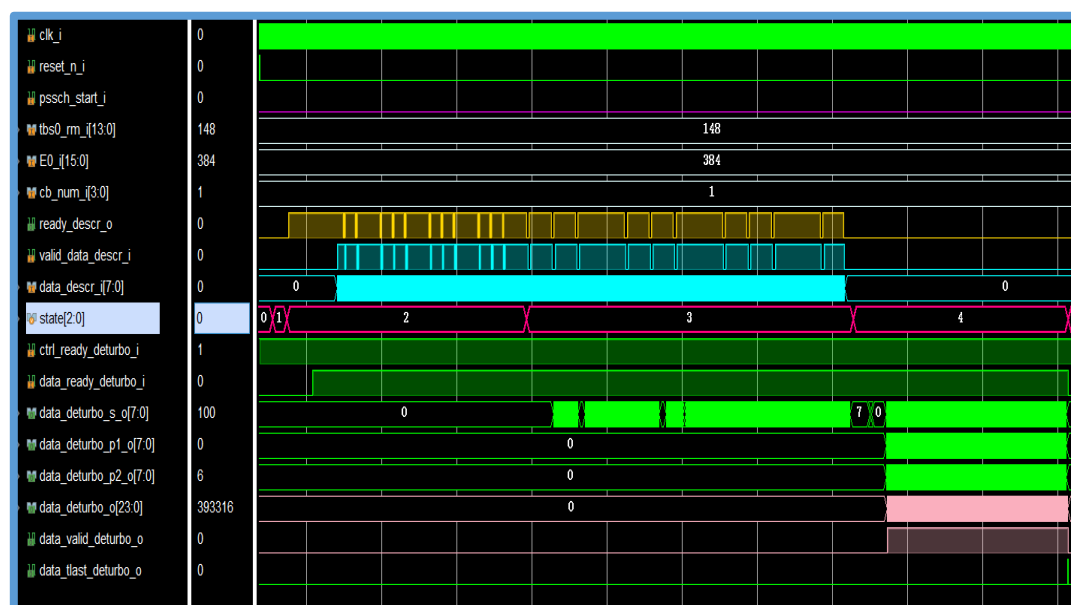


图 4.15 VIVADO 软件仿真结果

最后，使用 VIVADO 软件生成比特流文件并烧录到 FPGA 开发板上进行实际测试。在 FPGA 开发板测试过程中，通过个人电脑上的 SDK 软件更改配置参数，进行不同信道配置下的测试，观察通过 FPGA 中表示控制信道和共享信道解码数量的信号验证解速率匹配的功能是否正确。如图 4.16 所示，其中上下两个信号的值分别代表共享信道和控制信道解出的数据包数量（此次传输发送的数据包为 1599，由于其他模块的误差等外界因素，解出 1595 以上即可说明该模块功能正确）。实际测试结果表明，设计的解速率匹配模块支持最终系统要求的 368

种不同信道配置。

在早期的 LTE-V2X 工程中也设计了共享信道解速率匹配模块, 本文在之前工作的基础上重新设计了新的硬件实现架构, 两种硬件实现架构的对比如表 4.4 所示, 主要体现在综合频率、处理速率和支持配置数量的提升。表中处理周期指的是从接收数据到解速率匹配完成并输出所有数据所用的时钟周期数量, 可以得出新设计的解速率匹配模块计算延时比早期工程中模块的计算延时降低了一倍。降低的原因在于之前工作设计的解速率匹配模块需要等待数据全部输入后才能开始比特重排, 而本文设计的解速率匹配模块在数据输入的同时即可开始比特重排。此外, 在早期工程中, 为了解决码块数量过多无法在规定时间内处理完成的问题, 在共享信道中使用了两个相同的解速率匹配模块进行交替并行处理。本文设计的解速率匹配模块提高了处理速率, 只需使用一个解速率匹配模块即可满足系统时序的要求, 相比于之前的设计有效减少了硬件资源消耗。

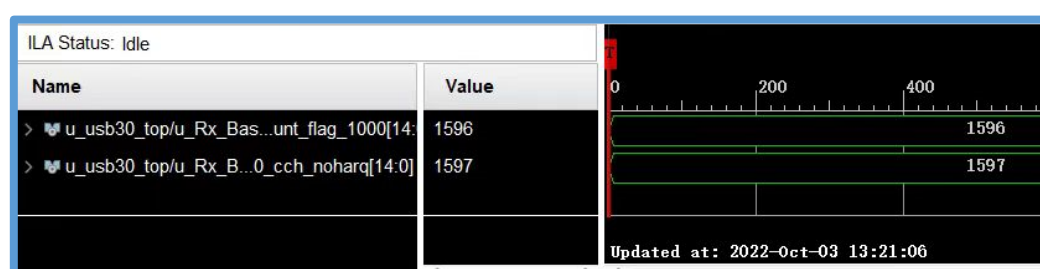


图 4.16 在 FPGA 开发板上的解码结果

表 4.4 本文设计的共享信道解速率匹配模块和之前工作对比（传输码块大小为 6192）

种类	可综合频率	处理周期	支持信道配置
早期工程中的解速率匹配模块	125MHz	24768	98 种
本文设计的解速率匹配模块	400MHz	12384	368 种

4.5 本章小结

本章针对 LTE-V2X 收端的解速率匹配技术提出了两种不同的硬件设计。对于参数配置固定的控制信道, 本章设计了一种类似查表的方式实现解速率匹配模块。对于参数配置不固定的共享信道, 本章通过虚比特校验的方法构建循环缓冲器, 以此实现解速率模块。本章编写 MATLAB 代码搭建参考模型, 根据参考模

型进行 FPGA 设计并优化，最终将设计的模块整合进 LTE-V2X 系统中进行了大规模测试，根据控制信道和共享信道解包结果判断功能是否正确。本章设计的两个模块均支持 307.2MHz 工作频率，处理速度相比早期工作提升了一倍，并且能够完成 368 种信道配置的解速率匹配处理，实际测试结果验证了设计的可靠性。

第五章 CORDIC 和 FFT 计算模块的设计

本章主要介绍了 CORDIC 和 FFT/IFFT 计算模块的设计。首先介绍了 CORDIC 和 FFT/IFFT 算法的原理和在 LTE-V2X 系统中的应用, 然后根据算法原理从基本的数学公式出发, 进行公式推导化简, 并对算法进行建模仿真, 接着确定硬件实现的具体结构, 编写 RTL 代码进行仿真验证, 最终在 FPGA 开发板上完成了长时间的实际测试。

5.1 引言

在无线通信系统中, CORDIC 和 FFT/IFFT 算法具有广泛的应用价值^[75-78]。在本文研究的 LTE-V2X 系统中, 发射机与接收机都需要使用这几种算法。对于 CORDIC 算法, 在 LTE-V2X 的发送端, DMRS 序列生成器需要利用 CORDIC 模块的正余弦计算模式来计算相位角度的正弦值和余弦值。同时, 为了降低频率偏移和噪声对信号传输质量的影响, 发端还需要使用 CORDIC 模块的相位旋转功能来实现发射信号的预补偿。而在收端, 相位旋转功能则被用于对信号进行频偏补偿。对于 FFT 算法, 其主要作用是完成信号在时域和频域之间的转换, 是 SC-FDMA 的关键技术之一^[79, 80]。在 LTE-V2X 的发端需要使用 IFFT 算法将信号从频域转换到时域, 在收端则需要使用 FFT 算法将信号从时域恢复到频域。

早期 LTE-V2X 系统工程中相应计算模块在硬件实现时使用赛灵思公司提供的 CORDIC、FFT 和 IFFT IP 核, 这几种 IP 核虽然可以根据用户需求自由配置, 但没有针对 LTE-V2X 工程进行定制化的设计。为满足 LTE-V2X 系统的实际需求, 本章对正余弦计算模式和相位旋转模式的 CORDIC 模块、2048 点的 FFT 和 IFFT 模块进行自主设计和开发。在确保功能完备的基础上, 针对 LTE-V2X 的特定场景进行了定制化设计, 并在资源和计算复杂度方面进行了相应的优化。

5.2 CORDIC 计算模块的设计

5.2.1 CORDIC 算法原理

CORDIC 算法的核心思想是向量旋转，通过一系列旋转操作，将初始向量逐步旋转至目标角度，从而计算出所需的三角函数值。这一系列旋转操作通过有限次的加法、减法、位移等基本运算进行实现，因此在硬件实现上具有较低的复杂度。如图 5.1 所示，横轴表示数据的实部，纵轴表示虚部，假设输入数据的坐标 Q_a 的位置为 (X_a, Y_a) ，那么数据的值为 $X_a + Y_a j$ ，给定相位角 θ ， Q_a 逆时针旋转 θ 后得到 Q_b ，坐标为 (X_b, Y_b) 。

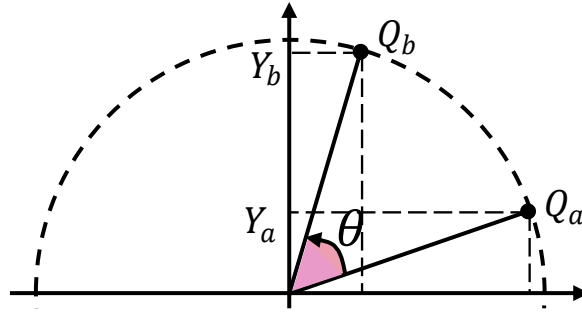


图 5.1 CORDIC 旋转示意图

根据三角函数关系的数学推导， Q_a 与 Q_b 坐标关系满足式 (5.1)。将两个公式进行变换得到式 (5.2)。

$$\begin{cases} X_b = X_a \cdot \cos \theta - Y_a \cdot \sin \theta \\ Y_b = Y_a \cdot \cos \theta + X_a \cdot \sin \theta \end{cases} \quad (5.1)$$

$$\begin{cases} X_b = \cos \theta \cdot (X_a - Y_a \cdot \tan \theta) \\ Y_b = \cos \theta \cdot (Y_a + X_a \cdot \tan \theta) \end{cases} \quad (5.2)$$

将旋转角度 θ 拆分成 n 个小角度，依次记为 θ_1 、 θ_2 、 $\theta_3 \dots \theta_n$ ，如式 (5.3)， θ_x 的大小满足式 (5.4)。根据二分法的思想，可以用有限个依次减半的角度尽可能逼近角度 θ ，因此可以通过多次旋转逼近最终的旋转角度 θ ，其中，旋转的次数即为算法的迭代次数。每次迭代过程中，提取一个乘法系数 $\cos \theta_x$ ($x=1,2,3,\dots,n$)，如式 (5.5)，式中 X_{q1} 和 Y_{q1} 是第一次旋转 θ_x 后的坐标。在硬件实现时可用移位方式实现乘 2^{-x} 的计算，避免了乘三角函数的计算，有效降低了硬件计算复杂度。

$$\theta = \sum_{x=1}^n \theta_x \quad (5.3)$$

$$\theta_x = \arctan(2^{-x}) \quad x=1,2,3,\dots,n \quad (5.4)$$

$$\begin{cases} X_{q1} = \cos \theta_x \cdot (X_a - Y_a \cdot 2^{-x}) \\ Y_{q1} = \cos \theta_x \cdot (Y_a + X_a \cdot 2^{-x}) \end{cases} \quad (5.5)$$

