



沈阳航空航天大学

# 工程硕士学位论文

论文题目 基于 FPGA 实现 JPEG-LS 无损  
压缩算法的研究

研究生 李长兴

工程领域 电子与通信工程

研究方向 嵌入式系统及应用

指导教师(学校) 张雷 副教授

指导教师(企业) 申洋 高工 北京中科远卓

二零一八年三月

分类号\_\_\_\_\_

密级\_\_\_\_\_公开\_\_\_\_\_

UDC\_\_\_\_\_

# 学 位 论 文

## 基于 FPGA 实现 JPEG-LS 无损压缩算法的研究

研 究 生 姓 名 : 李长兴

学科专业名称 : 电子与通信工程

研 究 方 向 : 嵌入式系统及应用

论 文 类 型 : 应用研究

申请学位类别 : 工 程 硕 士

指导教师(学校) : 张雷 副教授 沈阳航空航天大学

指导教师(企业) : 申洋 高工 北京中科远卓

论文提交日期 : 2018 年 01 月 10 日

论文答辩日期 : 2018 年 03 月 10 日

沈 阳 航 空 航 天 大 学

2018 年 03 月

**SHENYANG AEROSPACE UNIVERSITY**

**THESIS FOR MASTER'S DEGREE**

**RESEARCH OF JPEG - LS LOSSLESS  
COMPRESSION ALGORITHM BASED ON  
FPGA**

Candidate: ChangXing Li

Supervisor: Lei Zhang

Specialty: Electronic and Communication

Engineering

Date: March,2018



## 原创性声明

本人郑重声明：所呈交的学位论文是本人在导师的指导下独立完成的。除文中已经注明引用的内容外，本论文不包含其他个人或集体已经发表或撰写过的作品或成果，也不包含本人为获得其他学位而使用过的成果。对本文研究做出重要贡献的个人或集体均已在论文中进行了说明并表示谢意。本声明的法律后果由本人承担。

论文作者签名：李长兴

2018年3月18日



## 摘 要

JPEG-LS 是在 ISO/ITU 的新标准中用于对静态连续色调图像进行无损或近无损压缩的一种算法。本文在 JPEG-LS 无损压缩算法理论研究的基础上, 提出基于 FPGA 实现 Cameralink 图像采集处理系统的设计。由于 Cameralink 传输图像数据的可靠性高, 传输速率快, 因此采用 Cameralink 接口协议来传输图像数据, 解决了图像数据输出和采集之间的速度匹配问题。本文实现了基于 FPGA 的 CameraLink 图像采集与处理系统设计, 该系统主要由图像激励源单元和 Cameralink 图像采集处理单元组成。而 FPGA 实现 JPEG-LS 无损压缩算法是该系统图像处理重要组成部分。

本文重点研究了基于 FPGA 实现 JPEG-LS 无损压缩算法的设计。主要采用 VHDL 语言的模块化设计实现上下文建模、预测编码和 Golomb 编码, 并且在上下文建模设计中做了改进处理, 以及在预测误差计算上对预测校正值  $C[Q]$  的更新进行了改进处理。从而减少了逻辑资源的占用, 使得编码器时钟频率达到 41MHz, 由此提高了压缩效率, 缩短了编码时间。最后, 用 16bit 随机图像数据进行了综合仿真验证, 实现了编码器对图像数据的无损压缩。结果表明解压出来的图像数据和原图像数据对比没有差别。另经大量测试, 验证了编码的正确性和鲁棒性, 并且高效率完成了图像无损压缩, 使得各项指标均达到了既定要求。

此外, 本文对图像采集处理系统的主要单元进行了功能测试, 其中图像激励源单元可以按照不同模式发送不同类型的图像数据, 而 Cameralink 图像采集处理单元能同时对两路 Cameralink 图像数据进行采集并进行 JPEG-LS 无损压缩。结果表明, 该系统达到了图像的实时采集与无损压缩的效果。

**关键词:** JPEG-LS; Cameralink; FPGA; 无损压缩

## Abstract

JPEG-LS is an algorithm for the static continuous tone image lossless or near lossless compression in the new ISO/ITU standard. Based on the theoretical research of JPEG-LS lossless compression algorithm, this thesis presents the design of Cameralink image acquisition and processing system based on FPGA. Due to the high reliability and fast transmission speed of Cameralink transmission image data, the Cameralink interface protocol is used to transmit image data, which can solve the speed matching problem between image data output and acquisition. The thesis implements the design of CameraLink image acquisition and processing system based on FPGA, which mainly combines image excitation source unit with Cameralink image acquisition and processing unit. The implementation of JPEG-LS lossless compression algorithm by FPGA is an important part of the system image processing.

This thesis, based on FPGA, focuses on the realization of JPEG-LS lossless compression algorithm. It mainly using the modular design of VHDL language to achieve context modeling, predictive coding and Golomb coding, and improves the context modeling design. Moreover, it improves the update process of predictive correction  $C[Q]$  on the prediction error calculation, which can reduce the occupation of logic resources, and improve the encoder clock frequency reaches 41 MHz that improving the compression efficiency and shortening the encoding time. Finally, the 16bit random image data is used to carry out a comprehensive simulation verification, and realize the lossless compression of the image data. The results show that there is no difference in the contrast between the image data and the original image data. After a large number of tests, the correctness and robustness of the coding are verified, and the image lossless compression is accomplished efficiently, which makes all the indexes meet the established requirements.

In addition, the main units of the image acquisition and processing system are tested, in which the image excitation source unit can send different types of image data according to different modes. The Cameralink image acquisition and processing unit can collect two Cameralink image data at the same time and carry out JPEG-LS lossless compression. The results show that the system achieves the effect of real-time image acquisition and lossless compression.

**Keywords:** JPEG-LS; Cameralink; FPGA; lossless compression

## 目 录

摘 要 .....	I
<u>ABSTRACT</u> .....	II
第 1 章 绪论 .....	1
1.1 研究背景和意义 .....	1
1.2 国内外研究现状 .....	2
1.2.1 JPEG-LS 发展现状 .....	2
1.2.2 FPGA 发展现状 .....	3
1.3 FPGA 设计流程 .....	4
1.4 本文研究内容及章节安排 .....	5
第 2 章 JPEG-LS 图像压缩编码基本理论 .....	6
2.1 编码模型 .....	6
2.1.1 理想信源的无失真编码 .....	6
2.1.2 图像信源的建模与编码 .....	7
2.2 几种图像压缩算法介绍 .....	9
2.2.1 DPCM 压缩算法 .....	9
2.2.2 CCSDS 压缩算法 .....	10
2.2.3 JPEG2000 压缩算法 .....	12
2.3 JPEG-LS 图像压缩编码 .....	13
2.3.1 Golomb 编码 .....	14
2.3.2 游长编码 .....	16
2.4 本章小结 .....	17
第 3 章 Cameralink 图像采集处理系统设计 .....	18
3.1 Cameralink 介绍 .....	18
3.2 系统要求 .....	19
3.3 系统设计 .....	19
3.4 图像激励源单元设计 .....	20
3.4.1 接口设计 .....	21
3.4.2 图像激励源板卡设计 .....	22
3.4.3 图像激励源的 FPGA 软件功能设计 .....	22
3.5 Cameralink 图像采集处理单元设计 .....	26



3.5.1 缓存模块设计 .....	28
3.5.2 配置电路及 JTAG 设计 .....	29
3.5.3 时钟设计 .....	31
3.5.4 SATA IP 核设计 .....	31
3.6 性能分析 .....	33
3.7 本章小结 .....	35
<b>第 4 章 JPEG-LS 无损压缩算法设计 .....</b>	<b>36</b>
4.1 JPEG-LS 无损压缩算法设计 .....	36
4.1.1 上下文建模设计 .....	39
4.1.2 预测模块设计 .....	40
4.1.3 Golomb 编码模块设计 .....	45
4.2 仿真与验证 .....	47
4.3 本章小结 .....	50
<b>结论 .....</b>	<b>51</b>
<b>附录 I .....</b>	<b>52</b>
<b>参考文献 .....</b>	<b>53</b>
<b>致谢 .....</b>	<b>55</b>
<b>攻读硕士期间发表（含录用）的学术论文 .....</b>	<b>56</b>

# 第1章 绪论

## 1.1 研究背景和意义

全球信息化技术发展迅速,日常生活已经非常依赖网络以及信息。在快速发展的时代,信息处理必须满足快速、便捷、方便的前提条件,图像处理正是信息处理的典范。针对现在的信息化技术基本条件,对于数据量极大的数据处理工作,数据处理的速度并不那么容易提高,技术难度极大,困难重重,数字图像正是其中让大多数科研人员重点研究的对象。数字图像处理的研究领域中,压缩处理被广泛运用,它的关键性指标在于不仅能对图像进行压缩,而且原始图像仍能保持很好的画面品质,同时保证较小的存储、传输量。鉴于很多重要的应用场合图像不易得到或者应用特殊,静态图像在所占存储空间比视频图像高很多,电磁波段更广,更宽的灰度级和较为丰富的色彩。但它的数量一直大幅度增长中。运用图像数据压缩编码技术,传输数据量可以得到极大的减少,空间和时间耗费都大大降低。这种技术能使通信质量得到很好的保证,同时提高了通信速度。有用数据和冗余数据共同构成图像信源的数据。图像压缩的最终目的就是尽最大的能力去去掉冗余信息,在不损坏图像质量的前提下,原始数据用比较少的数据量无损地代替。

数据压缩<sup>[1,2]</sup>就是以较少的数据表示信源和受信者之间的信息传递,减少信息集合在传输时所占信道带宽和不必要的延时。信息的数据压缩在某种程度上指某信号集合所占用的空域空间、时域空间和频域空间。较少的存储空间将让其占用的带宽得到很大的降低,并且传输效率能得到提高。这就是说,只要采用某种方法来减少某一种空间,就可以压缩数据。数字压缩技术的优点有很多,但也大大的增加了数据量。从传输角度看:采样率低的数字电话,通常码率需要 64Kb/s;而实时传输一路高清晰度视频,传输码率可高达 1990Mb/s。因此数据压缩的给社会带来的经济效益将越来越显著。针对这些情况,尤其对数字图像的研究也逐渐变成热点。对于连续色调的静止图像压缩有 JPEG、JPEG-LS、JPEG2000 等国际标准<sup>[3-5]</sup>。

JPEG-LS 追求一种更加低复杂度的无损或者近似无损的图像压缩标准<sup>[6,7]</sup>,这样就可以比无损 JPEG 压缩效率更好。它能够得以发展是因为基于 Huffman 编码<sup>[8]</sup>的无损 JPEG 压缩或者其他压缩算法其压缩效果不理想,且去相关不彻底。由于 JPEG-LS 只有相对比较简单 Golomb 编码和不使用 DCT 变换,因而在图像无损压缩领域中得到了广泛的应用且易于硬件设计和实现。JPEG-LS 的核心是 LOCO-I 算法<sup>[9,10]</sup>,预测,建立误差模型,基于上下文误差编码是该算法的核心所在。除了无损压缩, JPEG-LS 也有近无损压缩,该近无损压缩的最大编码误差值可以由编码参数控制。JPEG-LS 压缩效率比

JPEG2000 高, 且压缩效果比无损的 JPEG 标准压缩要好。

近年来, 随着对电子系统的吞吐量要求的日益提高, FPGA 技术在当前的电子设计领域越来越火热<sup>[11,12]</sup>。作为现今主流的可编程器件, 它给电子系统带来不可限量的速度和带宽。充足的片上可编程资源、灵活的设计思路、小型性和逐渐成熟的 EDA 设计工具, 在 FPGA 上进行图像处理经典算法的硬件实现是非常方便的。所以 JPEG-LS 无损压缩算法编码的 FPGA 设计实现将会遇到巨大的机遇与挑战。

## 1.2 国内外研究现状

### 1.2.1 JPEG-LS 发展现状

图像编码的发展至今已走过了半个多世纪的历程。ISO 和国际电联制定了适用于不同类型图像的压缩编码标准。在 1998 年 JPEG-LS 成为静止图像压缩的国际压缩标准, 并逐渐取代 JPEG 的无损压缩部分。该标准的第一部分是在 1999 年完成的, 第二部分是在 2003 对外公开的, 且引入了算法编码。

由于压缩性能优越, JPEG-LS 已经在国外很多领域得到了广泛的应用。2000 年, M. Weinberger 等人提出 LOCO-I 无损图像压缩算法应用于 JPEG-LS 编码的原理和标准化<sup>[13]</sup>。2001 年 Shantanu D.Rane 提出了新的无损和近无损受控的静止图像压缩标准用于压缩高分辨率高程数据的 JPEG-LS<sup>[14]</sup>。2002 年 Savakis A 等人提出 JPEG-LS 的基准测试和硬件实现<sup>[15]</sup>。2003 年 D.Brunello 等人提出使用帧间信息进行 JPEG-LS 无损压缩视频方法<sup>[16]</sup>。2005 年 L.-J.Kau 等人提出了使用最小二乘法优化前向边缘检测的自适应无损编码<sup>[17]</sup>。2007 年 M.Papadonikolakis 等人提出高性能的无损压缩编码器的实现由一个高效的流水线 JPEG-LS 编码器组成<sup>[18]</sup>。2008 年 A.P.Kakarountas 等人提出高效率和性能较高的 15 位 JPEG-LS 编码器的实现<sup>[19]</sup>。2009 年 Pierantonio Merlino 等人提出针对卫星遥感领域采用流水线结构充分利用 JPEG-LS 算法的顺序特性易于硬件实现的无损和近无损压缩的实现<sup>[20]</sup>。2011 年 S.Wahl 等人提出了包括游程模式的 JPEG-LS 无损压缩通过延迟更新过程来提出对 JPEG-LS 上下文更新的去相关, 以达到保证并行度, 对压缩率的影响可以忽略不计<sup>[21]</sup>。2012 年 Z.Wang 等人提出了提出了一种基于中值边缘预测器和 JPEG-LS 的 Golomb 编码器的快速无损图像压缩(FLIC)算法<sup>[22]</sup>。2015 年 Sheorey S 等人提出了基于平面拟合的无损图像压缩的新的预测技术, 主要根据 JPEG-LS 上下文为每个像素选择最佳预测器采用自适应模型切换算法<sup>[23]</sup>。2016 年 Fidha Nazar 等人提出了基于 LOCO-I 的 JPEG-LS 图像压缩算法在 MATLAB 中设计和实现<sup>[49]</sup>。除此之外, JPEG-LS 在国外很多领域得到了广泛应用。

JPEG-LS 的发展在国内也得到很大的发展和应用。在卫星遥感领域应用较为突出, 如 2001 年, 北京航空航天大学程子敬对遥感图像高保真高速实时 JPEG-LS 压缩技术研究<sup>[24]</sup>。2002 年曹青, 吴乐南提出了利用 JPEG-LS 高效无损压缩气象卫星云图数据<sup>[25]</sup>。2

2003 年清华大学电子工程系吴美建和林行刚提出了一种改进的遥感图像无损压缩 JPEG-LS 算法<sup>[26]</sup>, 为适应遥感图像较高倍率无损压缩的需要,改进了 JPEG-LS 算法,该算法首先通过放宽游程检测门限,并通过引入局部梯度控制下的预测来增加平均游程长度,以提高压缩比。2007 年,北京工业大学王晋等人提出一种基于自适应预测的高光谱图像近无损压缩方法<sup>[27]</sup>, 使用三维自适应预测有效地去除高光谱图像的空间和谱间相关性;然后对预测误差进行量化,以进一步降低编码率。2008 年同济大学徐燕凌等人提出对 JPEG-LS 图像压缩方案的编码特性进行数据分析,提出一阶码率控制策略与二阶码率控制策略,有效地实现了近无损图像编码过程中的码率动态调整。2009 年中科院沈洪亮等人提出基于 JPEG-LS 的遥感图像无损压缩技术。2010 年中科院刘嘉晗等人提出了高性能全流水线可控参数 JPEG-LS 编码器实现。2011 年西安电子科技大学郝勇峥提出了基于 JPEG-LS 算法的星载图像压缩系统设计。2013 年牛慧卓等人 JPEG-LS 图像压缩算法的研究与实现。2014 年西安电子科技大学综合业务网国家重点实验室张毅等人提出一种新的基于先验数据表的 JPEG-LS 动态码率控制算法。2015 年海南大学陈益刚等人提出了基于 JPEG-LS 压缩比控制的图像压缩加密算法。2016 年中国飞行试验研究院韩涛等人改进的 JPEG-LS 编码方法在飞行试验中的应用。国内还有很多科研机构对 JPEG-LS 图像压缩都做出了很大贡献, 这里不在过多的介绍。

### 1.2.2 FPGA 发展现状

现场可编程逻辑门阵列简称为 FPGA 是一种让开发设计能够在现场对定制数字逻辑进行编程的集成电路<sup>[28]</sup>。FPGA 早在 20 世纪 80 年代就已出现,当时只是为了让所有开发设计者能够制作定制逻辑。早期的 FPGA 电路体积庞大,无法安装到单个芯片上。只能简单的搭建一个 FPGA 能够对该接口重新编程和改造。不久,就能够利用 FPGA 搭建整个子系统,不再受限于仅使用 ASIC 实施子系统。由于电路组件的体积越来越小,能够将越来越多的设备放置到同一芯片上,从而可以实现更加复杂的功能和更高速度的运算,这反过来又提高了计算速度,降低了功耗。如果在设计中使用 FPGA,即便只是执行一些简单功能,也须做大量编程工作。因此,开发设计大都避免使用 FPGA。FPGA 已经从有用但简陋的接口设备,发展成为系统级集成电路,拥有微处理器、存储块和接口等。与其他硬件搭建方式相比, FPGA 有两大优势。首先, FPGA 能够搭建需要的硬件,不再需要耗时、昂贵的专用应用集成电路(ASIC)并承担相关风险。另处, FPGA 能够进行定制,在 FPGA 中的运算往往比在微处理器内核里更加简单、快速和节能。换句话说, FPGA 提供了设计灵活性,它在不引入大量成本,并且不带来设计延期风险的前提下,能够修改系统组件的功能。

现代 FPGA 综合使用可配置的静态随机存取存储器(SRAM)、高速输入/输出管脚(I/O)、逻辑块和布线资源。具体而言, FPGA 包括可编程逻辑元件, 这些元件被称为逻

辑元件；还包括由可重新配置的内部连线组成的分层结构，内部连线使得逻辑元件相互之间能够实现物理连接。能够通过配置，让逻辑元件完成复杂的功能，或者简单执行逻辑门的工作。多数 FPGA 还包括存储块。随着技术的高速发展，IP 核组件能够内嵌到 FPGA 结构中，它们在提供丰富的功能性的同时，也减少了功耗，降低了成本。目前纳入 FPGA 的部分硬 IP 组件有：存储块、计算电路、收发器、协议控制器，甚至还有中央处理单元(CPU)。不过，这些 IP 核组件的定制程度并不像 FPGA 的其余部分。通过集成 IP 核组件来实现像数字信号处理器(DSP)<sup>[29]</sup>设计在将这些常用组件添加到系统时，可以省去额外开发一个 DSP IP 核的时间。FPGA 生产能够随手将硬 IP 组件并入 FPGA，因为这些硬 IP 组件的功能已经成为不可缺少的一部分，在多数电子系统上都相当统一。

### 1.3 FPGA 设计流程

FPGA 系统设计流程分为定义需求、创建架构、实现设计和设计验证等几个步骤，实现设计和验证流程如图 1.1 所示。系统设计流程，首先，需要定义需求，按照定义创建系统架构。在这里，决定需要的设计实现组件。然后，利用已经规划的架构实现系统。最后，验证系统是否满足全部需求。在 HDL 语言设计编写完成之后，下一步是编译 HDL 设计。在 FPGA 编程里，综合工具将 HDL 语言设计作为输入内容，将其转化成由逻辑门、寄存器和导线组成的网络，并且这个网络被配置成实现 HDL 语言描述的功能。然后，将通过其他工序，挑选将在 FPGA 中使用的特定逻辑门、寄存器和导线，并且创建一个编程文件，该编程文件将在 FPGA 启动后，对 FPGA 进行配置。FPGA 的编程环节包括：确定程序设计模块，选择一种硬件描述语言(HDL)，在文本编辑器里编写代码，设计综合，对设计布局布线，然后将设计载入 FPGA。在设计载入 FPGA 之后，需要一个调试周期，以修复功能性错误。调试指消除程序的 Bug，直至整个设计按照规定功能正常运行。

在完成代码编译之后，需进行测试，然后才可将代码配置到 FPGA。过去，面对的可编程逻辑芯片要简单，在测试时，只是验证能否正常工作。但是，由于现代 FPGA 的复杂性，早期的调试手段已不再适用。FPGA 设计调试通常在模拟激励源环境下进行。这种应用程序通常可以激励设计输入的行为。激励过程通过软件实现，并且通过软件能够看到每个寄存器的表现。然后，再将的设计下载到 FPGA。代码的调试和验证通常反复进行，直至非常确信 HDL 代码按照预想设计运行。通常使用“Testbench”的工具，来验证 FPGA 在实际电路能否正常运行。Testbench 是将软件仿真与实际硬件相结合，构成了实际的系统模型，该系统中包含了 FPGA。FPGA 大都包含数以万计或十万计的逻辑门，因此不可能全部测试。Testbench 主要用于测试主要设计区域中实际工作的逻辑门。激励环境可帮助隔离某些特定设计区域，以及在这些区域添加调试辅助工具，让设计能够按照设计思路运行。仿真验证就是采用 Testbench 和仿真软件实现的。

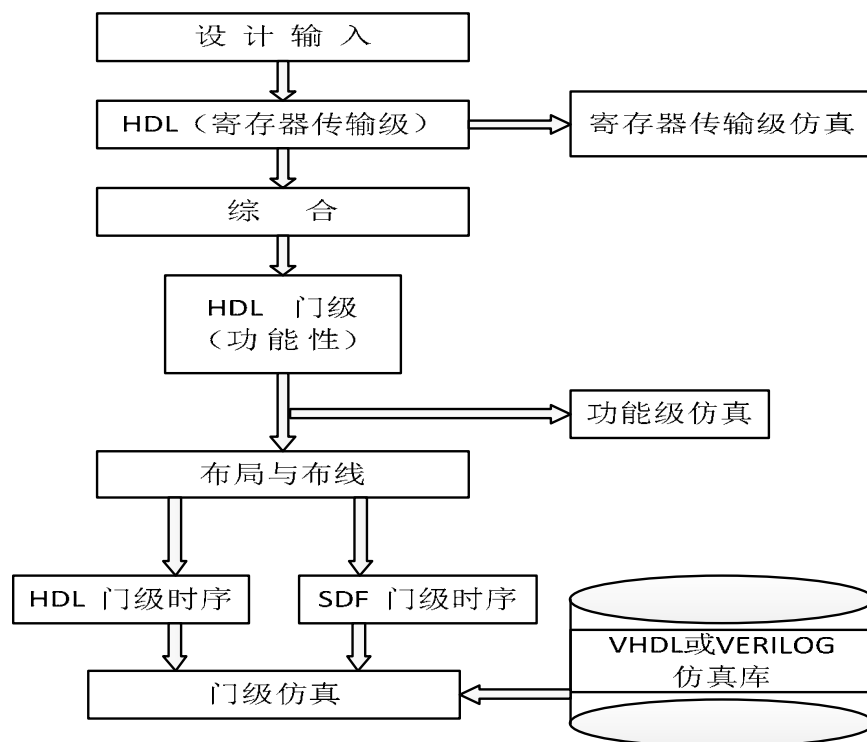


图 1.1 FPGA 设计实现和验证流程图

#### 1.4 本文研究内容及章节安排

本文提出一种基于 FPGA 实现 JPEG-LS 无损压缩算法的研究，该算法具有低复杂度、效率高和 VHDL 语言易实现等特点。由于应用于卫星遥感领域的图像处理需要对图像实时无损压缩。本文采用 FPGA 硬件平台设计了 JPEG-LS 无损图像压缩算法。首先，第一章介绍本论文方向的研究背景及意义以及 FPGA 设计基本流程。第二章首先压缩编码的模型针对这一模型介绍了几个压缩编码 DPCM、CCSDS 和 JPEG2000 的原理介绍，重点研究了 JPEG-LS 无损压缩编码 Golomb 编码和游程编码理论。第三章设计了基于 FPGA 实现 Cameralink 图像采集处理系统，系统主要包括图像激励源单元和 Cameralink 图像采集处理单元的设计和验证。验证结果表明系统的稳定性和可靠性高。第四章介绍了 FPGA 实现 JPEG-LS 无损压缩编码器的上下文建模设计、预测设计和 Golomb 编码设计以及仿真验证编码的正确性和实时性。第五章对本论文工作的总结与展望。



## 第 2 章 JPEG-LS 图像压缩编码基本理论

本章先对图像压缩编码模型进行研究,对理想信源的无失真编码和图像信源的建模与编码进行了详细介绍。针对这一模型介绍了几种压缩编码的算法,分别是 DPCM 压缩算法、CCSDS 压缩算法和 JPEG2000 压缩算法,对其优缺点进行比较。在分析几种无损压缩基础上选择 JPEG-LS 图像压缩算法,主要因为 JPEG-LS 无损压缩算法压缩效率高、编码处理速度快且易于硬件设计实现等特点,因此,选择了 JPEG-LS 无损压缩算法进行全面的理论研究。之后对 JPEG-LS 无损压缩编码理论并详细介绍了 Golomb 编码、游长编码等理论。

### 2.1 编码模型

#### 2.1.1 理想信源的无失真编码

由信息理论可知,信源输出的信息符号序列具有随机性,根据已知的信源符号概率分布就能确定编码可能达到的码率下界。对于离散信源包括离散有记忆信源和离散无记忆信源。

对于离散无记忆信源<sup>[30]</sup>,将信源抽象的描述为信源 X 从 K 个符号集中  $\{a_1, a_2, \dots, a_k\}$  发出符号序列,在 N 时刻发出的消息符号  $u_N$  与前面发出的消息符号  $u_1, u_2 \dots u_{N-1}$  无关。对无记忆信源编码时,可以用完备的离散型概率空间来描述,其主要特点是无记忆和离散。无失真编码码率的下界是该信源的一阶熵  $H_1$ ,即每个信源符号的平均码字长度。在编码变字长编码 VLC。其原理是使得最终的平均码长最短,使用短码形式对概率较高的信源符号进行编码,长码用于小概率信源符号。不等长码也称为非长码采用前缀码为了使不等长码字符符合唯一可译性和实时解码的要求,其构造特征中的任何一个码字是没有任何其他符号代码的前缀。构造特征的特点是由没有其他符号的前缀的码字构成。设  $\bar{L}$  是不等长码的平均码长,那么下面的不等式成立:

$$H_1 \leq \bar{L} < H_1 + 1 \quad (2.1)$$

其中,一阶熵  $H_1$  为:

$$H_1 = -\sum_i P(a_i) \log P(a_i) \quad (2.2)$$

上式中,符号  $a_i$  的概率是由信源  $P(a_i)$  发出的。在编码系统中,无失真(无损)编码也被称为熵编码。这表明平均码长不会超过最大信源熵 1 比特。

对于离散有记忆信源,情况更为复杂,信源 X 由 K 个符号集  $\{a_1, a_2, \dots, a_k\}$  发出一个

符号序列,在时刻  $N$  发出消息符号  $u_N$  与前面的消息符号  $u_{-m} \cdots u_1 \cdots u_k$  的情况有关。目前,假定是一个马尔可夫处理过程,只考虑相对简单的平稳随机过程。 $N-1$  阶的马尔可夫过程定义如下:信源发出任何一符号  $u_N$ ,它与前面发出的  $N-1$  个符号  $u_1, \cdots, u_{N-1}$  有关,用条件概率表示如下:  $P(u_N | u_1, u_2 \cdots u_{N-1}) = P(u_N | u_{-m} \cdots u_1 \cdots u_{N-1})$ 。

这样,每一种  $N-1$  个符号的组成代表编码中一个可能遇到的信源状态  $s_j$ ,也称为上下文(context)。信源发出一个符号  $u_N$  后,就从当前状态进入下一个新状态。每个信源符号的平均条件熵为  $H(U_N | U_1 \cdots U_{N-1})$ :

$$H(U_N | U_1 \cdots U_{N-1}) = \sum_{u_1 \cdots u_N} P(u_1 \cdots u_N) \log P(u_N | u_1 \cdots u_N) \quad (2.3)$$

对于马尔可夫信源的编码,可以使用前面所述的无记忆信源的熵编码方法,对每个状态  $s_j$  分别进行变字长编码,及对信源在每个状态  $s_j$  可能发出的符号  $u_n$  用码长为  $n_j$  的不等长编码。与式(2.1)相类似,在状态  $s_j$  的平均码长  $\bar{n}_j$  满足:

$$H(u_N | S_j) \leq \bar{n}_j < H(u_N | S_j) + 1 \quad (2.4)$$

对所有状态进行平均后,可知 Markov 过程的平均码长  $\bar{n}$  满足以下不等式:

$$H(U_N | U_1 \cdots U_{N-1}) \leq \bar{n} < H(U_N | U_1 \cdots U_{N-1}) + 1 \quad (2.5)$$

由于  $N-1$  阶 Markov 过程有  $J = K^{N-1}$  个状态,因此为实现上述编码需要  $J$  个变字长码表。这在实用中有很大的难度。例如  $K=2$ ,一般只对一些特殊的编码技术场合才采用。

### 2.1.2 图像信源的建模与编码

在进行图像的压缩之前,首先选择图像的格式,数字图像处理一般选择灰度图像进行算法处理,图像中的像素一般使用二维数组表示,假设处理的图像的像素的数据深度为  $P$ ,灰度等级根据图像的亮度分量分为 255 不同的量级,因此处理的图像的灰度等级为  $K = 2^P$ ,那么在信源所表示的符号集便可以表示出来,把图像作为一个独立的信源进行变长的熵编码。便可以将数字图像的下限码率调整为一阶熵。

实际在工程中使用的图像信源是“有记忆的”,上述编码处理方法的编码效率相对明显就不是很高了,主要是由于图像中有许多部分具有相似的灰度值。解决这个问题的办法是将图像源转换成另一个具有由数学过程设置的新符号的源。符号转换的过程称为建模。建模的目的是最小化源内符号的相关性,使其接近独立的源特征。编码率的下限是源的一阶熵。为了提高图像的编码效率,可以降低图像的相关系数,在去相关的过程中,即使发生可逆的变化,信息熵并不会发生变化。那么上述这一个处理过程就称之为

无失真编码或者无损编码。图 2.1 所示详细的描述了无损编码的具体处理流程，在编写编码模块时，需要将算法分解为不同的功能模块，然后对各个子模块进行编码开发，也可以通过相应的算法处理，将信源根据不同的特性分为不同的模块，之后对各模块进行相应的算法处理和图像编码。此外，还有一种建模的情况是新的信源是一个有记忆的过程，但是符号集的符号个数  $K$  较小（通常为 2），记忆的长度为  $N$  也较小（最多不超过 3），此时，熵编码考虑的下界就是条件熵，而不是一阶熵。总之，熵编码是信源编码的最后一步，在此之前，要通过各种变换或建模使编码符号尽量满足熵编码的要求。

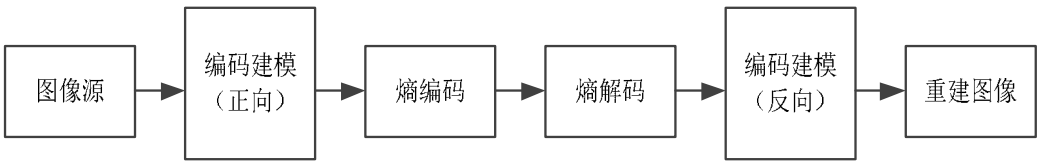


图 2.1 无损编码过程

由于无损编码后，处理前后的图像质量并不发生变化，加大了压缩编码的实现过程的开发难度，在视觉观测图像时，观测到的图像并不全是有用信息，在进行图像压缩时，可以将无效的图像信息人为进行删除，提取出所需图像信息进行数据压缩，这样压缩的数据量便大大减少，可以加快图像的压缩效率。在工程开发中大多使用该方法进行数据压缩，处理后的结果没有丢失关键的图像信息，理论上称为有损压缩。图 2.2 为有损编码过程的示意方框图。其中  $Q$  表示量化过程，通常使用均匀量化， $IQ$  表示反量化。量化是编码中唯一引入失真的地方，前提是不考虑计算误差的影响。