根据前面的公式推导，最终硬件实现参考的计算公式为式(5.6)，每次旋转后的坐标为 (X_q, Y_q) ， (X_i, Y_i) 是这一次旋转之前的坐标。这里引入了系数 g 控制旋转方向，每次旋转后判断当前角度和参考角度 θ 的大小，如果大于 θ ，则令 $g=-1$ ，即进行反向旋转，如果小于 θ ，则令 $g=1$ ，即进行正向旋转，反复旋转后逼近 θ 。在 LTE-V2X 中，CORDIC 模块输入数据的量化宽度是 16 比特，经过仿真测试发现迭代 16 次后精度可以达到数据量化后的最低位。由于量化位宽的限制继续提高迭代次数无法进一步提高精度，因此，设定迭代次数 n 为 16。根据设定的迭代次数可以预先计算出最终的外部系数 $\prod_{x=1}^n \cos \theta_x$ ，从而避免了这部分的硬件计算。硬件实现时每次迭代只计算 $(X_i - g \cdot Y_i \cdot 2^{-x})$ 和 $(Y_i + g \cdot X_i \cdot 2^{-x})$ ，使用该方法最终计算出的坐标结果为 (X_m, Y_m) ，与旋转后的期望坐标 (X_b, Y_b) 以及原始坐标 (X_a, Y_a) 的关系满足式(5.7)。

$$\begin{cases} X_q = \prod_{x=1}^i \cos \theta_x \cdot (X_i - g \cdot Y_i \cdot 2^{-x}) \\ Y_q = \prod_{x=1}^i \cos \theta_x \cdot (Y_i + g \cdot X_i \cdot 2^{-x}) \end{cases} \quad g = \pm 1 \quad (5.6)$$

$$\begin{cases} X_b = X_m \cdot \prod_{x=1}^n \cos \theta_x = X_a \cdot \cos \theta - Y_a \cdot \sin \theta \\ Y_b = Y_m \cdot \prod_{x=1}^n \cos \theta_x = Y_a \cdot \cos \theta + X_a \cdot \sin \theta \end{cases} \quad (5.7)$$

根据式(5.7)即可实现对正余弦和相位旋转的计算。当需要进行正余弦计算时，先计算外部系数 $\prod_{x=1}^n \cos \theta_x$ ，作为固定参数，并使其满足式(5.8)，将式(5.8)代入式(5.7)，可以得到旋转后的坐标，其中横纵坐标分别为输入角度的余弦和正弦值，如式(5.9)所示。

$$\begin{cases} X_a = \prod_{x=1}^n \cos \theta_x \\ Y_a = 0 \end{cases} \quad (5.8)$$

$$\begin{cases} X_m = \cos \theta \\ Y_m = \sin \theta \end{cases} \quad (5.9)$$

当需要实现输入信号相位旋转的功能时,直接令原始坐标分别为输入信号的实部和虚部即可,最终输出结果和理论计算结果相差固定的倍数,相当于对信号做了功率放大,不会使信号发生畸变。根据 LTE-V2X 系统的需求,硬件设计时不做系数补偿。

5.2.2 设计需求

针对当前 LTE-V2X 无线通信系统,需要实现两种不同计算模式的 CORDIC,分别是根据相位角度计算正余弦值和相位旋转。结合具体系统的需求和现有的条件,设计的两种 CORDIC 模块需要满足以下要求:

- 1) 分别支持计算输入角度正余弦值和对输入数据进行相位旋转的功能,其中正余弦计算模块支持的输入角度范围是 $(-\pi, \pi)$ 。
- 2) 使用 16 次迭代计算,计算精度尽可能接近赛灵思的 CORDIC IP 核。
- 3) 综合频率可达 400MHz,综合后的建立时间和保持时间余量不能少于赛灵思的 CORDICIP 核。
- 4) 采用全流水计算结构,计算延时与赛灵思的 CORDIC IP 核在相同精度配置下相同。
- 5) 资源上进行优化,整体资源开销和赛灵思的 CORDIC IP 核接近,进入整个 LTE-V2X 系统工程中能满足硬件资源的要求。
- 6) 能与前后级模块正确的交互,端口位宽可以匹配前后级的模块,并且支持复位功能。

5.2.3 正余弦计算模式 CORDIC 模块的实现

根据前面推导出的算法进行硬件设计,正余弦计算模式的 CORDIC 模块整体实现框图如图 5.2 所示,其中主要包括相位预处理模块、延时单元、正负选择模块和 16 个迭代计算单元。

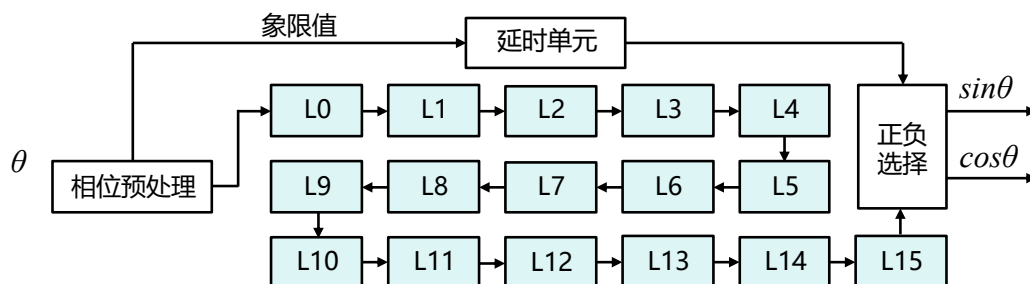


图 5.2 正余弦计算模式的 CORDIC 整体框图

其中, θ 是输入的相位角度, 输入范围为 $(-\pi, \pi)$ 。首先, 对输入相位进行预处理操作, 预处理中需要根据相位大小判断其所属象限, 并通过取反和加减 π 操作将相位搬移至第一象限中。随后, 根据式 (5.6) 进行 16 次迭代计算, 式中初始坐标 $x(0)$ 和 $y(0)$ 满足式 (5.8)。用 MATLAB 预先计算量化后的值为 $x(0)=19898, y(0)=0$, 每次迭代计算的实现方式如图 5.3 所示, 其中 “>>” 符号表示移位操作, 以此实现除 2 计算。 $x(k)$ 和 $y(k)$ 是旋转前的坐标, $x(k+1)$ 和 $y(k+1)$ 是旋转后的坐标, $z(k)$ 是当前剩余的相位角度, 根据 $z(k)$ 的正负选择进行正向旋转或者反向旋转, 并计算旋转后相位角度 $z(k+1)$ 。最后将输入角度的原始象限值用延时单元延时, 使其和 16 次迭代后的计算结果在时域上对齐, 对最后的计算结果进行正负选择。

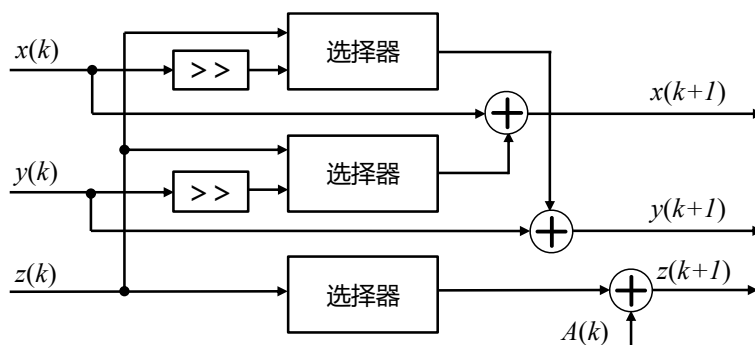


图 5.3 正余弦计算模式 CORDIC 模块每次迭代计算的示意图

图 5.3 中, $A(k)$ 是预先计算的参数, 满足式 (5.10)。由于前级模块输入的角度默认乘上了系数 π , 因此参考值 $A(k)$ 需要除以 π 。最终计算的参考值如表 5.1 所示, 这里的数据已经做了量化处理。

$$A(k) = \arctan(2^{-k})/\pi, k = 0, 1, 2, 3, \dots, 15 \quad (5.10)$$

表 5.1 预先计算的 $A(k)$ 参考值

k	$A(k)$	量化后的值
0	0.25	2097152
1	0.14758361	1238021
2	0.07797913	654136
3	0.03958342	332050
4	0.01986852	166669
5	0.00994394	83416
6	0.00497318	41718
7	0.00248674	20860
8	0.00124339	10430
9	0.00062169	5215
11	0.00031084	2608
12	0.00015542	1304
13	7.7712E-05	652
14	3.8856E-05	326
15	1.9428E-05	163

5.2.4 相位旋转模式 CORDIC 模块的实现

相位旋转模式 CORDIC 模块硬件实现结构与正余弦计算功能的设计类似，如图 5.4 所示。根据 5.2.1 节的原理，此时旋转起始坐标为输入数据的实部 din_re 和虚部 din_im ，同样使用了 16 次迭代。根据旋转角度的原始象限位置选择是否对结果进行取反操作，最终输出计算结果的实部 $dout_re$ 和虚部 $dout_im$ 。

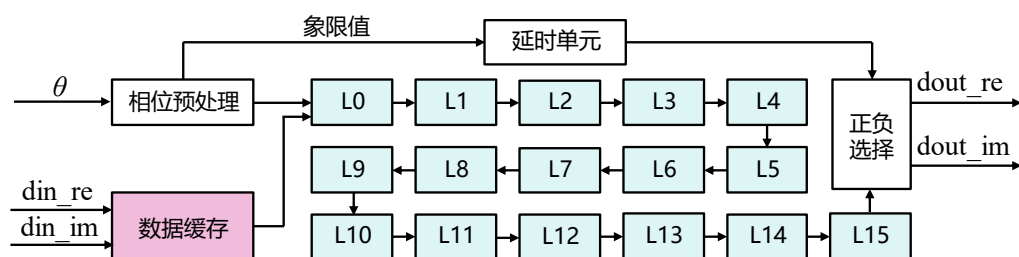


图 5.4 相位旋转模式的 CORDIC 模块整体框图

相位旋转模式 CORDIC 模块中每次迭代计算示意图如图 5.5 所示，与正余弦计算模式 CORDIC 模块的迭代过程有所不同，在正余弦计算模式下，起始坐标的象限是未知的。根据数学推理，如果起始坐标位于第一或第三象限，每次迭代旋转方向和正余弦计算功能的 CORDIC 相同。如果起始坐标位于第二或第四象限，那么每次迭代旋转方向和正余弦计算功能的 CORDIC 正好相反。因此判断时需要考虑起始坐标象限，硬件设计中用一组移位寄存器将是否为一三象限的结果 $Q(k)$ 进行延时，在每次迭代的时候与当前剩余角度 $z(k)$ 进行同或计算，根据结果选择旋转方向。相位旋转模式的 CORDIC 前级输入角度中系数未乘以 π ，此时， $A(k)$ 需要满足式 (5.11)，硬件设计过程同样包括量化处理。