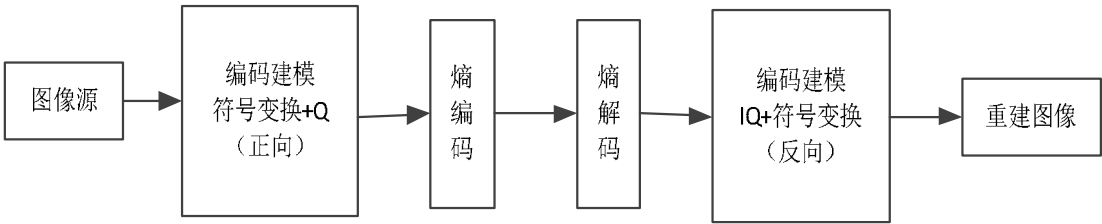


图 2.2 有损编码过程

最后，对于彩色图像的编码，其转换部分是独立于编码过程，可以用灰度图像的方法分别对各个彩色分量进行编码。为了获得高效的编码，可能有必要在编码之前执行一些彩色分量变换以去除各个分量之间的相关性。彩色转换部分不属于编码过程。在后续介绍的编码方法中，将只考虑灰度图像的编码处理。在后续中将以灰度图像编码处理作为主要介绍对象。

## 2.2 几种图像压缩算法介绍

本节主要介绍了几种常见的压缩算法简单的进行研究和比较，主要包括 DPCM 压缩算法、CCSDS 压缩算法和 JPEG2000 压缩算法。主要分析了它们各自的优缺点和实现的复杂度等。由于 JPEG-LS 无损压缩算法的硬件的实现复杂度较低、保真度高、压缩效率高和性能也是相差不大甚至在某些图像压缩领域还要比这几种算法要好的原因，所以本文综合考虑采用 JPEG-LS 无损压缩算法进行 FPGA 设计和实现。

### 2.2.1 DPCM 压缩算法

差分脉码调制简称 DPCM 是一种非常著名的图像压缩技术，由于其简单、易于实现，在图像处理中应用非常广泛。在应用中，为了达到更好的压缩比<sup>[31]</sup>，使用有损数据压缩方法比较多。

DPCM 是一种预测编码方法。在预测编码中，相关性理论用于预测相邻像素。基本 DPCM 的框图表示如图 2.3 所示。图中原始图像表示硬件的输入和被编码图像。预测因子是从紧密相关的像素中定义的，而这个预测因子是从实际的像素值中减去得到的。由图中的  $e(x,y)$  表示原始图像的差分图像。然后这个差分图像经历编码过程，这将有效地减小图像的大小<sup>[32,33]</sup>。

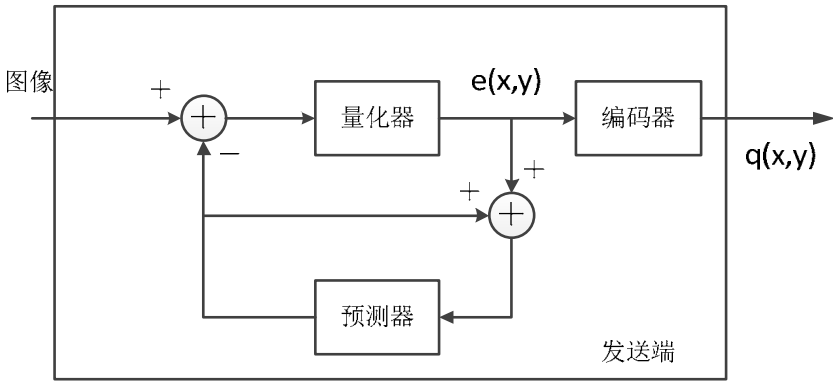


图 2.3 DPCM 编码器框图

编码是通过查找表(LUT)来完成的<sup>[34]</sup>。LUT 的设计是通过对不同类型图像的不同观察来完成的。在 LUT 设计阶段定义的映射范围将决定优先级的对比度。LUT 映射的过程是比较差值像素值  $e(x,y)$  与查找表的差异像素范围，并映射到相应的编码值。在接收端，可以使用相同的 LUT 从编码值构造解码值<sup>[48]</sup>。对于一般图像，DPCM 利用图像之间的相关性进行压缩，可以提高压缩比，有些图像 DPCM 压缩反而不如直接进行 huffman 后的压缩比高。

在预测编码中，图像样本  $x[n]$  被变换成等同于预测误差  $e[n]$  的阵列。采样这样的编码方法是因为  $x[n]$  和  $e[n]$  是等价的序列并且具有相同的采样率，所以它们具有相同的熵

率  $H(\{X[n]\}) = H(\{E[n]\})$ 。但是, 在面对实际的限制时,  $e[n]$  通常更容易有效地进行编码。

$$e[n] = x[n] - u_p(x_{\delta+n}) \quad (2.6)$$

这将构成预测残差序列。其中  $\delta+n$  是因果邻域,  $u_p(\cdot)$  是元素已被编码的预测矢量  $x_{\delta+n}$  的函数。通过理想化预测函数  $u_p$ , 可以使残差熵  $H^{(1)}(\{E[n]\}) = H(E)$  最小化。在实际中, 通常可以相当于通过设置预测值等于条件均值来最小化残差方差, 即

$$u_p(x_{\delta+n}) \approx E[X(n) | X_{\delta+n} = x_{\delta+n}] \quad (2.7)$$

如果可以找到一个好的预测器, 大多数残差将会接近于零。

举个例子, 设  $\delta = \{[0, -1], [-1, 0]\}$  因此靠近  $x[n]$  的上边和左边的样本的采样值构成的预测矢量, 也就是:

$$x_{\delta+n} = (x[n_1, n_2 - 1], x[n_1 - 1, n_2])' \quad (2.8)$$

早期的 JPEG 图像压缩标准定义了这种无损压缩算法。这是早期的无损压缩算法支持的预测器之一。

## 2.2.2 CCSDS 压缩算法

数据压缩对太空任务的好处是提高收集科学数据的能力, 减少机载存储和遥测带宽需求。由于这些好处, 空间数据系统协商委员会(CCSDS)一直在为空间应用推荐数据压缩标准。1997 年通过的第一个 CCSDS 数据压缩建议标准化了无损 Rice 压缩算法的一个版本<sup>[35]</sup>。太空任务从太空探测到近地天文台就是得益于这一建议。1998 年, CCSDS 数据压缩工作组开始评估建立适合于星载应用的图像压缩建议的可行性。在算法选择过程中最大的考虑就是优化率失真性能。执行渐进压缩的能力被认为是非常理想的特征, 但不是强制性的。

CCSDS122.0-B-1 是由 CCSDS 委员会在 2005 年开发的基于小波的图像压缩标准, 目的是用于空间系统和应用。CCSDS122.0-B-1 与现有的基于小波的图像压缩标准如 JPEG2000<sup>[36]</sup>和 DPCM 相比, 其特点是计算复杂度和存储器要求更低, 使其更适合于空间应用。具体而言, 这些功能使其可以轻松高效地在航天器上使用。此外, CCSDS122.0-B-1 支持有损和无损压缩, 并配备了先进的方法来精确控制压缩率。它还包括支持无限像高的图像传感器的固有手段。最后, 该建议允许在不使用外部存储器的情况下实现 ASIC 和 FPGA。本节所研究的 CCSDS 122.0-B-1 蓝皮书就是为压缩二维数字图像而推出的建议标准, 其主要目的是通过压缩图像数据, 减少下行信道带宽占用率, 节省缓存资源的消耗, 提高数据传输效率。CCSDS 122.0-B-1 图像压缩标准主要由两个

部分构成，分别是离散小波变换(DWT)和比特面编码。离散小波变换对图像进行去相关处理，比特面编码对去相关后的小波系数进行扫描、熵编码，完成压缩操作，系统结构如图 2.4 所示。这种一般的图像压缩方法被广泛使用，参见例如参考文献<sup>[36,37]</sup>。

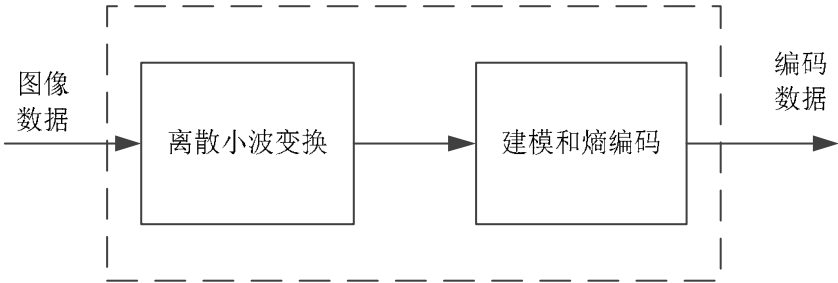


图 2.4 CCSDS 算法组成框图

CCSDS-122.0-B-1 标准采用的图像变换是二维(2-D)三级(L3)DWT。具体而言，该标准定义了两个一维变换，即使用两套 9/7DWT 滤波器。第一组使用浮点 DWT 系数并用于有损压缩，而第二组使用整数 DWT 系数，并用于有损和无损压缩。二维小波变换通过对二维输入图像的每个维度分别应用单个小波变换来实现。最初，DWT 被水平逐行然后垂直逐列应用。图 2.5 所示中描绘的小波变换的输出被分成四个子带，它们是 LL、HL、LH 和 HH 子带。如果对 LL<sub>i</sub> 子带重复该过程，则产生 2 级系数，可以通过用 2 级系数代替图 2.5(b)中的 LL<sub>i</sub> 子带来将其可视化。换句话说，LL<sub>i</sub> 子带被 LL<sub>2</sub>、HL<sub>2</sub>、LH<sub>2</sub> 和 HH<sub>2</sub> 取代如图 2.5(c)。最后，以相同的方式产生 3 级系数如图 2.5(d)。在多级二维 DWT 的每个阶段，LL 子带的尺寸减半；因此，3 级子带的尺寸是 H/8 和 W/8。在进行编码之前，DWT 系数必须被四舍五入为整数。

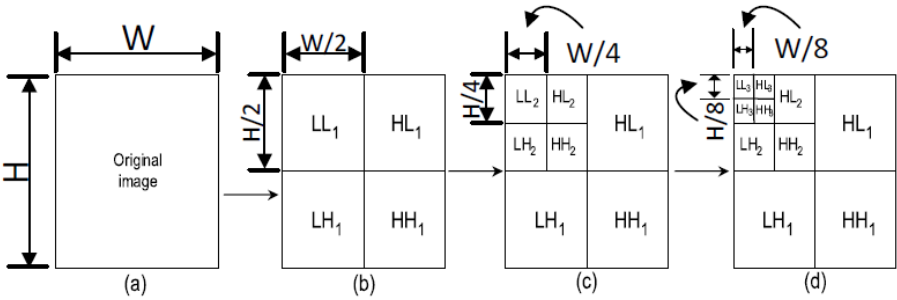


图 2.5 二维三级 DWT 示意图

在 DWT 完成之后，产生的 DWT 系数被给到建模器和熵编码部分，在其中将它们分组以形成 DWT 系数的块。如图 2.6 所示每个块由单个 DC 系数和 63 个 AC 系数组成。而且，每个系数块跨越所有子带和级别，这些松散地对应于原始图像中的局部区域。具



体而言，块的 64 个系数跨越如下：在等级 1 的每个子带中，将 48 个系数（图 2.6 中的孙系数）跨越四行，将 12 个系数（图 2.6 中的子系数）跨越两行级别 2 的每个子带，以及最终 4 个系数（图 2.6 中的父系数节点和 DC 系数）在每个级别 3 的子带中跨越一行。

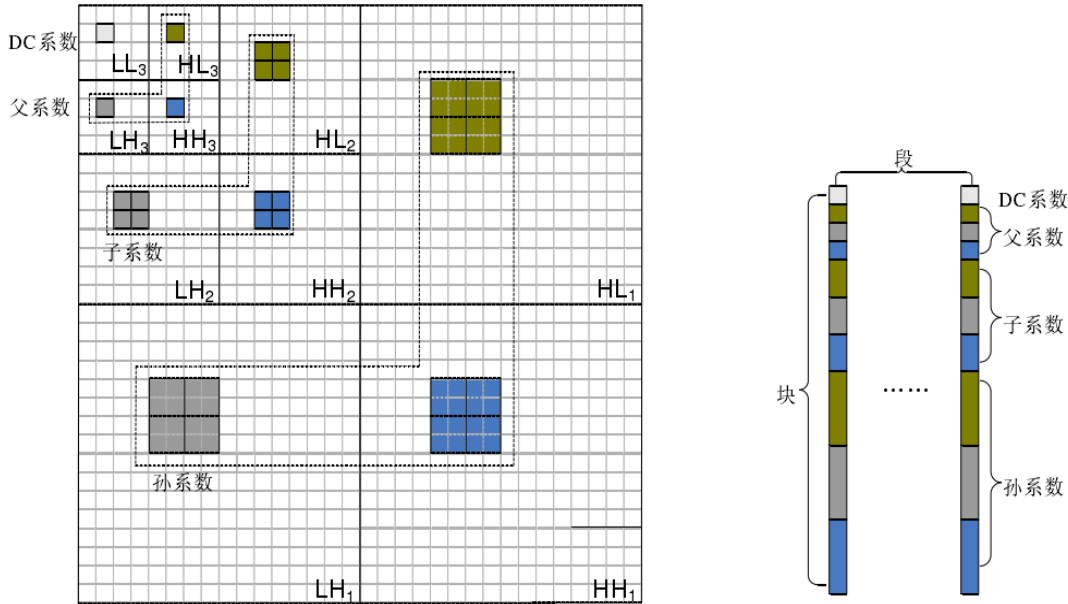


图 2.6 小波系数分块图

最后，将连续的系数块组合在一起以形成具有 16 个块的最小大小的块的段。在每个段内，比特面编码器在比特面中处理 DWT 块完成数据压缩。CCSDS 算法适用于各种成像仪器和需要立即处理数据的传感器。

### 2.2.3 JPEG2000 压缩算法

JPEG2000 图像压缩标准的完整描述可以在文件 ISO/IEC15444-1 或 ITU-T T.800 [5] 中找到，与 JPEG 编码标准相比，JPEG2000 图像编码标准提供了更好的压缩性能。JPEG2000 的一个关键特性就是用灵活的比特流表示适合在不同的环境中传输各种格式的图像，包括具有可选错误恢复工具（标准中不包括附加信道编码）的易出错的环境。它还允许使用其可伸缩性特征（分辨率，质量，位置和图像分量）以及在编码器或解码器级别考虑的感兴趣区域特征来描述图像的不同表示。应用这个功能可以从任何 JPEG-2000 压缩的源图像中控制或传输任何目标设备的基本信息。数据流可以适应用户终端的能力，并提供交互式解码机制。标准中已经有几个部分，为构建数字图像应用带来了必要的技术。如图 2.7 所示为 JPEG2000 的编码模块组成包括预处理、DWT、均匀量、自适应算术编码和码流组织等五个模块。