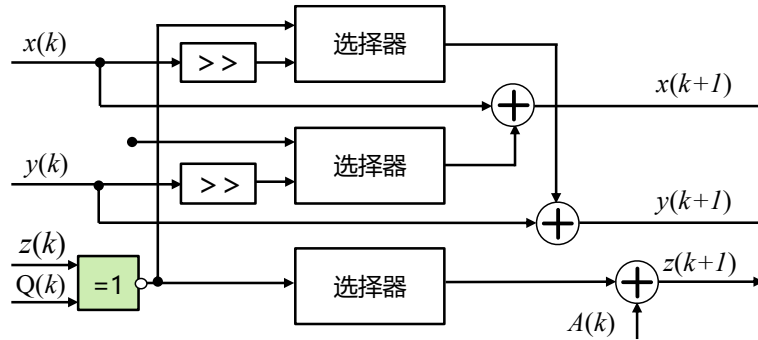


图 5.5 相位旋转模式 CORDIC 模块每次迭代计算的示意图

$$A(k) = \arctan(2^{-k}), k = 0, 1, 2, 3, \dots, 15 \quad (5.11)$$

5.2.5 测试结果

根据设计的结构编写 RTL 代码。在 VIVADO 软件中仿真时使用相同位宽和迭代次数配置的 CORDIC IP 核作为参考模型，采用第二章图 2.9 的验证方法进行功能验证。首先输入相同的随机测试向量，然后将参考模型和设计模块的输出结果在时域上对齐后进行相减，判断误差的大小。

本文设计的正余弦计算模式和相位旋转模式 CORDIC 模块仿真结果如图 5.6 和图 5.7 所示。图中的数据都转换成了模拟波形，theta 是输入的角度，输入从 $(-\pi, \pi)$ 递增（图中两根蓝线中间的范围），硬件设计时做了量化处理。sin_sub 和 cos_sub 是差值计算结果，在正余弦计算功能的 CORDIC 模块仿真时，差值的波形窗口幅度设置为 $(-3, 3)$ ，在相位旋转模式 CORDIC 模块仿真时差值窗口幅度

设置为 $(-10, 10)$ 。可以看出设计的正余弦计算功能 CORDIC 模块和赛灵思 CORDIC IP 核的数据误差为 ± 3 ，相位旋转模式 CORDIC 模块的数据误差为 ± 10 。经过和输出数据数值的比较分析，设计的两个 CORDIC 模块和赛灵思 CORDIC IP 核的相对误差小于 0.5%。经过大规模系统级的仿真和在 FPGA 开发板上的长时间测试，该误差可以满足 LTE-V2X 系统的要求。

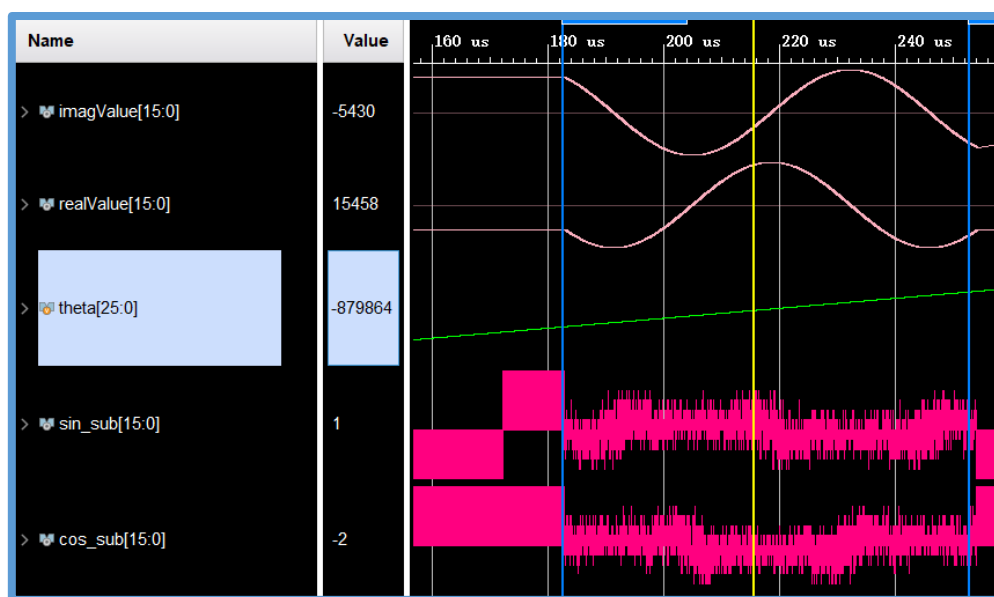


图 5.6 正余弦计算模式 CORDIC 模块的仿真结果

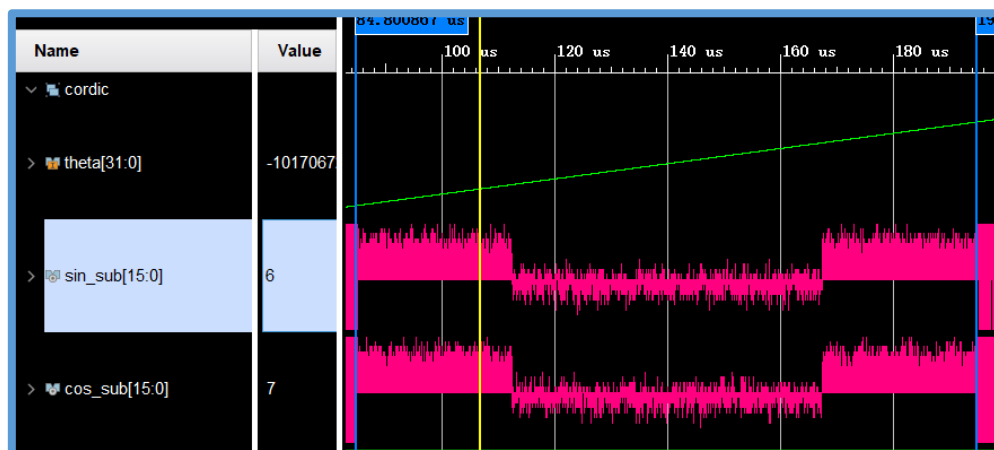


图 5.7 相位旋转模式 CORDIC 模块的仿真结果

完成功能验证后，对设计的模块分别进行综合和静态时序分析，优化了时序违例的关键路径，最终可以综合到 400MHz。综合后分析硬件资源的使用情况以及和赛灵思 CORDIC IP 核的对比情况，结果如表 5.2 和表 5.3 所示，可以看出设计的正余弦计算模式 CORDIC 相对于原工程中赛灵思 CORDIC IP 核资源消耗有

所增加。设计的相位旋转模式 CORDIC 比原工程中赛灵思 CORDIC 的资源开销降低了 50%以上。在 LTE-V2X 系统中总共使用了两个正余弦计算模式的 CORDIC 模块和三个相位旋转模式 CORDIC，根据表 5.2 和表 5.3 的结果进行推算，将工程中原有的 5 个赛灵思 CORDIC IP 核全部替换成本文设计的模块后，总体查找表和片寄存器的消耗分别降低 52.0%和 52.8%。

表 5.2 正余弦计算模式 CORDIC 模块的资源分析

资源种类	查找表	片寄存器	块随机存储器	数字信号处理器单元
赛灵思 CORDIC	1395	1420	0	0
本文设计的 CORDIC	1824	1710	0	0

表 5.3 相位旋转模式 CORDIC 模块的资源分析

资源种类	查找表	片寄存器	块随机存储器	数字信号处理器单元
赛灵思 CORDIC	4648	4675	0	0
本文设计的 CORDIC	1457	1511	0	0

本文设计的相位旋转 CORDIC 模块能够有效降低硬件资源消耗，其原因在于本文将特定的位宽和配置与 LTE-V2X 具体场景进行结合，在此基础上进行设计，而赛灵思 CORDIC IP 核则是为了多种不同场景和配置而设计的。本文的设计相当于根据定制化需求舍弃了灵活性，最终在硬件资源开销上和赛灵思同类产品相比更具优势。此外，设计的模块计算延时可以和原 IP 保持一致，如表 5.4 所示，其中单位 clk 为系统时钟周期。

表 5.4 计算延时比较分析

工作模式	正余弦计算模式延时 (clk)	相位旋转模式 (clk)
赛灵思 CORDIC	22	39
本文设计的 CORDIC	22	39

将设计的 CORDIC 模块整合到 LTE-V2X 总体工程中，替换了原来在发端和收端使用的 5 个 CORIDC IP 核，进行大规模仿真和在 FPGA 开发板上的长时间测试，通过检测控制信道和共享信道的解包成功率验证了设计的可靠性。

5.3 FFT 与 IFFT 计算模块的设计

5.3.1 算法原理

FFT 是一种用于计算 DFT 的高效算法，适用于大小为 2 的正整数次幂数据样本。DFT 将一个时域信号转换为频域信号，其定义如式 (5.12) 所示，其中， $X(k)$ 表示频域序列， $x(n)$ 表示时域序列， N 是变换大小， n 和 k 都是整数。

$$X(k) = \sum_{n=0}^{N-1} x(n) \cdot e^{-\frac{2\pi nk \cdot j}{N}} \quad (5.12)$$

FFT 算法通过将 DFT 分解为更小的 DFT，从而减少计算复杂度。对于 N 点 DFT，基本的 FFT 算法将其分解为两个 $N/2$ 点的子序列 DFT。这样可以大大减少计算量，并降低计算的复杂度。IFFT 的算法与 FFT 类似，只改变了蝶形因子的正负，如式 (5.13)。

$$X(k) = \sum_{n=0}^{N-1} x(n) \cdot e^{\frac{2\pi nk \cdot j}{N}} \quad (5.13)$$

具体实现时使用蝶形图进行计算，以 8 点的 FFT 为例，总体蝶形计算的流程如图 5.8 所示，一个蝶形单元如图 5.9 所示， $W(k)$ 表示蝶形因子，满足式 (5.14)。

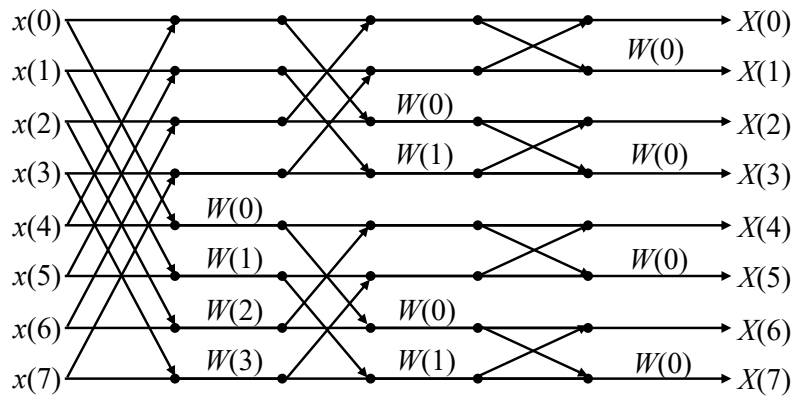


图 5.8 8 点 FFT 蝶形计算图

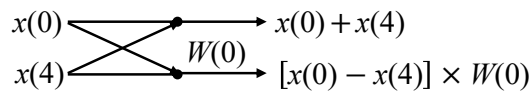


图 5.9 一个蝶形单元结构

$$W(k) = e^{\frac{-2\pi nk \cdot j}{N}} \quad (5.14)$$

根据图 5.8，计算一个 8 点 FFT，总共需要进行三级蝶形计算。对于第一级蝶形计算，有两种可以实现流水的结构，第一种是使用四个蝶形单元并行计算，这种方式最快，计算三级蝶形总共需要 7 个蝶形单元。另一种是使用一个蝶形单元依次串行计算四次，这种计算方式使用资源较少，但需要的时间更多，计算三级蝶形共需要 3 个蝶形单元。根据 8 点 FFT 的蝶形图进行推导，当计算 N 点 FFT 时，第一种并行计算方法和第二种串行计算方法所需要的蝶形单元数量 S_1 和 S_2 分别为式 (5.15) 和 (5.16) 所示，其中 N 是 FFT 的点数。在 LTE-V2X 中需要进行 2048 点大小的 FFT 计算，代入式 (5.15) 和 (5.16) 可以得出，使用并行计算方式时总共需要 1919 个蝶形单元，这对于硬件实现来说需要消耗大量的逻辑计算资源，不利于 FPGA 的实现，而如果利用串行计算方式计算每一级蝶形，最终只需要 11 个蝶形单元。因此，针对 LTE-V2X 中较大点数的 FFT 计算，硬件设计时采用了第二种串行计算方式计算每一级的蝶形。

$$S_1 = \sum_{k=1}^{\log_2 N} \frac{N}{2^k} \quad (5.15)$$

$$S_2 = \log_2 N \quad (5.16)$$

此外，经过蝶形计算后，最终输出的数据顺序会发生变换。为了得到和输入数据对应的顺序输出数据，需要将输出数据进行比特重排。比特重排的规则是将十进制的序号转换成无符号的二进制数，然后将二进制数按位前后倒置。对于 8 点 FFT 计算，假设输入数据的序号为 3，用 3 位二进制表示为 011，对应输出序号的二进制数就变为 110，转换为 10 进制就是 6，以此类推。