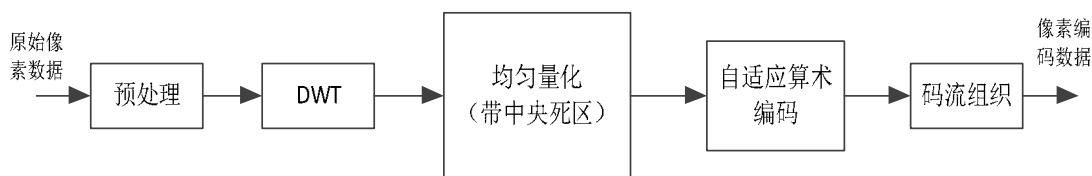


图 2.7 JPEG2000 基本编码模块组成

为了解决和改进 JPEG 标准中的不足之处，引入了一种新的 JPEG2000 图像压缩方案。JPEG2000 支持丰富的功能包括改进的压缩效率，无损编码，分辨率和失真(SNR)可扩展性，感兴趣区域编码以及对压缩图像的图像编辑的支持<sup>[38]</sup>。JPEG2000 与最初的 JPEG 标准有着本质的区别。首先，JPEG2000 用离散小波变换(DWT)代替 JPEG 的离散余弦变换(DCT)。DWT 具有多分辨率的图像表示功能，它允许嵌入比特流内的分辨率可扩展性。另外，DWT 比 DCT 具有出色的压缩效率更高质量的图像压缩效果，从而 DWT 通常应用于图像拼接或整个图像上。这种大规模的应用使得 DWT 可以将 JPEG 中的  $8 \times 8$  的 DCT 的块效应最小化。DWT 有着出色的编码效率得益于它在大范围内去除图像的相关性。

### 2.3 JPEG-LS 图像压缩编码

JPEG-LS 无损图像压缩算法正常模式下图像数据经过采用基于上下文建模、自适应预测和误差编码<sup>[40]</sup>实现整个编码过程，而误差编码采用 Golomb 编码实现。正常模式编码用于图像中的复杂区域，否则使用游程模式编码。JPEG-LS 无损压缩编码框图如图 2.8 所示。JPEG-LS 设计简单高效，压缩编码和硬件易于实现等等。可以实现近无损压缩和无损两种编码模式：正常模式和游程模式。对于正常模式，JPEG-LS 使用当前图像像素及其相邻像素，并其相邻像素用作当前像素上下文。上下文用于预测像素以及编码模式的选择等。在预测编码完成后计算当前像素的预测误差。根据上下文特征，JPEG-LS 对预测误差编码采用特殊 Golomb 编码<sup>[39]</sup>的受限长码字来进行。对于游程模式，编码计数器统计具有相同灰度值的图像像素，并对相同灰度值的像素进行游程长度编码。

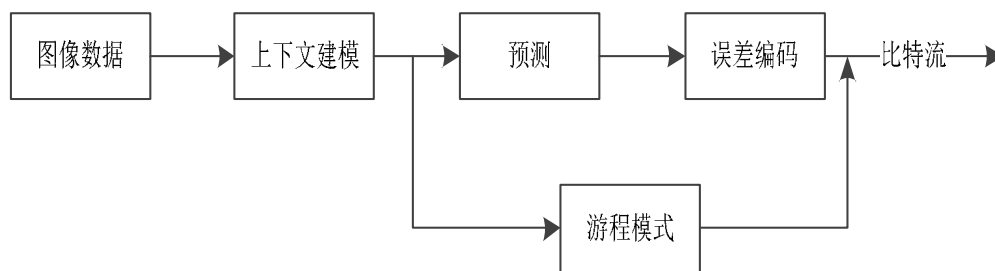


图 2.8 JPEG-LS 压缩编码框图

在正常编码模式中，使用简单的边缘检测器来确定初始预测值。然后，通过结合上下文自适应预测校正值得到预测值。预测误差概率分布满足双边几何分布(TSGD)<sup>[41]</sup>。对预测误差必须使用编码参数  $k$ ，它的值可以通过使用上下文模型编码来获得。熵编码器完成 Golomb 编码的快速一元编码时就是编码参数  $k$  值移位来实现的。当编码器进入游长编码模式时，必须先完成游长计数后，熵编码器采用效率更高的游长编码来完成图像无损压缩编码的。

### 2.3.1 Golomb 编码

数据压缩在减少数据冗余度的同时，也降低了传输的可靠性，使数据更容易出错。变长码最明显的缺点就是对差错的敏感性。码字的前缀性会导致 1 位出错就可能使解码器失去同步，甚至使解码器继续读入、解码和解释剩下的压缩流，不会判断数据是错误的。自适应 Huffman 编码试图使码字能够与信源符号的实际出现频率动态匹配，虽然使编码器和解码器的复杂化为代价，但却使编码器更容易受到损害。因为 Huffman 码表<sup>[2]</sup>自动建立的正确性可能受到误码的威胁。如果不对各种信源符号的实际出现频率都能最佳匹配，就可能基于某个预先假定的概率模型设计出最佳变长编码，使之在与信源符号真是概率模型失配不多的前提下，简化最佳变长编码器的设计，提高解码器的可靠性。

Golomb 编码就是具有这种潜力的一类唯一可译码，已经为许多图像、视频压缩编码所采用。本文 JPEG-LS 压缩算法的常规编码就是采用 Golomb 编码。由 S.W.Golomb 于 1966 年提出的编码方法可以使得服从几何分布的正整数数据流的平均码长最小化，并且可以不使用 Huffman 编码算法而直接给出最佳可变长度码。但需要给出满足

$$(1-p)^b + (1-p)^{b+1} \leq 1 < (1-p)^{b-1} + (1-p)^b, 0 \leq p \leq 1 \quad (2.9)$$

的  $b$  值（一定存在）。我们接下来就可以计算出数据流中的整数  $n$  出现概率为

$$p(n) = (1-p)^{n-1} p, 0 \leq p \leq 1 \quad (2.10)$$

$N$  的哥伦布码也由前缀码和尾码组成；

前缀码是  $q+1$  个比特的一个一元码字，而

$$q = \text{INT}\left(\frac{n-1}{b}\right) \quad (2.11)$$

是  $(n-1)/b$  的整数部分；

尾码是对  $(n-1)/b$  的余数

$$r = n - 1 - qb \quad (2.12)$$

的二进制编码，位数为  $INT[\log_2 b + 0.5]$ ，其中  $INT[\bullet]$  表示取整。

假设给出一个正整数  $n$  和正整除数  $m$ （大于 0），此时  $n$  对于  $m$  的 Golomb 编码表示成  $G_m(n)$ ，而一元编码与  $n \bmod m$  的二进制综合表示成  $\lfloor n/m \rfloor$ ， $G_m(n)$  的构建如下：编码形成商  $\lfloor n/m \rfloor$  的一元编码（整数  $q$  的一元编码定义为  $q$  个 1 后跟一个 0）。令  $k = \lceil \log_2 m \rceil$ ， $c = 2^k - m$ ， $r = n \bmod m$ ，并计算截短的余数  $r'$ ，例如使其满足

$$r' = \begin{cases} r \text{截短至 } k-1, 0 \leq r < c \\ r+c \text{截短至 } k \text{ 比特}, \text{ else} \end{cases} \quad (2.13)$$

例如，为计算  $G_4(9)$ ，可从求商  $\lfloor 9/4 \rfloor = 2$  的一元编码开始，它的一元码是 110。然后，令  $k = \lceil \log_2 4 \rceil = 2$ ， $c = 4 - 4 = 0$ ， $r = 9 \bmod 4$ ，在二进制中是  $1001 \bmod 0100$  或 0001。根据式(2.13)， $r'$  是  $r$  的截短到 2 比特的结果，它是 01。最后得到的结果就是 11001。

对于  $m = 2^k$ ， $c = 0$  的特殊情况，对所有的  $n$ ，在式(2.13)中， $r' = r = n \bmod m$  截短至  $k$  比特。产生 Golomb 编码所要求的除法变成二进制移位操作和计算上更简单的编码，这种编码称为 Golomb-Rice 码。表 2.1 中列出了 10 个非负整数的  $G_1$ ， $G_2$  和  $G_4$  码。因为在每种情况下  $m$  都是 2 的幂，所以它们也是前三个 Golomb-Rice 码。此外， $G_1$  是非负整数的一元码，因为对所有的  $n$ ，有  $\lfloor n/1 \rfloor = n$  和  $n \bmod 1 = 0$ 。

表 2.1 整数 0~9 的几种 Golomb 码

N	$G_1(n)$	$G_2(n)$	$G_4(n)$	$G_{\text{exp}}^0(n)$
0	0	00	000	0
1	10	01	001	100
2	110	100	010	101
3	1110	101	011	11000
4	11110	1100	1000	11001
5	111110	1101	1001	11010
6	1111110	11100	1010	11011
7	11111110	11101	1011	1110000
8	111111110	111100	11000	1110001
9	1111111110	111101	11001	1110010

Golomb 码只能用于表示非负整数，并且有许多 Golomb 码可供选择，在其有效应用中的一个关键步骤就是除  $m$  的选择。当被表示的整数具有质量函数（PMF）的几何分

布时，由式(2.10)就可以证明 Golomb 码是最佳的，在该意义上，即当

$$m = \left\lceil \frac{\log_2(1+p)}{\log_2(1/p)} \right\rceil \quad (2.14)$$

时， $G_m(n)$ 为所有唯一可判读的编码提供了最短的平均码长。

### 2.3.2 游长编码

游程编码(RLC, Run Length Coding)属于无损压缩编码，是一种统计编码。游程编码由字符(或信号取样值)构成的数据流中各个字符重复出现而形成的字符的长度。如果给定元素编码的码字，我们知道码字的长度和码字的位置就可以恢复原始数据流。游程长度编码(RLC)是一种使用二进制代码来提供此信息的方法。

基本的 RLC 方法最初需要加一个异字头的前缀，因而低效且不实用。但是，对于二值图像和连续色调图像，该前缀可以省去。因此，改进的 RLC 在图像编码中得到广泛的应用。

游程长度编码(run-length code)是栅格数据压缩的重要编码方法，其基本思想是：其实图像的本质在编码时可以看做是多维矩阵，对于一个栅格图像，往往在行（或列）方向上相邻的几个元素点具有相同的属性码字，可以采取一些方法来压缩这些重复的记录。编码方案是数据每行（或列）的代码变化时才压缩数据，然后依次记录代码和相同代码的重复次数。在诸如传真的二进制图像中，每条扫描线总是由许多连续的黑色像素用二进制表示的“1”和连接的白色像素二进制表示“0”组成，这些像素分别变成图 2.9 所示的“黑色运行”和“白色运行”。巡回长度编码是这些运行长度的 VLC 熵编码<sup>[41]</sup>。在解码之后，根据巡回的长时间值恢复黑白运行，重建原始的黑白图像。对比图 2.1 的无损编码模型，可见其在模型中符号转换相关的编码是把物理量的“黑”、“白”电平转换为游程长度。

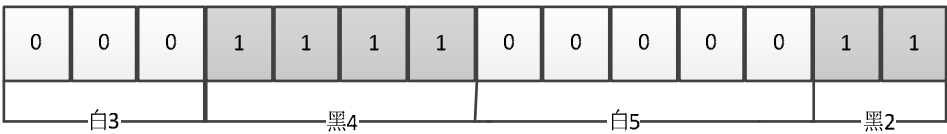


图 2.9 黑白游程示例

游程编码除了用于二值图像，也用在多值对象的编码上。在这一点上，符号变换产生一个符号对：其中一个是相同幅度的连续像素的数量，也就是游程长度；另一个是与幅度有关的量，这对于不同应用场景是不同的。然后符号进行 VLC 熵编码得到编码数据。

## 2.4 本章小结

本章首先介绍了静止图像压缩编码的编码模型以及理想信源的无失真编码和进行了介绍。其次，本章将几种图像压缩算法的基本理论和方法作比较简单的介绍以及对各压缩编码的利弊做了比较。主要介绍了几种常见的无损压缩算法简单的进行研究和比较，包括 DPCM 压缩算法、CCSDS 压缩算法和 JPEG2000 压缩算法。主要分析了它们各自的优缺点和实现的复杂度等。最后，详细介绍了 JPEG-LS 图像压缩算法的原理包括上下文建模、预测编码、Golomb 编码在 JPEG-LS 编码的应用以及游程编码等。为第四章 FPGA 设计 JPEG-LS 压缩算法作了重要的理论基础和铺垫。



### 第3章 Cameralink 图像采集处理系统设计

本章首先对 Cameralink 协议<sup>[43]</sup>进行了简单的介绍。然后对系统进行了需求分析后做出了针对该需求的系统详细设计。该系统的图像激励源单元和图像采集处理单元的硬件电路和软件设计具体包括：FPGA 电路设计、图像数据缓存设计、图像数据输出设计和接口控制设计等。最后针对 Cameralink 图像采集处理系统进行功能验证，验证结果表明系统可靠性高，处理速度快等特点。

#### 3.1 Cameralink 介绍

图像采集技术经过多年的发展，控制接口也得到不断的更新，Channellink 是最早出现的控制接口，通过在接口扩展新的传输控制信号，并修改相关的传输协议，美国自动化工业学会 AIA 发布了新的接口协议，Cameralink 可以在高速数字电路中进行数据传输，并且扩展了接口配置，针对不同的数据传输量选择不同的数据传输方式进行传输，为相机的数据接口选择提供更多地余地，可以通过不同的配置方式和输入端口的连接进行不同速度的数据传输。

图像激励源由 DS90CR285 发射器将 28 位 CMOS/TTL 数据转换为 4 个 LVDS（低电压差信号）数据流。锁相传输时钟并行传输通过第五个 LVDS 连接的数据流。传输时钟 28 位输入数据的每个周期都被取样和传输。

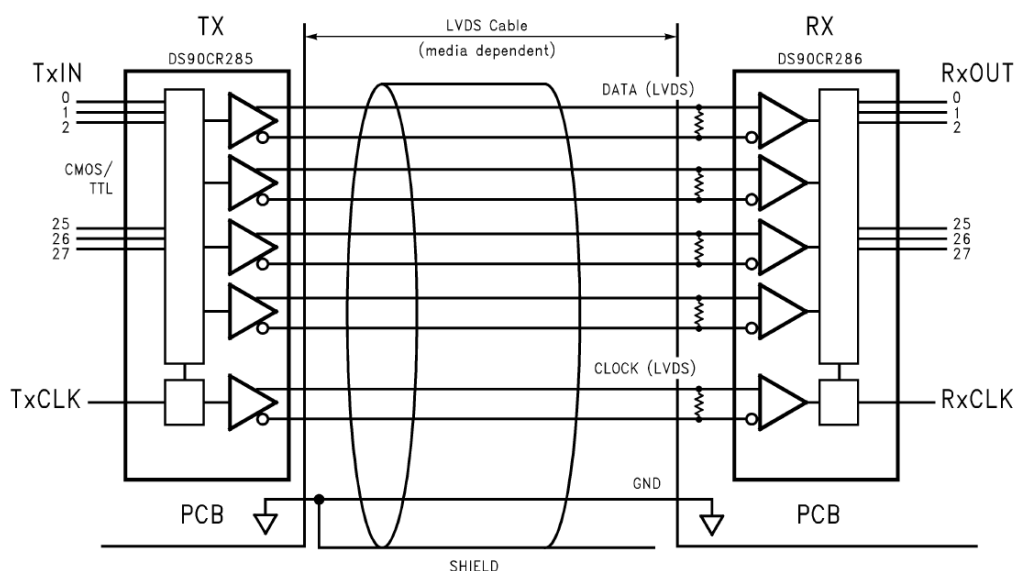


图 3.1 Cameralink 典型的应用

DS90CR286 接收器将 LVDS 数据流转换回 28 位 CMOS/TTL 数据。在收发器之间

典型的应用如图 3.1 所示。数据的传输都是通过在系统时钟频率控制下进行同步传输，在传输时钟频率为 66MHz 时，28 位 TTL 数据以每 LVDS 数据通道 462Mbps 的速率传输。使用 66MHz 时钟，数据吞吐量为 1.848Gb/s(231Mb/s)。数据线的多路复用提供了大量的电缆缩减。长距离平行单端总线通常需要一个接地线每一个主动信号并且有非常有限的噪音抑制能力。因此，对于一个 28 位宽的数据和一个时钟，需要多达 58 个导体。在通道链路芯片组中，只有 11 个导体（4 个数据对，1 个时钟对，至少 1 个接地）是必需的。这提供了 80%减少所需的电缆宽度，这提供了一个系统成本节约，减少了连接器的物理尺寸和成本，并减少了屏蔽的要求，因为电缆的较小的形式因素。28 个 CMOS/TTL 输入可以支持多种信号组合。