5.3.2 设计需求

数字设计流程第一步是确定整体的设计需求，针对目前的 LTE-V2X 工程，需要分别实现 2048 点 FFT 和 2048 点 IFFT 模块，具体需求如下：

- 1) 分别支持 2048 点的 FFT 和 IFFT 计算，在相同数据量化位宽的情况下和赛灵思 FFT IP 核的计算误差尽可能小。
- 2) IFFT 模块需要支持插入可配置长度循环前缀的功能。

- 3) FFT 和 IFFT 输出模式分别支持定点截位和不截位的输出格式。
- 4) 采用基 2 的流水结构, 计算延时和赛灵思 FFT IP 核保持一致。
- 5) 优化资源利用率, 合理分配存储资源 (主要难点)。
- 6) 综合频率可以达到 400MHz, 满足静态时序要求。
- 7) 使用参数化定义, 提高模块可复用性。
- 8) 代码可读性强, 易于修改和配置不同位宽和大小的 FFT 和 IFFT。

5.3.3 硬件设计

根据指定的设计需求开始具体的设计, 首先使用 MATLAB 模拟硬件设计的计算过程搭建参考模型。首先生成 2048 个随机数作为测试向量, 接着按照式 (5.10) 生成 1024 个蝶形因子, 然后按照蝶形计算的流程进行 11 级蝶形计算。接着将输出结果进行比特重排, 最后得到顺序输出的 FFT 计算结果。由于 MATLAB 中自带 FFT 和 IFFT 系统函数, 可以使用 FFT 和 IFFT 系统函数和按照 11 级蝶形结构编写的 MATLAB 参考模型进行比较, 从而验证最终的计算结果。图 5.10 是搭建的 MATLAB 参考模型与 MATLAB 系统 FFT 函数的结果对比, 其中分别比较了输出复数结果实部和虚部的误差, 从图中可以看出搭建的参考模型和 MATLAB 系统 FFT 函数的计算误差小于 1.5×10^{-11} 。

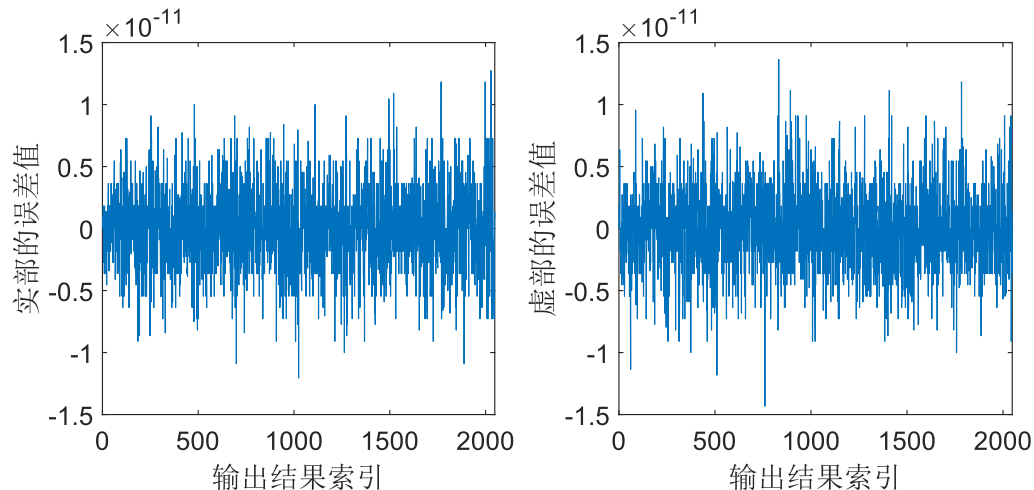


图 5.10 MATLAB 仿真结果实部 (左) 和虚部 (右) 的误差值

完成 MATLAB 参考模型的搭建后开始硬件设计, 2048 点 FFT 和 IFFT 计算模块硬件实现的总体框图如图 5.11 所示, 其中 PE 表示每一级蝶形模块。FFT 和

IFFT 的硬件实现结构几乎相同。根据式 (5.9)，在计算相同点数 IFFT 时只需要将蝶形因子虚部的值取反即可。在实际 LTE-V2X 系统工程中，FFT 和 IFFT 模块的输出要求有所不同，2048 点 FFT 模块需要将输出的 2048 个数据截位保留 16 比特数据并给出输出数据的缩放尺度，IFFT 模块的输出则不需要截位，但增加了添加循环前缀的功能，需要根据前级模块控制信号选择添加长度为 144 或 160 的循环前缀，满足 LTE-V2X 发端需求，避免符号间的串扰。

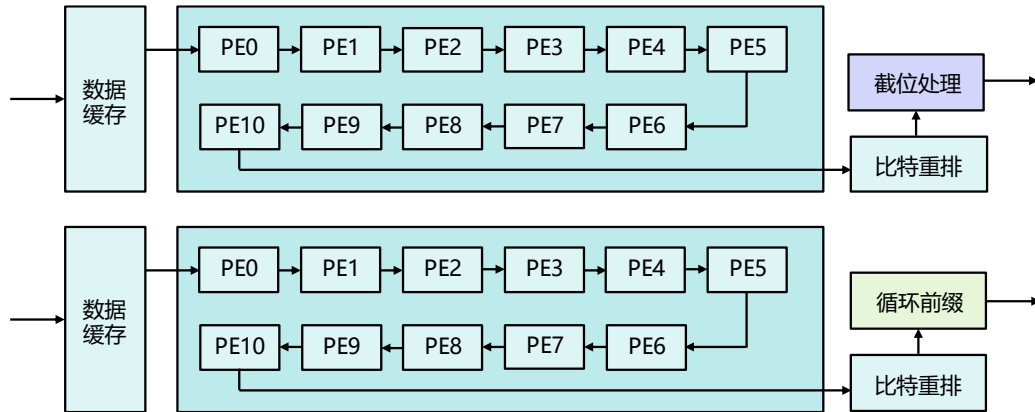


图 5.11 FFT（上）与 IFFT（下）计算模块的硬件实现结构

硬件设计的关键是每一级蝶形设计，根据 5.3.1 节的推导，具体设计时每一级蝶形使用了串行的计算方式，复用同一个蝶形计算单元完成多次蝶形计算。第一级蝶形模块的内部结构如图 5.12 所示，根据蝶形计算图，首先将输入的数据延迟 1024 个数据点，然后将延时后的数据与原数据在时域上分别进行加法和减法，并将减法的结果延迟 1024 个数据点，通过数据旋转模块将减法结果与加法结果在时域上合并，接着乘上对应的蝶形因子，这里的蝶形因子使用 MATLAB 预先计算和量化后存储在 ROM 中。由于每次乘法的结果数据位宽都会增大，为了减少硬件资源的消耗，每次乘法后都对结果进行截位。为了防止进行加法和乘法操作时溢出，每一级蝶形计算保留的数据位也做了相应扩充。参考图 5.9 的蝶形图，在进行 2048 点 FFT 计算时，第一级蝶形需要将 $x(0)$ 和 $x(1024)$ 在时域上对齐后进行计算，因此需要的数据缓存深度为 1024，存储蝶形因子的 ROM 大小为 1024。到第二级的蝶形计算时， $x(0)$ 需要和 $x(512)$ 在时域上对齐，且旋转因子的数量也减半，因此只需要深度为 512 的 RAM 缓存数据和深度为 512 的 ROM 存储蝶形因子。以此类推，每一级需要的存储模块深度都会减半，通过该设计合理利用了蝶形计算的特点，有效降低流水计算的硬件资源开销。

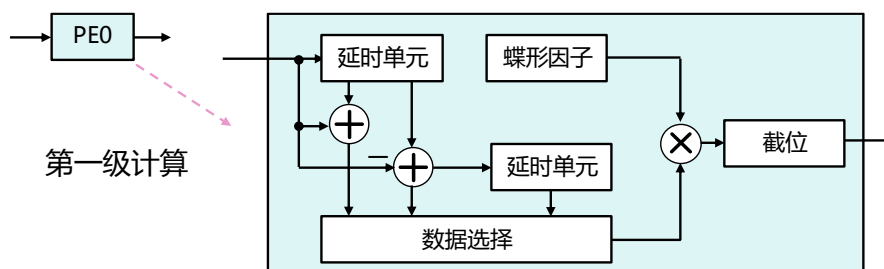


图 5.12 第一级蝶形计算模块的结构

对于最后的比特重排模块，硬件设计时使用了一块深度为 2048 的 RAM，首先将输出数据顺序存入 RAM 中，然后生成顺序的输出序号，将序号的二进制比特位进行倒置后作为 RAM 的读地址，读出数据时就完成了数据的重排。根据需求，在 FFT 模块中，需要对输出进行定点截位，因此需要遍历所有的计算结果确定最大值。从最大值的最高有效位向下保留截位，这里可以复用比特重排的 RAM，在存数据的时候就可以找到最大值，读出数据的时候就可以截位。对于 IFFT，需要添加循环前缀，在产生 RAM 的读信号时增加一段循环的读地址信号即可添加任意长度的循环前缀。

5.3.4 测试结果

根据 MATLAB 参考模型完成硬件设计，每一级蝶形的计算结果都和 MATLAB 的计算结果进行比对，当完成整个 FFT 和 IFFT 的设计后，使用在前面 CORDIC 相似的验证方法进行仿真验证。在 VIVADO 软件仿真环境中使用的参考模型是赛灵思 FFT 和 IFFT IP 核，输入的测试向量在测试脚本文件中使用随机数生成函数产生。在验证过程中发现了一些问题，例如数据溢出，通过和 MATLAB 的参考模型进行比对找到了这些问题的原因。

最终与赛灵思 FFT/IFFT IP 核进行了长时间的仿真测试，如图 5.13 所示。将设计的 FFT 与赛灵思 FFT 的差值信号转化为模拟波形，图中每一段 `real_cut` 和 `im_cut` 信号都表示 2048 点 FFT 计算结果的差值，最终结果显示本文设计的 FFT 和 IFFT 与赛灵思 FFT/IFFT IP 核的误差小于 0.5%，经过分析该误差是由于旋转因子在量化时最后一位不同引起的（本文设计的 FFT 和 IFFT 模块旋转因子量化时采用了四舍五入的保留方式，赛灵思采用的是直接截位），在实际测试中该误差不会影响后级模块的解包成功率。

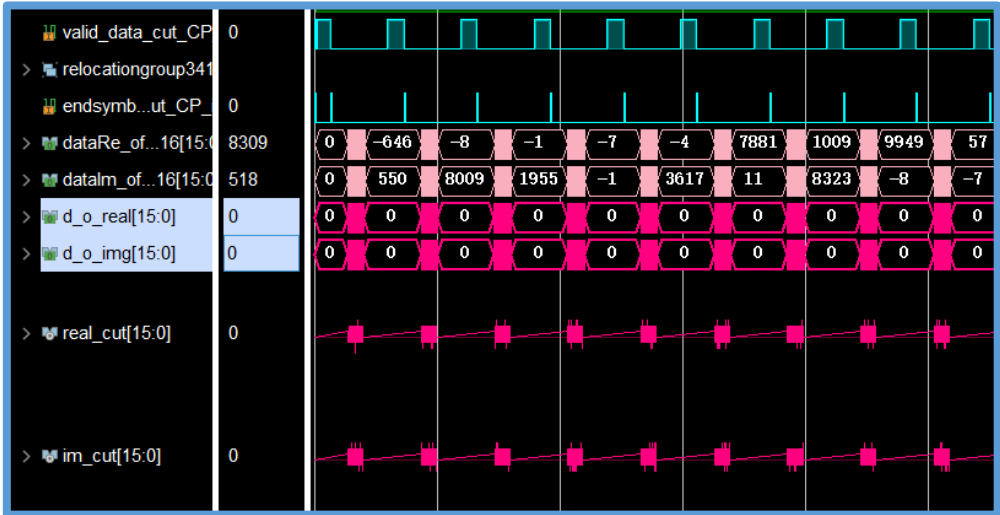


图 5.13 VIVADO 软件仿真结果

设计的 FFT 与 IFFT 模块输出延时比赛灵思对应的 IP 核减少了 16 个时钟周期，为了不改变系统时序，增加了延时模块。经过仿真和上板测试，设计的 2048 点 FFT 模块延时为 6309 个时钟周期，IFFT 模块为 6314 个时钟周期，工作频率为 307.2MHz，能够在 20.57 微秒内完成一次 2048 点 FFT 或 IFFT 计算。

仿真验证完成后，对设计的 FFT 和 IFFT 模块进行综合，分析硬件资源使用情况，并和赛灵思 FFT/IFFT 进行比较。根据对比结果可以得出设计的 FFT 与 IFFT 模块与商用的赛灵思 IP 核在资源方面仍有差距。目前 LTE-V2X 工程中查找表资源较为宽裕，块随机存储器资源比较紧张，因此设计时更关注块随机存储器资源的开销。本文设计的 FFT 和 IFFT 模块共消耗了 16 个块存储器资源，占整体工程块存储器资源的 2.94%，可以满足系统对于资源的要求。

表 5.5 设计的 FFT 模块资源分析

资源种类	查找表	片寄存器	块随机存储器	数字信号处理器单元
赛灵思 FFT	3754	6202	8	19
本文设计的 FFT	8364	9805	7	41

表 5.6 设计的 IFFT 模块资源分析

资源种类	查找表	片寄存器	块随机存储器	数字信号处理器单元
赛灵思 IFFT	5676	8205	7	48
本文设计的 IFFT	9027	10762	9	50

将设计的 FFT 和 IFFT 模块加入 LTE-V2X 总体工程中，替换原有的 IP 核，进行整体工程仿真、综合、布局布线等，通过了前仿和后仿测试以及静态时序分析，最终在 FPGA 开发板上进行了长时间的测试，测试结果表明设计的 2048 点 FFT 和 IFFT 可以满足系统精度和资源的需求。

5.4 本章小结

本章介绍了 CORDIC 和 FFT/IFFT 模块原理和实现方案，设计实现了四个 LTE-V2X 系统中的关键模块，分别是：正余弦计算的 CORDIC 模块、相位旋转的 CORDIC 模块、2048 点 FFT 模块和 2048 点支持配置循环前缀的 IFFT 模块。具体实现过程中，本文从数学式开始推导，搭建 MATLAB 参考模型，之后完成硬件设计，最终和赛灵思的 IP 核进行验证对比。经过充分的仿真和在 FPGA 开发板上长时间的测试，由测试结果可知，设计的四个模块综合频率可以达到 400MHz，工作频率为 307.2MHz，支持 368 种信道配置。其中，本文设计的相位旋转模式 CORDIC 模块资源消耗比之前的工作降低了 52.1%，设计的 FFT 与 IFFT 模块消耗的块存储器资源分别为 7 和 9，可以在 20.57 微秒内完成 2048 点输入数据的 FFT 或 IFFT 计算。

第六章 LTE-V2X 系统集成与测试

本章主要介绍了 LTE-V2X 系统集成工作，将前几章设计的模块接入整个 LTE-V2X 工程中，进行整体工程仿真和在 FPGA 开发板上的测试。首先介绍了 LTE-V2X 整体系统组成和具体的测试方案，然后分析了系统的时序和资源使用情况，最终在 FPGA 开发板上进行了和商用 C-V2X 芯片的视频传输测试。

6.1 LTE-V2X 系统集成

将本文第三、四、五章设计的模块整合到 LTE-V2X 整体工程中进行系统集成。系统集成图如图 6.1 所示，其中使用两台电脑分别作为发端和收端的上位机，两块课题组自研的 FPGA 开发板分别作为系统的发射机和接收机。

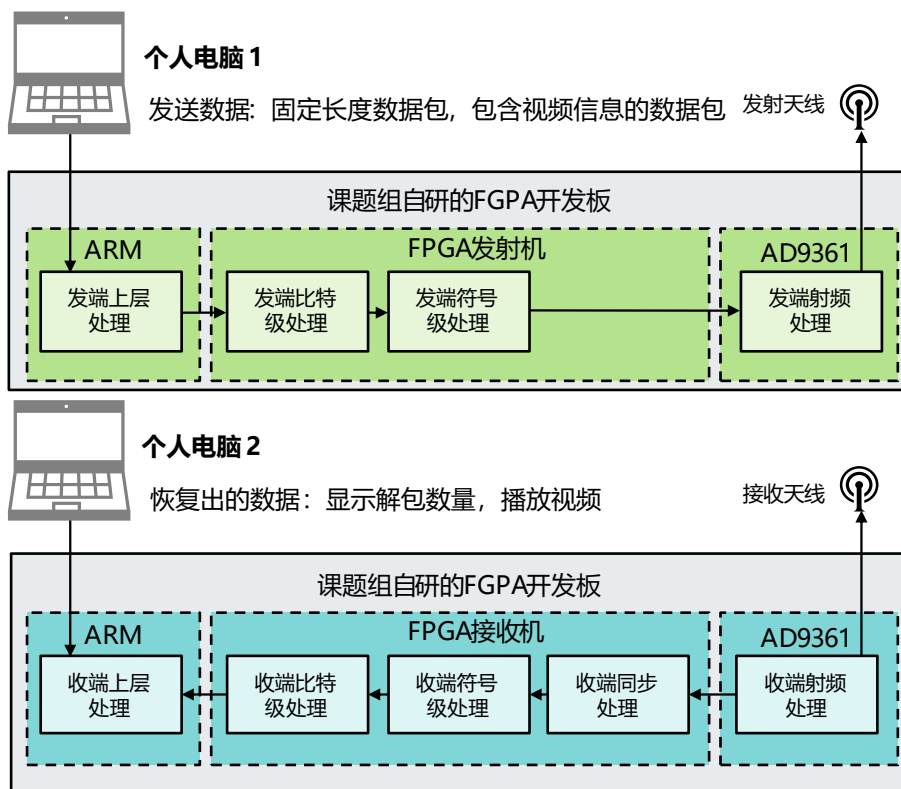


图 6.1 LTE-V2X 系统集成图

6.2 整体时序和资源分析

在 VIVADO 软件中对整体工程进行综合、布局布线、生成比特流等操作，

最终对实现结果的整体时序和资源结果进行分析。表 6.1 是工程在工作频率为 307.2 MHz 时的静态时序分析结果，在 VIVADO 软件中将 LTE-V2X 工程文件成功生成比特流文件后，软件会给出最终的实现结果。其中，最差负总裕量（Worst Negative Slack, WNS）表示所有时钟域中，时钟边沿之间最短的负时序余量，单位是纳秒，负值表示存在时序问题，如果 WNS 为正数或零，则表示电路满足时序要求。总负总裕量（Total Negative Slack, TNS）是另一个用于衡量数字集成电路设计时序性能的重要参数。TNS 是所有不满足时序要求的寄存器路径上负时序余量的总和，正值或零值表示整个设计满足时序要求。最差保持时间余量（Worst Hold Slack, WHS）表示电路中保持时间余量最小的路径，正值或零值表示电路满足保持时间要求。根据上述这些参数的概念，结合表 6.1 的结果可以得出，LTE-V2X 总体工程满足数字设计中对于静态时序的要求。

表 6.1 工作频率为 307.2MHz 时的静态时序分析结果

静态时序分析标准	时间（/纳秒）
WNS	0.059
TNS	0.000
WHS	0.012

在使用 VIVADO 软件综合实现后，软件会给出整体硬件资源的使用情况，LTE-V2X 工程详细的整体资源使用情况如表 6.2 所示。

表 6.2 LTE-V2X 系统总体资源具体分析

资源种类	资源使用量	可用资源	利用率%
LUT	109836	218600	50.25
LUTRAM	8610	70400	12.23
FF	124565	437200	28.49
BRAM	350.50	545	64.31
DSP	407	900	45.22
IO	49	362	13.54
MMCM	1	8	12.50
PLL	1	8	12.50

其中资源的英文缩写含义如下。

LUT 表示查找表 (Look-Up Table, LUT)，是一种常用的逻辑单元，用于实现数字电路中的各种逻辑功能。LUTRAM 表示与 LUT 相关的内存单元，可用于实现存储器和计数器等功能。FF 表示触发器 (Flip-Flop, FF)，是一种常用的存储单元，用于实现时序逻辑和状态机等功能。BRAM 表示块随机存储器 (Block Random Access Memory, BRAM)，是一种可编程的存储器单元，可以用于实现大容量的存储器和 FIFO 等功能。DSP 表示数字信号处理器 (Digital Signal Processor, DSP)，是一种专用的硬件模块，用于加速数字信号处理算法的运行。IO：输入/输出 (Input/Output, IO)，是 FPGA 设备与外部设备进行数据传输和通信的接口。MMCM 表示多模时钟管理器 (Mixed-Mode Clock Manager, MMCM)，是一种可编程的时钟模块，用于实现多个时钟域之间的时钟转换和时序控制。PLL 表示锁相环 (Phase-Locked Loop, PLL)，是一种常用的时钟模块，用于生成稳定的时钟信号，并可以对时钟信号进行频率合成和时钟延迟等控制。