### 3.2 系统要求

系统图像激励源需要提供 12 路 Channelink 数字图像输出，可根据图像信息格式配置成 Cameralink 接口的 base、medium、full 模式。可以通过软件进行加载图像并配置图像模拟源的输出方式。系统图像采集处理部分需要同时对两路 Cameralink 图像数据进行采集和压缩存储等处理，同时需要上位机软件进行图像数据解压和显示功能。

### 3.3 系统设计

本系统设采用 FPGA 硬件实现电路设计<sup>[44]</sup>。系统组成主要包括图像激励源和 Cameralink 图像采集卡和上位机 PC 端。图像激励源需要通过 Camerlink 接口发送模拟高空 CCD 图像、红外图像、高光谱图像、长焦 CCD 图像、SAR 点迹图像和 SAR 图像等类型的图像，且参数可调整，具各默认输出方格图像源的功能；Cameralink 图像采集处理系统能同时对两路 Cameralink 进行采集和进行图像压缩处理并存储到固态硬盘中。基于 FPGA 实现硬件设计的 Cameralink 图像采集处理系统的需要完成如下功能，由图像激励源产生图像数据，通过 Cameralink 协议将图像发送给图像采集处理系统，图像采集来完成图像数据的采集和解串处理发送给 FPGA 芯片 I/O 口，然后对接收的图像数据流进行 JPEG-LS 的无损压缩编码，最后将压缩编码的数据通过 SATA 协议存储到 SSD 中指定的地址空间。除此之外还需要以对压缩后的图像数据通过软件进行读取并解压缩，由 PCIE 接口输出到 PC 端显示设备；系统可以同时同时对 2 路 Cameralink 图像数据进行采集和压缩编码。

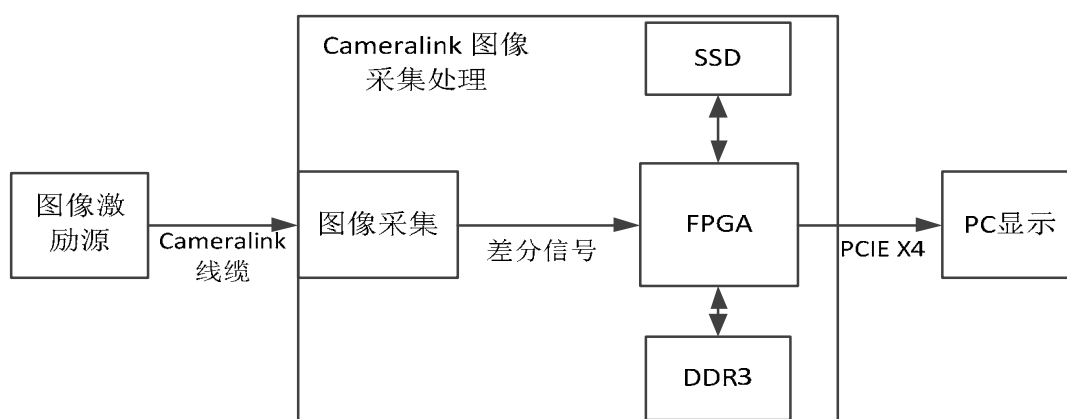


图 3.2 Cameralink 图像采集处理系统框图

系统框图如图 3.2 所示。该结构主要以 FPGA 设计应用为核心，再外加相应的外围电路，如存储设备和必要外围接口如 Cameralink 口等。可完成图像的编码、解码和图像压缩数据的传输。

(1) 图像激励源来由 FPGA 设计通过 Cameralink 接口发送芯片 DS90CR285 发送图像数据，通过 Cameralink 接口采集芯片 DS90CR286 支持最高 66MHz 时钟。对图像数据进行采集并进行解串处理。

(2) JPEG-LS 图像压缩部分由 FPGA 实现，具体实现可见第四章实现过程进行编码。

(3) 图像数据的缓存都是由 FPGA 实现。图像模拟源数据缓存通过 FPGA 外挂四片 DDR2 实现数据缓存。在图像处理完成后通过图像采集处理系统 FPGA 控制 DDR3 进行数据缓存。FPGA 为每一路图像数据分配相应的内存空间进行数据缓存。

(4) 图像数据存储部分，由 FPGA 实现 SATA 协议并分配内存来存储相应的压缩图像数据。

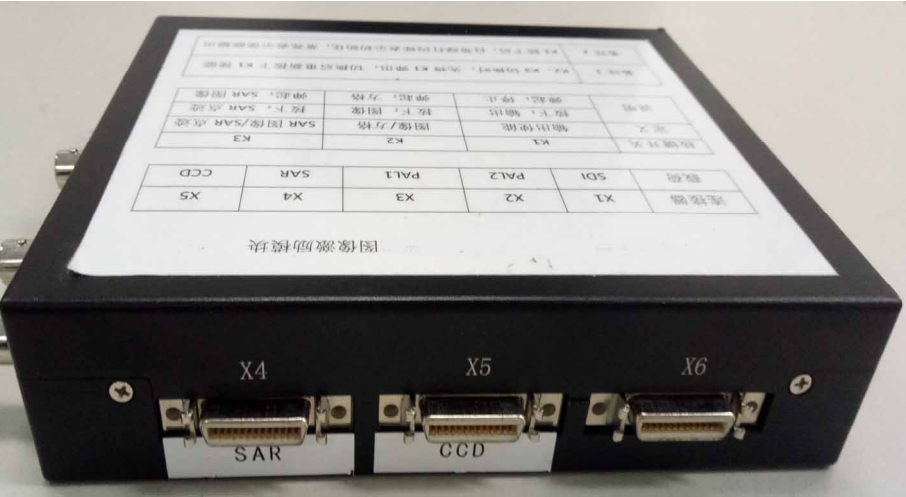
### 3.4 图像激励源单元设计

图像激励源由两部分组成：电源转换模块、图像激励源板卡。电源模块实现 220V 转 5V，供电给图像激励源板卡。

图像激励源板卡功能包括：数据加载，即通过千兆以太网接口接收任务图像模拟数据，并存储至板载固态存储器中，数据主要包括图像源；图像输出，即加电后根据控制开关或指令，将板载固态存储器中的图像模拟数据读入板载高速内存中，然后按照图像格式输出数字图像和数据；或者程序生成方格图像及辅助数据按照图像格式输出。如图 3.3(a)和(b)所示设计的图像激励源正面和侧面的实物图。



(a) 图像激励源的正面图



(b) 图像激励源的侧面图

图 3.3 图像激励源实物图

### 3.4.1 接口设计

Cameralink 接口采用 3M 公司 MDR-26 插件，其中每个插座提供一路 Base 接口或可通过配置修改定义一路 Medium 的图像接口。采用 6 个 MDR-26 插座，每个插座包含两路独立的 BASE 数据传输或一路 Medium 数据传输。

以太网接口用于数据注入接口，采用 RJ45 接口。由 FPGA 实现以太网 MAC 协议和应用层协议；PHY 芯片：88E1111-XX-BAB\_BGA117；变压器芯片：HX5008NL；以太网实现图像数据注入模拟图像源数据输出接口配置，输出使能控制等功能。

### 3.4.2 图像激励源板卡设计

图像激励源采用一片 Xilinx Virtex 5 作为的控制和处理单元,以太网接口通过 FPGA 内部 MAC 核外扩 PHY 芯片和 RJ45 连接器实现, Cameralink 接口通过 FPGA 的 IO 接口外扩 12 片 Cameralink 发送芯片 DS90CR285 和 6 个 MDR-26 连接器实现, 硬件控制和指示接口通过 FPGA 的通用 I/O 接口连接控制开关和 LED 实现, 高速内存通过采用两组 (每组 4 片 DDR2 芯片, 共 1GB) DDR2 来实现, 板载固态存储通过 FPGA 的高速串行接口外扩 MSATA 固态存储模块实现; 模块组成框图如 3.4 所示。图像激励源主要包含 Cameralink 接口和 FPGA 主芯片, 并提供以太网和固态存储器。

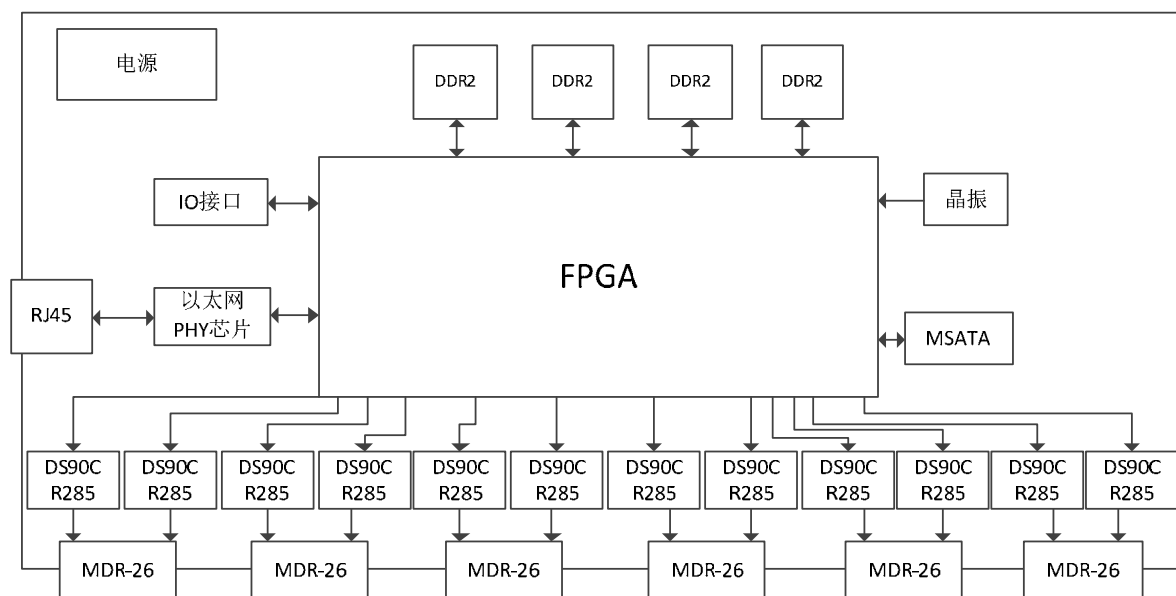


图 3.4 图像激励源板卡组成框图

Cameralink 数据传输接口采用 MDR-26 插座，每一个 MDR-26 连接器里都含有 Channel\_1、Channel\_2；本设计默认 MDR 连接器为 Cameralink 的 Y、Z 两个通道，各对应一片 Cameralink 发送芯片，Y、Z 通道的调整在逻辑程序的 FPGA 源文件中更改。

### 3.4.3 图像激励源的 FPGA 软件功能设计

图像数据加载,即通过千兆以太网接口接收任务图像模拟数据,并存储至板载固态存储器中,数据主要为图像源和视频源等;由于本系统设计只针对图像源的采集和处理,所以视频图像源暂不考虑。图像输出,即加电后根据控制开关或指令,将板载固态存储器中的图像模拟数据读入板载高速内存中,然后按照图像格式输出数字图像和数据;或者程序生成方格灰度图像及辅助数据按照需要的图像格式输出;程序生成的方格图像数据的好处是防止FPGA外围电路故障,FPGA还可以生成模拟图像数据以备系统使用。

设计输入分析，主要包括数据流的接口名称和方向，控制与状态等。具体定义如表 3.1 所示设计使能等。

表 3.1 功能接口定义与说明

类型	接口名称	方向	说明
数据流	以太网	双向	图像数据注入和指令控制以及状态监测
	DDR2	双向	高速内存
	MSATA 固态硬盘	双向	板载数据存储
	Cameralink 输出	输出	通过 Cameralink 协议输出数据到 Cameralink 发送数据芯片
控制与状态	模式选择	输入	选择当前图像的输出模式
	输出总使能	输入	有效时，所有数据才会输出（相当电平复位）
	图像输出使能	输入	CCD 图像,红外图像,高光谱图像,SAR 图像,SAR 点迹图像的输出使能
	图像源选择	输入	选择是程序生成方格图像源或板载固态存储数据源（共用设计同上）
	图像输出指示灯	输出	指示灯亮时，表示当前图像正在输出

设计图像激励源发送图像数据，图像激励源可发送三种模式分别为模式一、模式二和模式三，其中模式一包括两路高空 CCD 图像、红外图像、高光谱图像、两路长焦 CCD 图像、两路 SAR 点迹图像和一路 SAR 图像等图像数据的模拟发送。模式二包括高空 CCD 图像、红外图像、高光谱图像、SAR 点迹图像和 SAR 图像等图像的模拟发送。模式三包括 SAR 图像和 CCD 图像的模拟和发送。由于需要模拟的图像数据种类多。为简化 FPGA 软件设计图表 3.2 所示，增强系统可靠性和高效性，特将针对不同模式要求的 FPGA 设计成 3 个版本，在上电启动时根据选择开关，选择将那个版本软件加载进 FPGA 运行。数据以太网通信，数据输出的功能均由 FPGA 来实现，功能模块示意图如 3.5 所示。



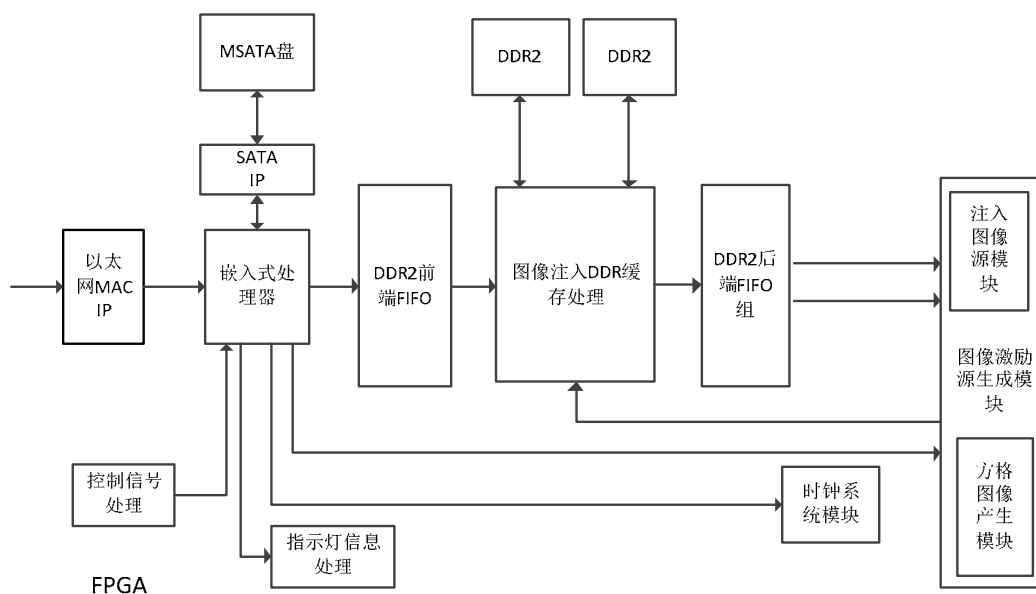


图 3.5 FPGA 功能框图

表 3.2 软件功能需求表

模式	功能名称	说明
图像模拟输出模式	根据硬件开关选择模式与输出	1、根据模式开关确定当前为模式一或模式二图像输出； 2、根据载图像出使能，确定需要输出图像； 3、根据图像模拟源选择，确定输出方格图像源或板载固态存储数据源； 4、输出状态的指示灯信号； 5、SAR-IMG 默认输出最大分辨率图像的数据，可通过以太网接口更改； 6、模式二的高空 CCD 默认输出最大分辨率的数据，可通过以太网接口更改；
设计更新及配置模式	根据以太网指令确定模式，更新板载固态存储数据	1、以太网连接后，进入板载数据更新及配置模式； 2、在此模式下，通过以太网接口注入新的图像源数据至板载固态存储器； 3、此模式下可以配置的内容包括： 通过以太网指令配置 IR 的输出格式； 通过以太网注入图像数据，并可选择输出注入的数据，也可以选择输出方格源； 模式一的 SAR-IMG 可以选择最小和最大分辨率两种模式，模式二的高空 CCD 可以选择两种输出模式；