根据上述数据进行分析可以得出，整体设计的资源使用分配较为合理，在满足系统功能需求的同时查找表资源有 49.75% 的余量，块存储器有 35.69% 的余量。

6.3 固定长度数据包传输测试

将本文设计的所有模块接入 LTE-V2X 整体工程中进行测试。首先进行固定长度数据包的传输测试，在 FPGA 开发板使用表 6.3 中的 4 种方式进行测试。

表 6.3 FPGA 开发板固定长度数据包传输测试方式

序号	测试方式
1	单块开发板自发自收通信 (直连)
2	单块开发板自发自收 (空口)
3	两块开发板通信 (直连)
4	两块开发板通信 (空口)

其中直连指的是使用屏蔽线连接 FPGA 开发板发端和收端的微型同轴射频接口 A 型 (SubMiniature version A, SMA) 接口，进行有线传输，这种传输方式的噪声和频偏干扰最小，用于早期的功能验证。空口指在开发板发端和收端的

SMA 接口分别接上天线进行无线传输，这是后期大规模测试的主要场景。通过两块带有天线的 FPGA 开发板模拟了车联网中车与车、车与道路设施之间的无线通信场景。

在烧录配置的软件中配置顶层参数，从而选择不同的信道配置。最终的 LTE-V2X 工程支持 368 种不同的信道配置，表 6.4 展示了部分信道配置的参数，PRB_Start 表示有效信息资源块起始的位置是第几个 RB，I_MCS 代表 LTE 协议中调制编码策略的序号，

表 6.4 部分信道配置和解码测试结果

序号	码块大小	PRB_Start	I_MCS	控制信道	共享信道
1	2664	0	0	500	499
2	3496	0	1	500	500
3	4264	0	2	499	499
4	5544	0	3	499	499
5	6968	0	4	499	499
6	8504	0	5	499	499
7	9912	0	6	499	499
8	11832	0	7	499	499
9	13536	0	8	499	499
11	16992	0	10	499	499
12	16992	0	11	499	499
13	19080	0	12	499	499
14	22152	0	13	499	499
15	24496	0	14	500	499
...
368	632	0	17	500	499

通过控制信道和共享信道的解码数量判断传输的正确性和可靠性。在上面的信道配置表中，上位机设定的发送数据包为 500 个，根据收端的控制信道和共享信道正确解码数据包的结果就可以验证整体系统各个模块功能的正确性，后期进

行了长时间的测试工作，利用按键脚本等工具在 FPGA 开发板上进行了几千次烧录和测试结果记录，最终的测试结果如表 6.5 所示。

表 6.5 LTE-V2X 系统具体测试结果

类别	测试结果
支持的信道配置数量	368 种
解包成功率	99.74%
单次传输最大有效比特数	31704 比特
传输数据包平均速度	998.38/s
最大平均传输速率	31.65Mbps

在 LTE-V2X 整体工程中每次添加新的模块都需要重新进行测试，在测试中遇到了许多问题，例如一些模块数据溢出，或者处理时间不够等问题。上板测试与仿真不同，无法同时观察到所有信号，因此首先标记可能存在问题的信号，接着在上板测试过程中拉取相关的信号，将信号进行格式转换后导入 MATLAB 参考模型，定位问题位于 LTE-V2X 发端还是收端。如果 MATLAB 中无法正确解码，则大概率说明是发端存在问题，否则可能是后端的硬件处理过程与 MATLAB 不一致，再通过整体工程仿真和单模块仿真方法即可进一步的定位问题。通过上板测试和软件仿真相结合的方法，本工作最终解决了调试过程中遇到的问题，成功通过了 368 种信道配置的上板测试。

6.4 无线视频传输测试

完成固定长度数据包测试后，使用两块 FPGA 开发板进行无线视频传输的测试。将本文集成的 LTE-V2X 工程分别烧录到两块待测的开发板上，按照图 6.1 中的系统场景进行测试。视频传输测试场景如图 6.2 所示，共使用两块开发板和两台电脑进行无线通信。用这种测试方法来模拟车辆网中的实际通信场景，并使用投屏的方式将两台电脑的视频显示结果显示在同一台显示器上，其中左边窗口是发送的视频，右边窗口是接收的视频。视频传输过程中，首先在电脑中选择待传输的 720P 分辨率视频，用 VLC 视频软件将视频转换为传输的数据包格式，然后通过发射机进行发送，接收机接收数据信号并解码，最终在收端的电脑流畅地

播放出发送的原始视频，实现了开发板之间实时的无线视频传输。

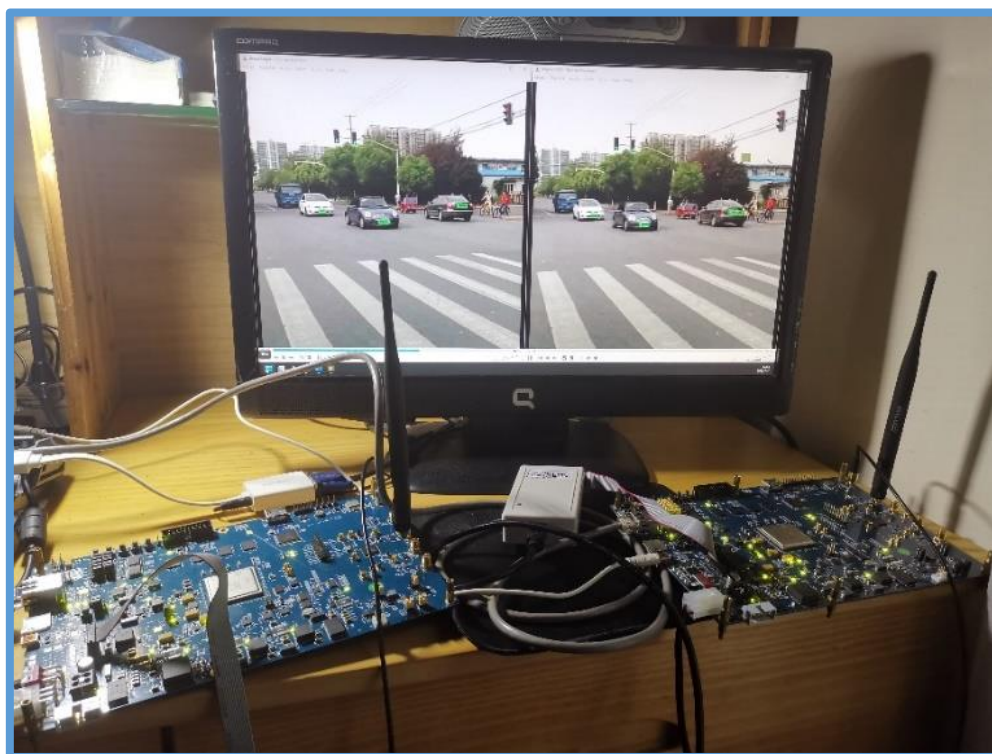


图 6.2 LTE-V2X 收发机视频传输测试场景

在完成 LTE-V2X 工程发射机和接收机的无线视频传输测试后，本文也进行了与商用大唐 VU3004 C-V2X 车载终端芯片的互连测试，使用的大唐 C-V2X 终端芯片型号如图 6.3 所示。



图 6.3 测试的大唐 C-V2X 终端

图 6.4 中测试场景一展示了大唐终端设备之间的收发测试与本文集成 LTE-V2X 系统之间的收发测试，图中的大唐 TX、大唐 RX 分别代表大唐设备的发射机和发射机，ACE-TX、ACE-RX 分别代表 LTE-V2X 系统发射机和接收机。通过

该场景的测试确认了 LTE-V2X 系统和大唐设备的发射机和接收机都工作正常。

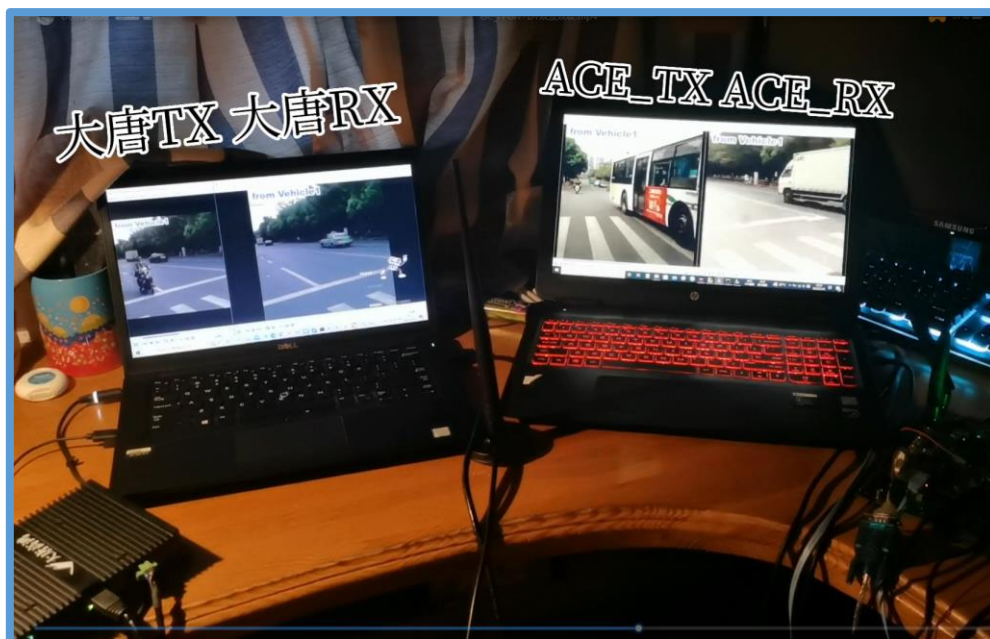


图 6.4 视频传输测试场景一

接着，在图 6.5 的测试场景二中，关闭了大唐设备的发射机，使用本文研究的 LTE-V2X 发射机发送视频信号，大唐设备的接收机接收信号。最后，在图 6.6 的测试场景三中使用大唐的设备作为发射机，LTE-V2X 作为接收机。通过上述三个场景的测试实现了本文研究的 LTE-V2X 系统与大唐商用 C-V2X 芯片的互连，进一步验证了本文研究工作的正确性。



图 6.5 视频传输测试场景二



图 6.6 视频传输测试场景三

6.5 本章小结

本章介绍了 LTE-V2X 系统的集成工作，将本文第三、四、五章设计的模块加入整体 LTE-V2X 工程进行实际测试。验证了本文设计的关键模块在不同测试场景下的可靠性，并分析了整体工程的时序和硬件资源情况。最终，成功实现了两块 FPGA 开发板之间的视频传输，并与大唐 VU3004 C-V2X 芯片实现了互连，进一步验证了本文工作的正确性。

第七章 结论与展望

7.1 论文工作总结

过去几十年,道路车辆数量持续增长,导致交通拥堵和交通事故发生次数不断增加。为解决这些问题,智能交通系统快速发展。其中,车联网技术是智能交通系统中的重要技术,LTE-V2X 技术为车联网技术发展提供了有力的支持。本文针对 LTE-V2X 的场景,设计了部分 LTE-V2X 物理层无线通信系统中的关键模块,给出了相关的算法原理和硬件设计方案,整体工作成果如下:

1) 提出了一种基于主同步信号的 LTE-V2X 收端同步算法,参考传统算法,针对 LTE-V2X 的实际场景进行优化,使用 MATLAB 对优化后的算法建模和分析,参考 MATLAB 模型进行了 FPGA 设计,并进行了充分的仿真和验证,最终在 FPGA 开发板上进行了长时间的测试。测试结果表明,设计的收端同步模块可以有效地实现在接收机中对接收信号的定时和频偏补偿,支持 307.2MHz 的工作频率,同步建立时间小于 1 秒,维持时间大于 24 小时,同步成功后收端解包成功率为 99.74%。