### (1) 时钟和缓存设计

各类型图像的同步时钟不一致，根据需要产生不同的输出时钟。为提高系统时钟输

出的稳定性,减少时钟切换带来的影响,将高空 CCD 图像 54Mhz 和 SAR 图像的 57.5Mhz 的时钟统一调整为 60Mhz,对 IR 时钟统一调整 30Mhz。具体模式输出图像类型和时钟设计如表 3.3 所示。

表 3.3 模拟图像源输出图像时钟要求

模式一	模式二	模式三	时钟 (Mhz)	Cameralink 芯片编号
GKCCD-base0	GKCCD0-X/HIS0-X/SAR-D-X		60/60/20/60	1
GKCCD-base1	GKCCD0-Y/HIS0=Y/SAR-D-Y		60/60/20/60	2
IR			30	3
HIS			25	4
LFCCD-X	GKCCD1-X/HIS1-X/SAR-I-base		60/60/20/60	5
LFCCD-Y	GKCCD1-Y/HIS1-Y		60/60/20	6
SAR-DATA-X		SAR-DATA	50/50	7
SAR-DATA-Y		SAR-DATA	50/50	8
SAR-IMG			40	9
				10
		CCD-0	x	11
		CCD-1	x	12

注: GKCCD 为高空 CCD 图像、IR 为红外图像、HIS 为高光谱图像、LFCCD 为长焦 CCD 图像、SAR-DAT 为 SAR 点迹图像、SAR-IMG 为 SAR 图像。其中-X 指连接器 X 通道, -Y 同理。Base 指 Cameralink 的 base 模式。

嵌入式处理器通过以太网接收注入数据后, FPGA 应将数据存入 DDR 缓存内, 然后各图像模拟图像源生成模块根据指令从 DDR 缓存内读取数据。FPGA 为每种图像分配相应的缓存空间。当收到某图像读取指令(即该图像启动, 并且输出源选择为注入数据)时, 本模块根据当前图像配置模式, 循环读取一定数量 DDR 缓存的数据到后端 FIFO。图像读取停止时, 应将后端 FIFO 复位。DDR 数据的读取与停止, 由图像激励源生成 IP 模块控制产生。

(2) 图像输出设计

模拟图像源包括方格源和注入图像输出。每个子模块含方格图像源生成单元模块和注入图像输出单元模块。

方格图像源单使用 FPGA 逻辑自动产生图像输出, 根据图像的速率以及分辨率, 产生方格渐变图像, 并可以根据指令调整图像大小的分辨率。

注入图像输出模块分为两种类型, 一种是将 DDR 数据循环读(DDR 缓存中的数据

只读取到后端 FIFO 中), 每行图像数据进行分块, 然后按照 Cameralink 格式输出; 第二种是辅助数据需要单独处理 (模式一-高光图像, 模式一-长焦 CCD 图像, 模式二-高空 CCD 图像), 从 DDR 缓存中循环读出的有辅助数据, 在需要发送辅助数据时, 将辅助数据行读出, 然后按照要求对每行数据 (包括辅助数据) 分块处理后按照 Cameralink 格式输出; 再读取图像数据做相同操作; 如果当前不需要发送辅助数据, 则该行辅助数据读取后丢弃。该模块接收图像启动指令以及配置模式, 并生成 DDR 读取和停止指令 (同时传递数据块循环读取时, 读取的数据量参数)。

### (3) 处理器设计

嵌入式处理器软件采用简单的单线程 (无操作系统) 实现 IO 的管理以及网络通信。采用 C 代码编程, 采用 Xilinx 自带的 EDK 进行调试和编译。FPGA 工程中的模块 “system” 即为在系统工程中调用的嵌入式处理器模块。外部硬件指令, 指示灯均通过嵌入式处理器软件控制。网络通信采用 TCP/IP 协议使用 FPGA 自带的 MAC IP 核。

实现 TCP/IP 网络协议, 将数据接收后写入 MSATA 盘, 之后在模拟输出模式下, 将数据从 MSATA 盘读出, 发送到高速内存进行缓存, 对系统进行管理和控制, 对外部开关量进行处理。DDR2 缓存处理: DDR2 接口的实现和数据仲裁, 以及根据配置模式读取 DDR2 中的数据。图像模拟源生成模块: 含方格图像源生成模块, 以及将 DDR2 中的图像数据转换成图像输出格式, 最后将数据以 Cameralink 协议要求输出; 同时根据指令和参数启动 DDR2 缓存处理模块的数据读取操作。

## 3.5 Cameralink 图像采集处理单元设计

Cameralink 图像采集处理单元同时能对两路 Cameralink 图像数据进行采集和处理。Cameralink 接口采集芯片选用 TI 的 DS90CR286, 对图像数据进行采集并进行解串处理, 经 FPGA 先缓存 DDR3 中, 处理后经 JPEG-LS 无损压缩数据向 SATA 链路向数据存储板传输。DS90CR286 支持最高 66MHz 时钟。FPGA 器件选用 Xilinx 的 XC7K325T 芯片单端 I/O 接口为 500 个, 扩展的 I/O 接口可以连接图像采集接口进行数据的传输, 本系统采用 Cameralink 接口传输数据, 并将数据并行输出到系统的存储控制模块中, 控制模块通过预充电、刷新等命令操作控制数据通过对应的地址进行相应的数据存储。缓存的硬件使用 DDR3 进行高速缓存。可用于连接 Cameralink 接口芯片, 采集图像数据。然后将采集到的数据由 FPGA 控制缓存到 DDR3 指定地址空间中, 从 DDR3 相应的地址空间读取图像数据经过 FIFO 转换 16 位输出给到 JPEG-LS 编码器的 16 为输入端口, 然后由 FPGA 进行图像无损压缩编码, 然后将压缩数据经过后端 FIFO 存储到 SSD 中, 应用到后续的解压与显示。Cameralink 图像采集处理系统设计如图 3.6 所示。

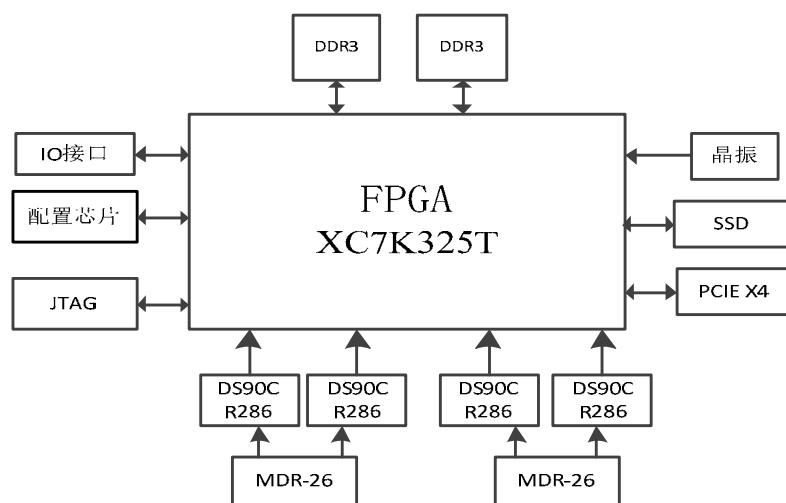


图 3.6 Cameralink 图像采集处理系统设计

在设计 Cameralink 图像采集处理时,由于考虑到板卡的可兼容更多功能使用、测试和采集等需求,采用通用的设计方式设计板卡。模板就是 FPGA 芯片挂载一些外围典型的应用电路如 DDR3、SSD 和 PHY 芯片还可以增加更多的功能设计。板卡只有 4 片 DS90CR286 和两个 MDR-26 型连接器。如图 3.7 所示设计 Cameralink 图像采集处理板卡实物图。

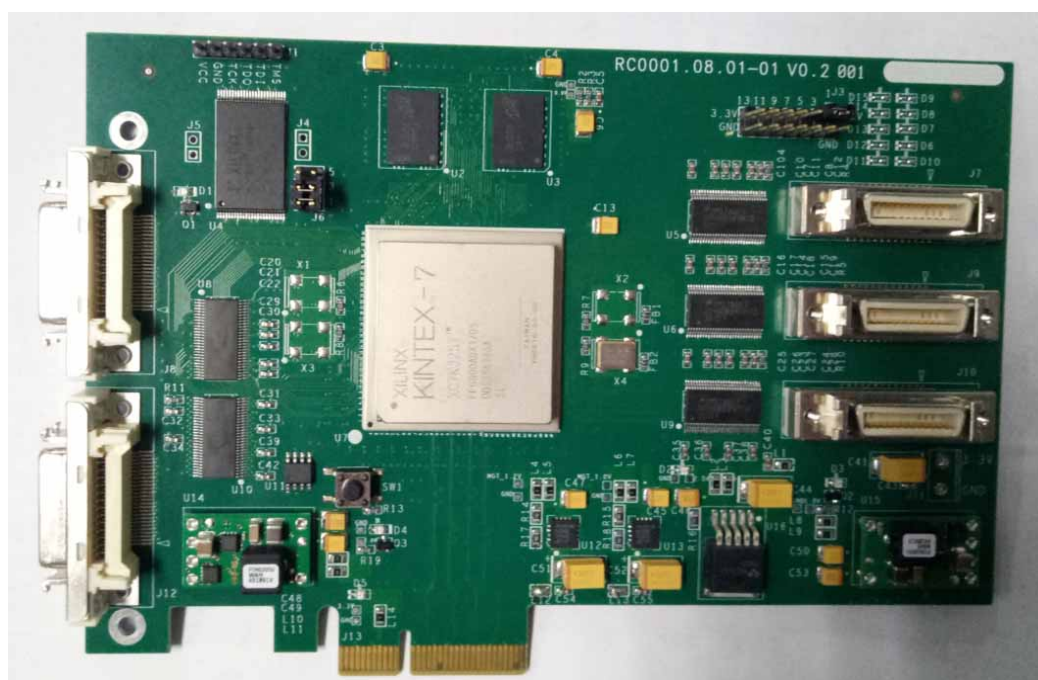


图 3.7 Cameralink 图像采集处理卡

本处理单元通过 PCIE X4 插在上位机上,由于对外空间有限所以使用两个 MDR-26

连接器。图像发送器 DS90CR285 和接收器 DS90CR286 之间两个 MDR-26 接口选择的 Cameralink 电缆接口需要支持不同的 LVDS 对。21 位通道链路芯片组(DS90CR215/216)需要 4 对信号线和 28 位通道链路芯片组(DS90CR285/286)需要 5 对信号线。理想的电缆或者连接器接口将有一个恒定的 100W 差动阻抗贯穿整个路径。电缆的斜度需保持在 150ps(66MHz 时钟频率)以下，以便在接收器上保持足够的数据采样窗口。

### 3.5.1 缓存模块设计

在该系统中 Cameralink 图像采集处理单元缓存设计使用 MCB 硬核进行数据的传输控制，MCB 硬核能够支持多种存储硬件，对于工程来讲，其 MCB 硬核具有更好的误码校验和偏移时钟校验。不仅支持高速存储芯片，也可以控制其它的高速存储芯片进行图像数据的存储，在该工程设计中使用 ISE 软件中的 IP 库选用 MCB 硬核，大大减少了开发周期，加快开发进度，可以在工程中直接调用，并且更具不同的数据传输方式，配置不同的传输参数，并且可以通过配置界面选择不同的存储器件类型，设置不同的数据位宽和满足系统不同的速度传输等级，如果所选择的器件与硬核不能够兼容，通过修改传输接口的 VHDL 语言控制数据的传输方式。MIG IP 控制器是 Xilinx 为用户提供的—个 DDR 控制的 IP，这样即使不了解 DDR 的控制和读写时序也能通过 MIG 控制器方便的读写 DDR 存储器。MIG 控制器的框图如 3.8 所示。

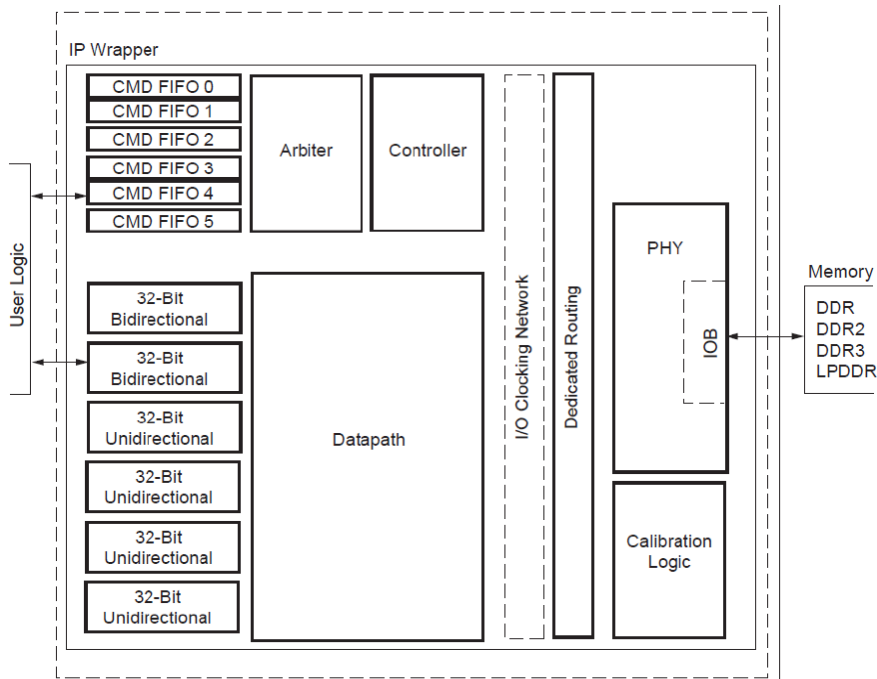


图 3.8 MIG 控制器框图

DDR3 和 FPGA 之间的连接的数据宽度都为 16 位，另外设计板上对 DDR3 的地址

线和控制线都做了端接电阻上拉到 VTT 电压, 保证信号的质量。在 PCB 的设计上, 完全遵照 Xilinx 的 DDR3 设计规范, 严格保证等长设计和阻抗控制。数据以总线形式体现另一端接 FPGA 的 I/O 端口。DDR3 部分的电路设计原理图如附录图 1 所示。

### 3.5.2 配置电路及 JTAG 设计

Cameralink 图像采集处理的 FPGA 配置芯片选择芯片 N25Q128 来设计。N25Q128 是一个 128Mbit (16Mb x 8) 串行闪存, 具有先进的写保护机制。它通过高速 SPI 兼容总线进行访问, 并具有在 XIP (“优先执行”) 模式下工作的可能性。N25Q128 支持创新的高性能四双 I/O 指令, 这些新的指令可以使读取和编程操作的传输带宽增加一倍或四倍。此外, 内存具有三种不同的协议进行操作包括标准 SPI (扩展 SPI 协议)、双 I/O 的 SPI 和四路 I/O 的 SPI。标准 SPI 协议由新的四指令和双指令 (扩展 SPI 协议) 丰富。对于双 I/O SPI(DIO-SPI)的所有指令代码, 地址和数据总是通过两条数据线传输。对于四路 I/O 输出 SPI(QIO-SPI)指令代码, 地址和数据总是通过四条数据线传输, 从而大大提高了随机存取时间和数据吞吐量。

Flash 可以工作在 “XIP 模式”, 设备只需要地址而不是指令输出数据。这种模式大大减少了随机存取时间, 从而使许多应用程序需要快速的代码执行, 而不会影响 RAM 上的存储器数据。XIP 模式可以与 QIO-SPI, DIO-SPI 或扩展 SPI 协议一起使用, 并且可以使用不同的专用指令进入和退出, 以实现最大的灵活性: 对于在设备加电后立即进入 XIP 模式的应用, 可以通过使用专用的非易失性寄存器(NVR)位将其设置为默认模式。也可以通过快速 POR (上电复位) 功能降低上电顺序时间, 从而在执行第一条读指令之前缩短等待时间。另一个功能是通过使用专用的编程/擦除暂停和恢复指令来暂停和恢复编程和擦除周期。本设计主要选择在 XIP 模式工作设计。XIP (现场执行) 模式可用于每种协议: 扩展 SPI, DIO-SPI 和 QIO-SPI。XIP 模式允许通过向器件发送地址, 然后根据要求在一个, 两个或四个引脚上并行接收数据来读取存储器。它为应用程序提供了最大的灵活性, 节省了指令开销, 并大大减少了随机访问时间。

在 NVCR (非易失性配置寄存器) 11 到 9, 取决于需要哪种 XIP 类型, 单个, 双或四个 I/O 时, 存储器仅在下一个上电序列之后进入选定的 XIP 模式。非易失性配置寄存器 XIP 配置位允许内存在上电后直接以所需的 XIP 模式启动。如图 3.9 所示为工作时读取过程流程图。

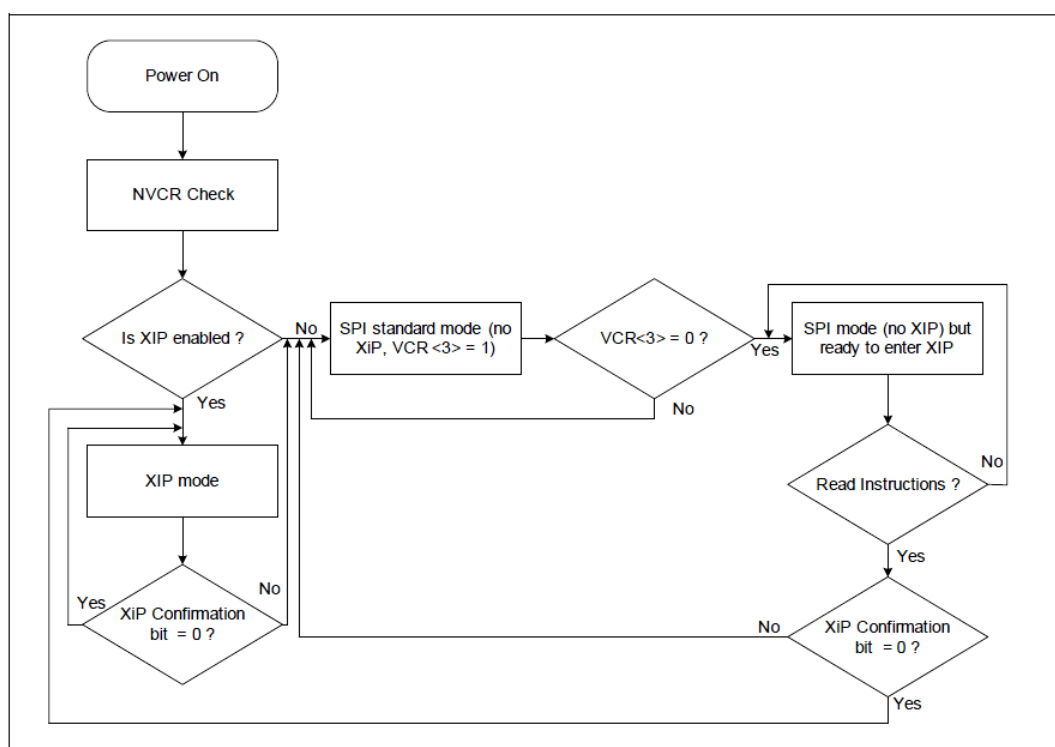


图 3.9 读取功能流程图

当 FPGA 处于主串行模式时，它会产生驱动 PROM 的配置时钟。使用 CF 高电平时，启用 CE 和 OE 后的访问时间短，数据在连接到 FPGA DIN 引脚的 PROM 数据(D0)引脚上可用。新的数据在每个时钟上升沿之后的访问时间较短。FPGA 产生适当数量的时钟脉冲来完成配置。当 FPGA 处于 Slave 串行模式时，PROM 和 FPGA 都由外部时钟源或可选的芯片提供时钟，PROM 可用于驱动 FPGA 的配置时钟。Platform Flash PROM 的芯片还支持 Master SelectMAP 和 Slave SelectMAP（或 Slave Parallel）FPGA 配置模式。当 FPGA 处于主 SelectMAP 模式时，FPGA 产生驱动 PROM 的配置时钟。当 FPGA 处于从 SelectMAP 模式时，外部振荡器产生驱动 PROM 和 FPGA 的配置时钟，或者可选择使用芯片 PROM 来驱动 FPGA 的配置时钟。BUSY 为低电平，CF 为高电平时，使能 CE 和 OE 后，数据在 PROM 数据(D0-D7)引脚上可用。新的数据在每个时钟上升沿之后的访问时间较短。数据在 CCLK 的下一个上升沿时钟输入 FPGA。自并行/从模式 SelectMAP 模式下可以使用自由运行的振荡器。Platform Flash PROM 的芯片提供了额外的高级功能。内置的数据解压缩器支持使用压缩的 PROM 文件，设计修改允许将多个设计修订存储在单个 PROM 中，或存储在多个 PROM 中如图 3.10 所示 PROM 框图。

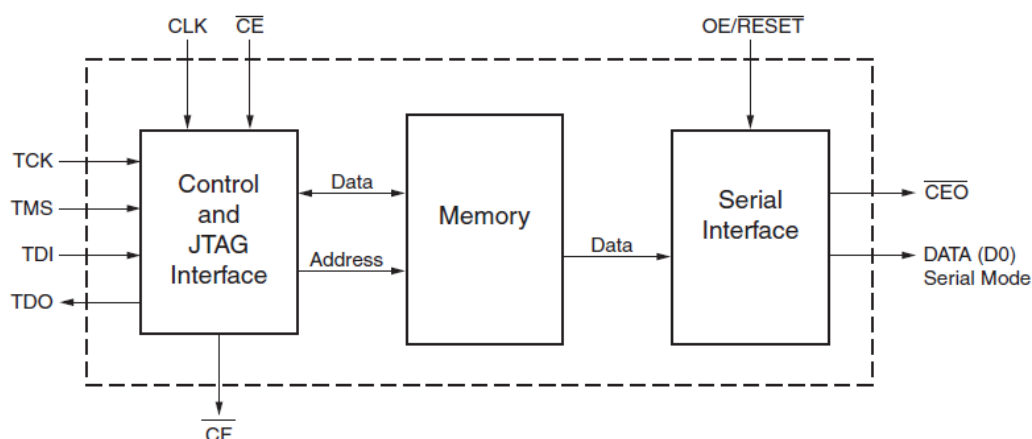


图 3.10 Flash PROM 框图

### 3.5.3 时钟设计

时钟设计采用 FPGA 自动生 IP 核，由于系统接口使用的不同时钟频率，所以需要采用时钟分频来对系统时钟分频以供相应的元件电路使用。根据外部时钟接入到 FPGA 内部进行分频设计来实现时钟分频，其电路包括鉴相器 PD、环路滤波器、压控振荡器和分频器等模块，其原理框图如图 3.11 所示。由于 Cameralink 接口提供发送的时钟，所以不考虑 Cameralink 的时钟设计，只需要设计 DDR3，SATA，还有分频所用的参考时钟。DDR3 设计时钟频率 125MHz 差分时钟，SATA 时钟设计 150MHz 差分时钟等。

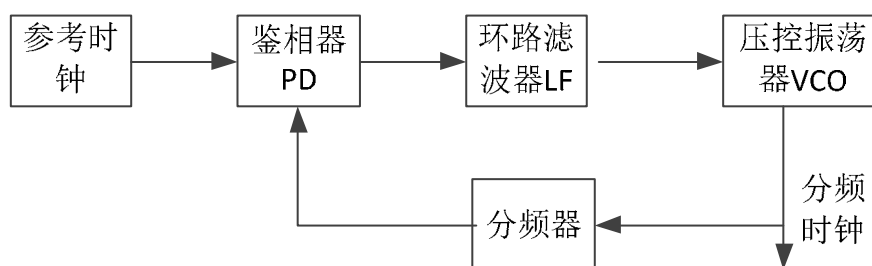


图 3.11 时钟分频原理框图

### 3.5.4 SATA IP 核设计

SATA 是 2003 年产生的外设接口标准<sup>[45]</sup>，用于取代并行 ATA(PATA)。最快的 PATA 速度是 133MB/s，而 SATA 第一代(SATA I)则以 150MB/s 的速度运行。SATA 具有许多其他功能，使其优于 ATA，包括更小的电缆，更少的引脚和更低的工作电压。SATA 协议使用分层体系结构，其中每个层使用其下层的的服务。应用层通常代表使用 SATA 设备的软件，命令层模拟 PATA 命令以实现向后兼容。传输层负责创建和格式化称为帧信息



结构(FIS)的数据帧和有效的 FIS 序列。在该层之下是链路层，它编码和封装 FIS，处理控制信号并检查 FIS 的完整性。

堆栈的最底层是物理层，它处理电信号的发送和接收并保持数据对齐。SATA 使用低电压差分信号(LVDS)发送和接收数据，以及用于时钟和数据恢复的 8b/10b 编码。编码为每个 8 位数据值分配一个 10 位字符。所发送的数据值不是基于发送数据的两个导体之间的电压差，而是相对于公共地发送 1 和 0。物理层也建立与磁盘的通信如图 3.12 所示 SATA 协议的主控制框图所示显示内部组成的功能让结构。通过带外 OOB 信令执行的，其中发送对的通信线路被驱动到相同的电压以产生不存在信号差异。这个状态是在一个预先定义的突发模式中执行的，以发送一个 OOB 原语。为 SATA 定义了三个 OOB 原语(COMRESET, COMINIT 和 COMWAKE)，它们被用作初始化握手的一部分以开始与 SATA 设备的通信。

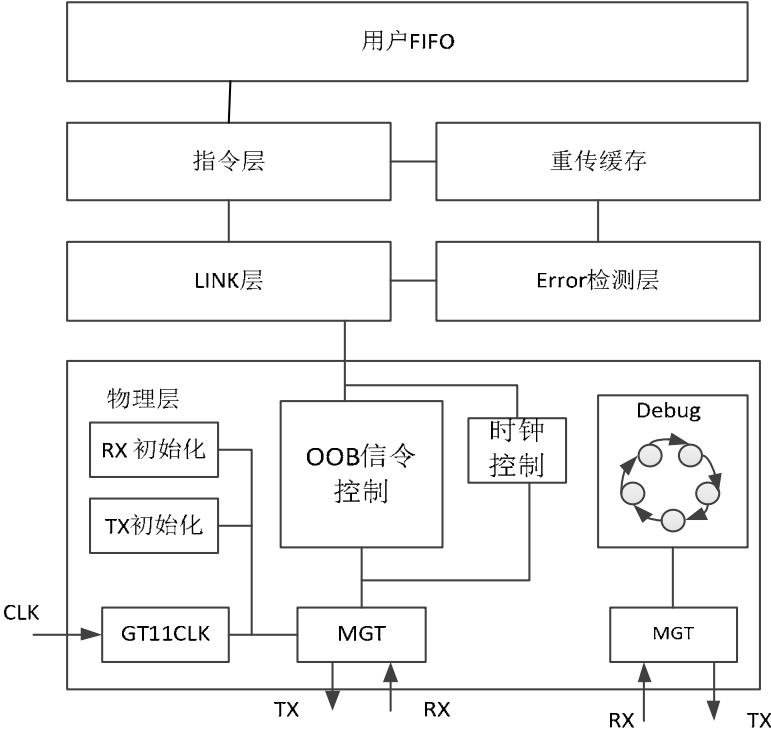


图 3.12 SATA 主控制器框图

在本节介绍 FPGA 设计的开源串行 ATA 内核设计。测试内核使用接近 1.5Gb/s 的速率向 SATA II 固态驱动器传输信息。另外，使用不同的电路板和 FPGA 芯片显示成功的 SATA II 传输。用户电路可以通过 FIFO 接口与 IP 进行信息交互。在本设计中采用现有 SATA IP 设计进行集成到 FPGA 的处理单元里实现调用接口和内存分配管理等功能。

### 3.6 性能分析

Cameralink 图像采集处理单元采用图像激励源单元生成的 SAR 图像数据大小  $4008 \times 5344$ ，深度为 8bit 灰度图像方格图像数据进行测试验证。图像激励源单元主要用于测试验证所设计 Cameralink 图像采集处理单元的功能性能和调试 FPGA 对采集的数据进行 JPEG-LS 压缩存储等。FPGA 与 PC 端之间的数据通信通过 PCIE 协议从 SSD 中对压缩数据的读取并且在终端正常解压缩恢复原始图像显示等。Cameralink 图像采集处理单元通过 PCIE 金手指插在 PC 端插槽内，实际交联如图 3.13 所示。图 3.14 所示为 Cameralink 图像采集处理单元所占逻辑资源统计情况。



图 3.13 Cameralink 采集处理卡与 PC 端

Device Utilization Summary				
Slice Logic Utilization	Used	Available	Utilization	Note(s)
Number of Slice Registers	6,283	69,120	9%	
Number used as Flip Flops	6,279			
Number used as Latches	4			
Number of Slice LUTs	5,876	69,120	8%	
Number used as logic	4,873	69,120	7%	
Number using O6 output only	3,197			
Number using O5 output only	939			
Number using O5 and O6	737			
Number used as Memory	929	17,920	5%	
Number used as Shift Register	929			
Number using O6 output only	921			
Number using O5 output only	4			
Number using O5 and O6	4			
Number used as exclusive route-thru	74			
Number of route-thrus	1,015			
Number using O6 output only	1,013			
Number using O5 output only	2			
Number of occupied Slices	3,150	17,280	18%	
Number of LUT Flip Flop pairs used	8,796			
Number with an unused Flip Flop	2,513	8,796	28%	
Number with an unused LUT	2,920	8,796	33%	
Number of fully used LUT-FF pairs	3,363	8,796	38%	
Number of unique control sets	393			
Number of slice register sites lost to control set restrictions	708	69,120	1%	
Number of bonded IOBs	284	640	44%	
Number of LOCed IOBs	284	284	100%	
IOB Flip Flops	153			
Number of BlockRAM/PIFO	20	148	13%	

图 3.14 逻辑资源占用统计

将测试平台搭建好进行上板验证，验证结果表明，cameralink 图像采集处理系统通过 PCIE 与计算机通信正常，按照预定协议从 SSD 中设置的地址空间读取压缩 SAR 图像数据，并能正确解压接收到的数据和显示。其中解压和显示在 PC 上运行。经过解压后的数据分析表明整个系统工作稳定、压缩编码效率高、实时处理速度快等特点可以实际应用工程中。解压后显示的方格图像如图 3.15 所示。

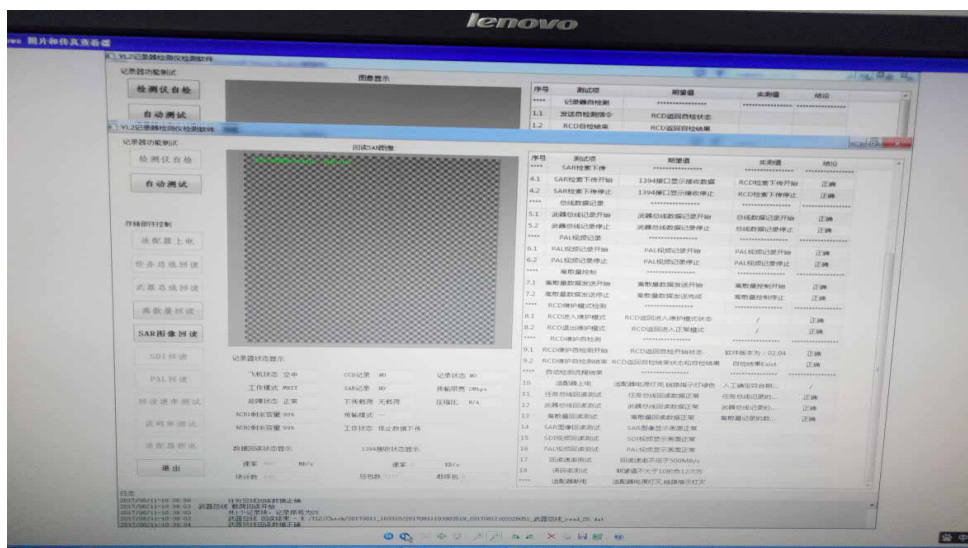


图 3.15 图像数据解压后显示

### 3.7 本章小结

本章介绍了 Cameralink 图像采集处理系统设计,选择通过 FPGA 为主的数据处理的硬件平台来实现系统设计。首先设计了图像的模拟源,通过 FPGA 实现数据源不同类型的图像数据的发送还可以通过网络加载图像数据存储到 MSATA 里,在根据需要发送相应的图像数据。其次实现了基于 Cameralink 进行图像采集和处理并把像压缩数据存储到 SSD 中。最后通过对系统的验证测试表明系统的工作正常,能够完成图像的采集和处理功能,实时处理速度快、鲁棒性高和可实际工程中应用。选择 Cameralink 传输图像数据首先传输速率快,其次可靠性高,因此对于要求较高的场合是不错的选择。