2) 针对 LTE-V2X 接收机中的解速率匹配问题提出了两种解速率匹配硬件设计方案,在硬件实现时有效降低了算法复杂度和资源开销。对于控制信道的固定配置的场景,设计了一种类似查表的解速率匹配结构,对于共享信道灵活配置的场景,设计了一种基于虚比特校验方法的解速率匹配方案,可以满足 368 种不同的信道配置。对设计的两种结构都进行了优化和大规模仿真,最终在 FPGA 开发板上通过了测试,根据收端解包结果验证了设计的可靠性。

3) 根据 LTE-V2X 收发机中对信号处理的需求设计了 CORDIC 和 FFT/IFFT 模块。其中 CORDIC 模块需要完成正余弦计算和相位旋转的功能,FFT/IFFT 模块需求完成 2048 点数据时域和频域之间的转化。从算法推导开始,依次进行 MATLAB 建模和 FPGA 实现,完成了预先设定的所以设计指标,对设计的模块进行了充分的仿真验证、综合分析等。将设计的模块和赛灵思 IP 核进行了比较,最终测试结果表明,设计的 CORDIC 和 FFT/IFFT 可以替换原 LTE-V2X 工程中相应的 IP 核,能够满足精度和资源的要求。

4) 完成了 LTE-V2X 系统的集成与测试。将本文设计的模块加入整个 LTE-V2X 系统中, 并与其他模块进行了联调, 对整体的资源和时序进行了分析, 测试结果显示该系统支持 368 种不同的信道配置, 最大平均传输速率为 31.65Mbps, 解包成功率为 99.74%。此外, 本文还测试了 LTE-V2X 系统与商用大唐芯片的互连通信, 实现了 720P 无线视频的传输, 验证了本文所设计模块的正确性。

7.2 未来工作展望

本文主要工作是设计了 LTE-V2X 无线通信系统中的一系列关键模块, 进行了算法的推导和优化, 按照数字设计流程完成了相应算法的硬件设计, 最终通过了在 FPGA 开发板上的实际场景测试。在本文研究的基础上, 后续可以进行如下的改进和扩展:

1) 本研究中, 设计的同步模块实际测试方法是通过两块开发板相隔一段距离进行通信测试, 实际车联网场景下车辆与车辆之间往往会处于高速相对运动的情况, 因此后续可以继续研究在高速场景下的频偏和噪声对同步性能的影响。

2) 解速率匹配硬件设计结构仍有继续优化的空间, 可以考虑并行计算和串行计算和乒乓结构, 进一步提高吞吐量。针对 368 种不同的信道配置, 可以继续完善验证的平台, 规范化验证的流程, 从而提高验证的效率。

3) 设计的 CORDIC 和 FFT/IFFT 模块只针对 LTE-V2X 的特定场景进行了设计, 可配置性仍不如赛灵思提供的 IP 核, 例如 CORDIC 模块的旋转因子精度和 FFT 模块中的量化位宽还不能自由配置, 可以合理设计灵活的结构提高模块的可配置性, 在计算延时、硬件资源消耗和功耗方面仍有可以优化的空间。

参考文献

- [1] Hu P, Schmitt R, Schwarzer J, et al. Transportation statistics annual report 2021[J]. Transportation Statistics Annual Report (TSAR), 2021(27): 1-2.
- [2] Araniti G, Campolo C, Condoluci M, et al. LTE for vehicular networking: a survey[J]. IEEE Communications Magazine, 2013, 51(5): 148-157.
- [3] Ersal T, Kolmanovsky I, Masoud N, et al. Connected and automated road vehicles: state of the art and future challenges[J]. Vehicle System Dynamics, 2020, 58(5): 672-704.
- [4] Yun D S, Lee S J. A study on the architecture of the in-vehicle wireless sensor network system[C]//2013 International Conference on Connected Vehicles and Expo (ICCVE). IEEE, 2013: 826-827.
- [5] Murali A, Bhanupriya K, Smitha S B, et al. Performance evaluation of IEEE 802.11 p for vehicular traffic congestion control[C]//2011 11th International Conference on ITS Telecommunications. IEEE, 2011: 732-737.
- [6] 陈山枝, 胡金玲, 时岩等. LTE-V2X 车联网技术、标准与应用[J]. 电信科学, 2018, 34(04): 1-11.
- [7] Karagiannis G, Altintas O, Ekici E, et al. Vehicular networking: A survey and tutorial on requirements, architectures, challenges, standards and solutions[J]. IEEE Communications Surveys & Tutorials, 2011, 13(4): 584-616.
- [8] Naik G, Choudhury B, Park J M. IEEE 802.11 bd & 5G NR V2X: Evolution of radio access technologies for V2X communications[J]. IEEE Access, 2019, 7: 70169-70184.
- [9] Eiza M H, Owens T, Ni Q, et al. Situation-aware QoS routing algorithm for vehicular ad hoc networks[J]. IEEE Transactions on Vehicular Technology, 2015, 64(12): 5520-5535.
- [10] Twardokus G, Rahbari H. Towards protecting 5G sidelink scheduling in C-V2X against intelligent DoS attacks[J]. IEEE Transactions on Wireless Communications, 2023: 1-2.
- [11] Liu J, Liu J. Intelligent and connected vehicles: Current situation, future directions, and challenges[J]. IEEE Communications Standards Magazine, 2018, 2(3): 59-65.
- [12] MacHardy Z, Khan A, Obana K, et al. V2X access technologies: Regulation, research, and remaining challenges[J]. IEEE Communications Surveys & Tutorials, 2018, 20(3): 1858-1877.

- [13] Zhou H, Xu W, Chen J, et al. Evolutionary V2X technologies toward the Internet of vehicles: Challenges and opportunities[J]. *Proceedings of the IEEE*, 2020, 108(2): 308-323.
- [14] 3GPP TS 36.211, Evolved Universal Terrestrial Radio Access (E-UTRA); Physical channels and modulation[S], TS 36.211 V14.2.0(2017-04), 3rd Generation Partnership Project, 2017.
- [15] Miller H J, Shaw S L. Geographic information systems for transportation: principles and applications[M]. Oxford University Press on Demand, 2001.
- [16] Ahmad A, Riedl A, Naramore W J, et al. Comparative study of security in IEEE 802.11-2007 and IEEE 802.15. 4-2006 for patient monitoring environments[C]//2010 Seventh International Conference on Information Technology: New Generations. IEEE, 2010: 914-918.
- [17] Federal Communications Commission. Dedicated short range communications report and order[J]. *R&O FCC*, 2003: 03-324.
- [18] Huang C L, Sengupta R, Krishnan H, et al. Implementation and evaluation of scalable vehicle-to-vehicle safety communication control[J]. *IEEE Communications Magazine*, 2011, 49(11): 134-141.
- [19] Chen X, Yao D. An empirically comparative analysis of 802.11 n and 802.11 p performances in CVIS[C]//2012 12th International Conference on ITS Telecommunications. IEEE, 2012: 848-851.
- [20] Tseng Y L. LTE-advanced enhancement for vehicular communication[J]. *IEEE Wireless Communications*, 2015, 22(6): 4-7.
- [21] Gallo L, Härri J. Short paper: A LTE-direct broadcast mechanism for periodic vehicular safety communications[C]//2013 IEEE Vehicular Networking Conference. IEEE, 2013: 166-169.
- [22] Saifuddin M, Zaman M, Toghi B, et al. Performance analysis of cellular-V2X with adaptive & selective power control[C]//2020 IEEE 3rd Connected and Automated Vehicles Symposium (CAVS). IEEE, 2020: 1-7.
- [23] Cordero C. Optimizing 5G for V2X—Requirements, implications and challenges[J]. *IEEE VTC Mission-Critical 5G for Vehicle IoT*, 2016: 1-14.
- [24] Gyawali S, Xu S, Qian Y, et al. Challenges and solutions for cellular based V2X communications[J]. *IEEE Communications Surveys & Tutorials*, 2020, 23(1): 222-255.
- [25] Chen S, Hu J, Zhao L, et al. Spectrum needs and planning[M]//Cellular Vehicle-to-Everything

- (C-V2X). Singapore: Springer Nature Singapore, 2023: 309-328.
- [26] Cheng N, Lu N, Zhang N, et al. Vehicular WiFi offloading: Challenges and solutions[J]. Vehicular Communications, 2014, 1(1): 13-21.
- [27] Myung H G, Goodman D J. Single carrier FDMA: a new air interface for long term evolution[M]. John Wiley & Sons, 2008.
- [28] Dalakas V, Mathiopoulos P T, Di Cecca F, et al. A comparative study between SC-FDMA and OFDMA schemes for satellite uplinks[J]. IEEE Transactions on Broadcasting, 2012, 58(3): 370-378.
- [29] Navita A, Amandeep N. Performance analysis of OFDMA, MIMO and SC-FDMA technology in 4G LTE networks[C]//Proc. 6th Int. Conf.-Cloud Syst. Big Data Eng. 2016: 554-558.
- [30] Myung H G, Lim J, Goodman D J. Single carrier FDMA for uplink wireless transmission[J]. IEEE Vehicular Technology Magazine, 2006, 1(3): 30-38.
- [31] Roy J S, Mishra S. Performance of SC-FDMA for LTE uplink under different modulation schemes[C]//2019 International Conference on Mechatronics, Robotics and Systems Engineering (MoRSE). IEEE, 2019: 202-206.
- [32] LTE for UMTS: OFDMA and SC-FDMA based radio access[M]. John Wiley & Sons, 2009.
- [33] Cho Y S, Kim J, Yang W Y, et al. MIMO-OFDM wireless communications with MATLAB[M]. John Wiley & Sons, 2010.
- [34] 3GPP TS 36.300, Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN); Overall description; Stage 2[S], TS 36.300 V14.2.0(2017-04), 3rd Generation Partnership Project, 2017.
- [35] Budhdev N, Chan M C, Mitra T. Pr 3: Power efficient and low latency baseband processing for lte femtocells[C]//IEEE INFOCOM 2018-IEEE Conference on Computer Communications. IEEE, 2018: 2357-2365.
- [36] Xu W, Zhang Z, Tong J, et al. FPGA Acceleration Architecture Design and Implementation of LTE Physical Downlink Shared Channel[C]//2020 IEEE 6th International Conference on Computer and Communications (ICCC). IEEE, 2020: 31-36.
- [37] Li C M, Shu Y L, Wang P. Synchronization for LTE-A system downlink: A proposed novel method used at LTE wireless communications[J]. IEEE Consumer Electronics Magazine,

2019, 8(3): 40-43.

- [38] Nassralla M H, Ayoub H, Akl N, et al. Multiplication-Free detection algorithm of the primary synchronization signal in LTE[C]//2020 28th European Signal Processing Conference (EUSIPCO). IEEE, 2021: 1772-1776.
- [39] Kim D, Jeong T, Chung E Y. Resource utilization optimized design method for matched filter of pss searcher[C]//2020 International SoC Design Conference (ISOCC). IEEE, 2020: 290-291.
- [40] Chung M K, Liu L, Edfors O, et al. Robust timing synchronization for full duplex communications: Design and implementation[C]//2017 IEEE Global Conference on Signal and Information Processing (GlobalSIP). IEEE, 2017: 883-887.
- [41] Filipovic N, El Mezeni D, Radošević A. Hardware Implementation of 5G NR Deinterleaver and De-rate Matcher[C]//2021 15th International Conference on Advanced Technologies, Systems and Services in Telecommunications (TELSIKS). IEEE, 2021: 57-60.
- [42] 郭杰. 查表法实现 LTE 中 Turbo 码解速率匹配[J]. 数字通信世界, 2017, No.145(01): 19-22.
- [43] Guanghui Z, Kai N, Lifeng H, et al. A method of optimizing the de-Rate Matching and demodulation in LTE based on GPP[C]//7th International Conference on Communications and Networking in China. IEEE, 2012: 828-832.
- [44] Bukris M, Gazit I. Rate Matching and De-Rate Matching for an LTE Transport Channel: U.S. Patent Application 12/403,621[P]. 2010-9-16.
- [45] He S, Hu Q, Zhang H. Implementation of rate matching with low latency and little memory for LTE turbo code[J]. Journal of Information & Computational Science, 2013, 10(13): 4117-4125.
- [46] Nimbalker A, Blankenship Y W, Classon B K. Method and apparatus for rate matching within a communication system: U.S. Patent 8,520,496[P]. 2013-8-27.
- [47] Volder J E. The CORDIC trigonometric computing technique[J]. IRE Transactions on Electronic Computers, 1959 (3): 330-334.
- [48] Volder J E. The birth of CORDIC[J]. Journal of VLSI signal processing systems for signal, Image and Video Technology, 2000, 25(2): 101-105.

- [49] Meher P K, Valls J, Juang T B, et al. 50 years of CORDIC: Algorithms, architectures, and applications[J]. IEEE Transactions on Circuits and Systems I: Regular Papers, 2009, 56(9): 1893-1907.
- [50] Aggarwal S, Meher P K, Khare K. Concept, design, and implementation of reconfigurable CORDIC[J]. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2015, 24(4): 1588-1592.
- [51] Van Loan C. Computational frameworks for the fast Fourier transform[M]. Society for Industrial and Applied Mathematics, 1992.
- [52] Cooley J W, Tukey J W. An algorithm for the machine calculation of complex Fourier series[J]. Mathematics of Computation, 1965, 19(90): 297-301.
- [53] Xilinx. FFT Intellectual Property[EB/OL].<https://www.xilinx.com/products/intellectual-property/fft.html>.
- [54] Lalwani V, Munir S. Area Efficient VLSI Architecture for Reversible Radix-2 FFT Algorithm using Folding Technique and Reversible Gate[C]//2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI). IEEE, 2019: 1429-1433.
- [55] Akkad G, Mansour A, ElHassan B, et al. Fft radix-2 and radix-4 fpga acceleration techniques using hls and hdl for digital communication systems[C]//2018 IEEE International Multidisciplinary Conference on Engineering Technology (IMCET). IEEE, 2018: 1-5.
- [56] Manir S B, Rahman M, Ahmed T. Comparison between FDD and TDD frame structure in SC-FDMA[C]//2012 International Conference on Informatics, Electronics & Vision (ICIEV). IEEE, 2012: 795-799.
- [57] 李伟. 浅谈无线移动通信 LTE TDD 与 LTE FDD 技术简介和比较[J]. 中国新通信, 2014, 16(18): 38-39.
- [58] 杨健, 李方村. FDD LTE 与 TDD LTE 技术对标研究[J]. 电信工程技术与标准化, 2015, 28(05): 66-69.
- [59] 延凯悦. 基于 LTE-V2X 城市道路场景的资源分配算法改进[J]. 邮电设计技术, 2019, (06): 36-41.
- [60] Griffith D W, Cintrón F J, Rouil R A. Physical sidelink control channel (PSCCH) in mode 2: Performance analysis[C]//2017 IEEE International Conference on Communications (ICC).

- IEEE, 2017: 1-7.
- [61] Griffith D, Cintron F, Galazka A, et al. Modeling and simulation analysis of the physical sidelink shared channel (PSSCH)[C]//2018 IEEE International Conference on Communications (ICC). IEEE, 2018: 1-7.
- [62] Mori K, Kishikawa H, Goto N. Modulation format conversion from QPSK to 16QAM using delay line interferometer and spectral shaping filter[C]//2016 21st OptoElectronics and Communications Conference (OECC) Held Jointly with 2016 International Conference on Photonics in Switching (PS). IEEE, 2016: 1-3.
- [63] 何荣森, 何希顺, 张跃. 从 ARM 体系看嵌入式处理器的发展[J]. 微电子学与计算机, 2002(05): 42-45.
- [64] Qaqos N. Optimized FPGA implementation of the CRC using parallel pipelining architecture[C]//2019 International Conference on Advanced Science and Engineering (ICOASE). IEEE, 2019: 46-51.
- [65] Yu L, Wang X, Liu J. An improved rate matching algorithm for 3GPP LTE turbo code[C]//2011 Third International Conference on Communications and Mobile Computing. IEEE, 2011: 345-348.
- [66] Warty C, Mattigiri S, Gambi E, et al. De bruijn sequences for secure scrambling at long term evolution-advanced physical layer[C]//2013 IEEE Global Communications Conference (GLOBECOM). IEEE, 2013: 4044-4049.
- [67] MathWorks. MATLAB: High-Level Language and Development Environment[EB/OL]. [2023-03-06]. <https://ww2.mathworks.cn/products/matlab.html>.
- [68] Xilinx. Vivado Design Suite - Xilinx[EB/OL]. [2023-03-06]. <https://china.Xilinx.com/products/design-tools/vivado.html>.
- [69] Cummings C. SystemVerilog's priority & Unique-A Solution to Verilog's full_case & parallel_case Evil Twins[J]. SNUG. Israel, 2005.
- [70] Uryvsky L, Moshynska A, Osypchuk S, et al. Features of the OFDM technology usage as a means of fighting against multipath effects in a wireless communication channel[C]//2021 IEEE International Conference on Information and Telecommunication Technologies and Radio Electronics (UkrMiCo). IEEE, 2021: 142-145.

- [71] 刘战存. 多普勒和多普勒效应的起源[J]. 物理, 2003, 32(07): 488-491.
- [72] Zepernick H J, Finger A. Pseudo random signal processing: theory and application[M]. John Wiley & Sons, 2013.
- [73] Chu D. Polyphase codes with good periodic correlation properties[J]. IEEE Transactions on Information Theory, 1972, 18(4): 531-532.
- [74] de Figueiredo F A P, Lenzi K G, José Filho A B, et al. LTE-Advanced channel coding generic procedures A high-level model to guide low-level implementations[C]//2013 Wireless Telecommunications Symposium (WTS). IEEE, 2013: 1-7.
- [75] Puzyrev P I, Semenov K V, Zavyalov S A. Spurious-free dynamic range of cordic based digital quadrature demodulator[C]//19th International Conference of Young Specialists on Micro/Nanotechnologies and Electron Devices (EDM). IEEE, 2018: 167-171.
- [76] Masram B Y, Karule P T. Design and implementation of 111MHz frequency compression efficient CORDIC based 2D-DCT using FPGA and its power performance[C]//2021 IEEE Bombay Section Signature Conference (IBSSC). IEEE, 2021: 1-6.
- [77] Rajeswari L M, Manocha S K. Design of data adaptive IFFT/FFT block for OFDM system[C]//2011 Annual IEEE India Conference. IEEE, 2011: 1-5.
- [78] Anbarasan A, Shankar K. Design and implementation of low power FFT/IFFT processor for wireless communication[C]//International Conference on Pattern Recognition, Informatics and Medical Engineering (PRIME-2012). IEEE, 2012: 152-155.
- [79] Tsiropoulou E, Kapoukakis A, Papavassiliou S. Uplink resource allocation in SC-FDMA wireless networks: A survey and taxonomy[J]. Computer Networks, 2016, 96: 1-28.
- [80] de Temino L A M R, Berardinelli G, Frattasi S, et al. Channel-aware scheduling algorithms for SC-FDMA in LTE uplink[C]//2008 IEEE 19th International Symposium on Personal, Indoor and Mobile Radio Communications. IEEE, 2008: 1-6.

作者在攻读硕士学位期间公开发表的论文

- [1] XXX, XXX, XXX A Resource- and Power-Efficient Implementation of PSS Synchronization on FPGA in Vehicular Networks[C]//IoT as a Service: 8th EAI International Conference, IoTaaS 2022, Manchester, Great Britain, November 17-18, 2022. (已录用, 未出版) (第一作者)
- [2] XXX, XX, XX, XXX 在车联网中基于 FPGA 的 PSS 同步方法的实现[J]. 工业控制计算机, 2023 年第 11 期 (11 月 25 日出版). (第一作者)
- [3] XXX, XX, XX, XXX, XXX 基于 FPGA 的 LTE-V 收端同步的方法及系统[P]. 中国专利申请中, 专利申请号: 202111208064.7, 专利申请日期: 2021 年 10 月. (第一发明人)
- [4] XXX, XX, XXX, XXX, XX 基于 FPGA 的 LTE-V 收端 CCH 信道解速率匹配的系统及方法[P]. 中国专利申请中, 专利申请号: 202210180518.2, 专利申请日期: 2022 年 2 月. (第一发明人)

作者在攻读硕士学位期间参与的项目

- [1] 国家重点研发计划政府间国际科技创新合作项目“基于边缘智能的车联网研究”
(编号 XXXXXXXXXXXXXXXX)

致 谢

时光荏苒，不知不觉我已经在上海大学学习和生活了七年。七年的时光给我留下了丰富有趣的回忆，期间转过专业，换过校区，见过东区凌晨的日出，也看过校内天台的晚霞，认识了许许多多有趣的朋友。

首先，我想感谢上海大学给我提供了自由广阔的平台，可以选择自己感兴趣的专业和方向，丰富的校园活动和比赛机会使我认识了很多优秀的同学。

其次，我想感谢我的导师 XXX 和项目组的 XXX，XXX 对学生温柔且有耐心，在硕士期间在科研项目中给予了我很多帮助。本科期间我只对 FPGA 有初步的了解，进入实验室后在导师的指导下慢慢接触了很多 IC 相关的项目，对 IC 设计有了更全面的了解和认识。项目组的 XXX 和蔼可亲，对项目组的学生在学习和生活上都很关心，在我们的项目中负责上层的协议和算法的指导，合理地统筹各方面的工作，平时也不断在学习 IC 相关的知识，也参与过数字模块的编写设计，XXX 严谨的科研态度和不断学习的习惯值得我们学习。

接着，我想感谢实验室中的师兄和同学还有学弟学妹们。XX，XXX 和 XX 学长在我的科研过程中给了我很多帮助，耐心地并且毫无保留地和我分析相关的知识经验，在项目中遇到困难的时候给了我很多技术方面的支持和思维方面的启发。此外，我也想感谢实验室的 XXX，XXX，XXX，XX，XX，XXX 同学，他们在平时也给予了我很多帮助，使我顺利地完成了相关的项目。

然后，我想感谢我的家人，他们一直都很支持我的各种决定，生活也一直给我充分的经济支持，使我可以顺利完成七年的学业。此外我还想感谢我的女朋友 XXX，硕士开学第一个月就在一起并且一直走到现在，度过了快乐的大学时光。

最后特别感谢百忙之中抽出宝贵时间参加我论文评审和答辩的老师，感谢你们对我论文的批评和指导。