## 第4章 JPEG-LS 无损压缩算法设计

FPGA 在数字图像压缩领域与传统的数字信号处理器件相比,不但具有很高的可靠性高、灵活的设计思路和稳定性较高的编程语言,还可以减少图像压缩处理系统的体积和功耗,实现图像高速实时处理和高保真等特点。对于多数应用而言,FPGA 的功耗也能满足需要。因为 FPGA 并行执行的特点非常适合图像处理,所以总体拥有成本更低、灵活性更高的 FPGA 往往是最佳技术选择。因此,本章在前面 JPEG-LS 图像无损压缩算法上采用 FPGA 的设计语言 VHDL 语言<sup>[46]</sup>对其进行了实现。前面两节对 FPGA 设计流程和 VHDL 语言设计,主要实现采用 VHDL 语言对上下文建模设计、预测编码设计和 Golomb 编码的设计等。本文设计的 JPEG-LS 无损压缩编码器相对于文献<sup>[50]</sup>具有输出码流稳定、较高的鲁棒性和恢复原始图像无失真等特点,并且在上下文建模设计和预测误差计算上对预测校正值  $C[Q]$  的更新进行了改进处理,减少了逻辑资源的占用、提高了编码效率,进而提高了编码器时钟频率。最后对设计的 JPEG-LS 编码器进行功能仿真验证。

### 4.1 JPEG-LS 无损压缩算法设计

基于 FPGA 实现 JPEG-LS 无损压缩算法采用并行流水线操作的设计方式,在 FPGA 实现中采用 TOP-DOWN 设计思想,可划分为以下几个模块:数据接收缓存模块 RAM 缓存图像数据并得到当前像素位置和建立上下文模板。梯度计算模块完成上下文模板梯度值的计算、映射并得出上下文局部梯度量化值。固定预测编码模块预测当前像素的值  $P_x$ ,然后自适应修正模块对预测值进行误差  $Errval$  修正,并且对预测误差  $Errval$  进行映射得到非负的数值  $MErrval$ ,然后对映射误差  $MErrval$  进行 Golomb 编码,即对映射误差  $MErrval$  作除法计算,对商进行一元编码,对余数进行二进制编码。顶层参数配置完成编码参数的初始化和更新功能。JPEG-LS 无损压缩算法的实现既在整体流程上保证了并行执行操作,又在局部上采用并行设计,降低流水深度。另外,在每个模块的内部计算也最大可能的并行展开,集中体现了流水与并行互相嵌套的设计思想,提高系统的效率。FPGA 实现 JPEG-LS 无损压缩算法流程如图 4.1 所示。

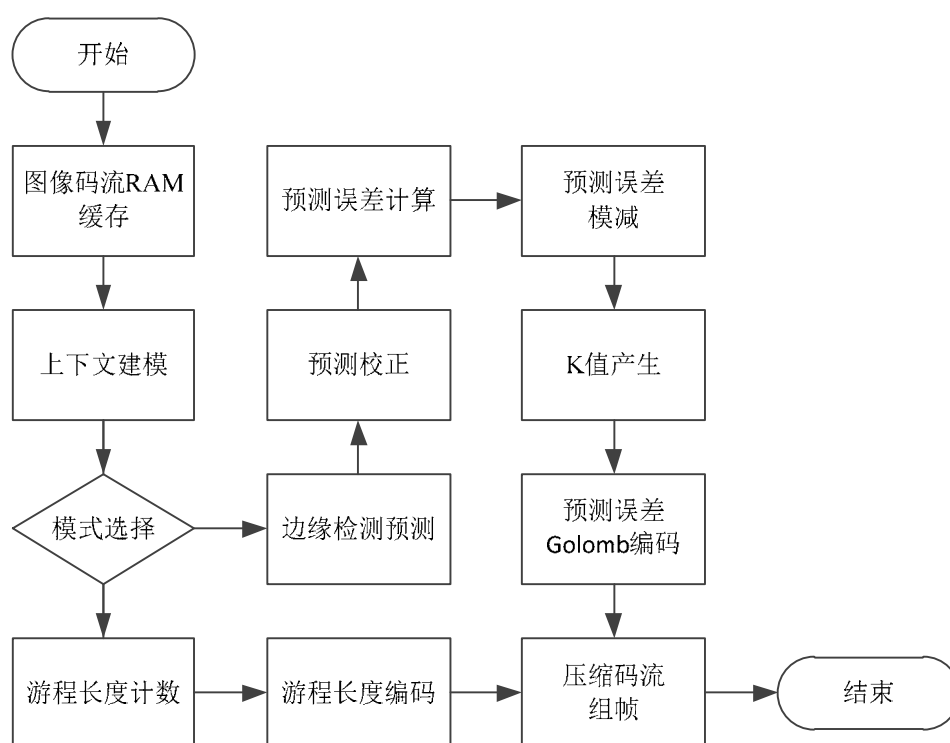


图 4.1 FPGA 实现 JPEG-LS 无损算法流程图

由于到卫星遥感图像中出现大块的纹理复杂域图像的概率相对而言比较大,从而选择游程模式的比例很小,体现不了游程编码的优点,且增加游程模式会使 FPGA 设计的模式选择时增加逻辑判断和不必要的延迟,会使编码速度减慢并且设计复杂度和实现难度也会相应的增加,但由于 FPGA 在并行处理编码图像时编码效率完全可以弥补编码中没有游程长度编码,因此现阶段的硬件实现上只考采用常规模式进行无损图像压缩。对红外两个波段(4.3mm 和 2.7mm)的卫星遥感图片的实验结果表明<sup>[47]</sup>,去掉游程编码的图像无损压缩算法性能的压缩比影响不大。

本章的总体设计思路是由 JPEG-LS 压缩编码标准<sup>[40]</sup>的正常编码而来的,由于 VHDL 语言设计方法采用自顶向下,从顶层进行功能划分和结构设计。基于 FPGA 设计 JPEG-LS 无损压缩算法先从顶层设计开始,首先给出顶层的输入输出分配,以及内部子模块所需要的系统时钟、复位、使能以及 16 位的图像码流等输入管脚。其次,给出 32 位的编码 FIFO 输出比特流、Golomb 编码输出的 5 位的 K 值、预测误差 Errval 映射的非负数值 MErrval 的 16 位比特流、编码 FIFO 输出写请求 `async_fifo_wreq_code`、使能信号等输出管脚。JPEG-LS 编码器顶层模块 RTL 图如下图 4.2 所示。

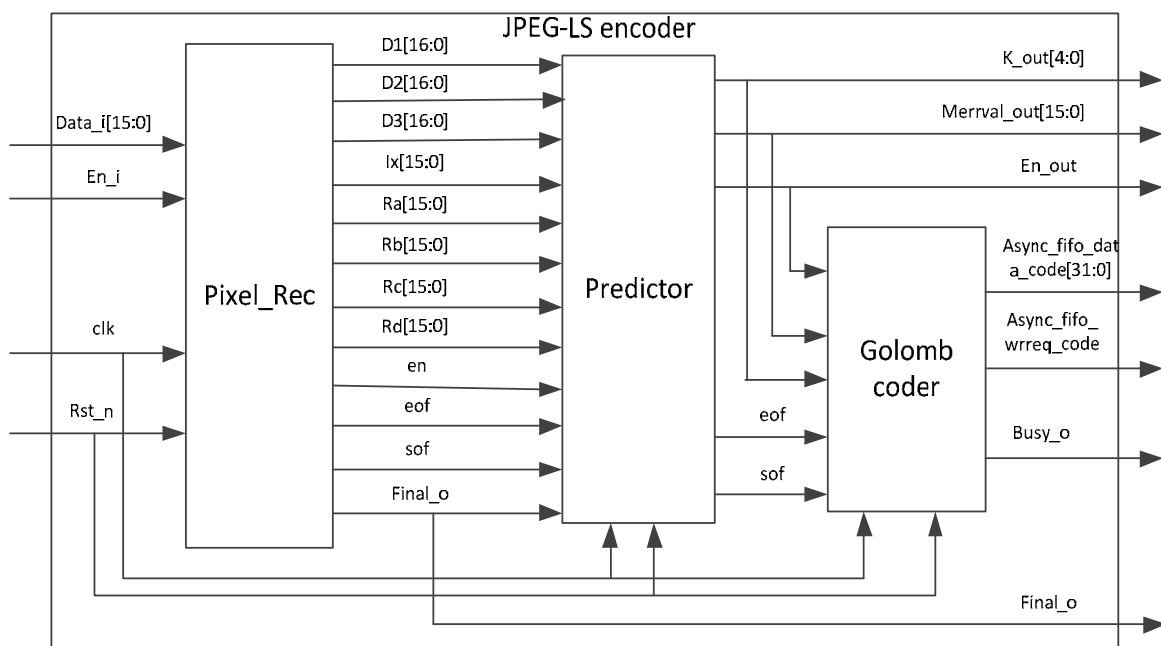


图 4.2 JPEG-LS 编码器顶层结构图

在 JPEG-LS 的硬件设计中，用 VHDL 语言实现上下文建模模块、预测编码模块和 Golomb 编码模块等子模块。存储器 RAM 中读取当前编码像素和上下文像素 a、b、c 和 d。首先求出上文的像素 a、b、c 和 d 的重建值（无损模式下就是像素的真实值），然后求出梯度值 D1、D2、D3 值。当且仅当 D1、D2 和 D3 不等于 0 时，编码模式才会进入正常编码模式。预测误差 Errval 的计算采用当前像素的预测值 Px 和像素的实际值 Ix 来计算。然后对预测误差 Errval 映射到一个非负整数 MErrval 使用 Golomb 编码进行熵编码。考虑到 JPEG-LS 编码器对实时性的要求，通过利用 FPGA 中的并行和流水线来完成硬件的实现。设计的编码器系统的框图如图 4.3 所示。

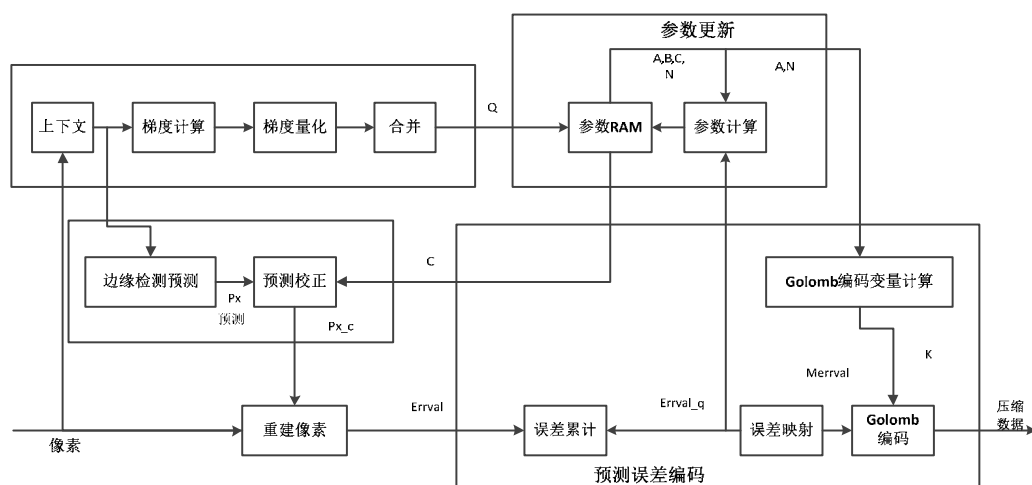


图 4.3 JPEG-LS 编码器数据流图

在编码之前先对编码器编码初始化参数进行设定，代码中的变量类型都假设为整数，它们具有足够大的精度，以便在执行所需的算术运算不会产生上溢或下溢。

#### 4.1.1 上下文建模设计

上下文建模模块主要实现功能是对输入图像的比特流进行接收缓存到 1K 的片内 RAM 中，从片内 RAM 读出以  $16 \times 16$  像素块读出采用计数器方式实现。同时，通过列计数器作为 RAM 的读地址操作，当列计数器达到 16 时清零同时行计数器累计加 1，这里的计数器就是根据时钟的上升沿的累计加 1 的，并且在编码过程中输出待编码的像素点局部梯度值以及相邻像素点的重构值，图像的采样值就是像素的真实值并在每个像素点上下文建模编码完毕后续存其重构值。定义了 JPEG-LS 在无损压缩编码过程像素的采样重构值实际上就是图像的真实像素值。还实现了采样像素点在  $16 \times 16$  像素块边缘的检测和像素的重建采样值及输出开始使能信号 sof 和结束使能信号 eof。本模块设计思路参照 JPEG-LS 压缩标准的上下文建模的思路来实现。上下文建模的 RTL 图如图 4.4 所示。

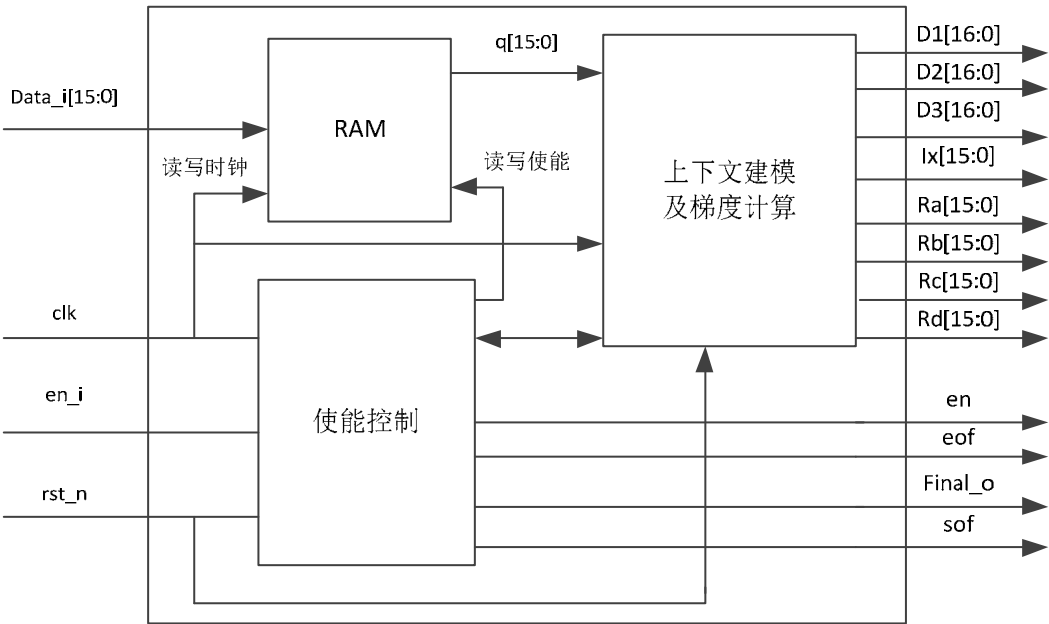


图 4.4 上下文模块结构图

当对图像的第一行进行编码时，位置 b，c 和 d 处的像素不存在，并且其重构值被定义为零。如果位置 x 上的像素位于一条线的起点或终点，使得 a 和 c 或 d 不存在，则位置 a 或 d 上的像素的重建值被定义为等于 Rb，重建在位置 b 处的像素值，或者在图像中的第一行为零。在对前一行中的第一个像素进行编码时，位置 c 的像素处的重构值将被复制（对于除第一行以外的行）的值被分配给 Ra。决定下下文的局部梯度计算  $D_1=Rd-Rb$ 、 $D_2=Rb-Rc$ 、 $D_3=Rc-Ra$ ，然后通过使能控制输出给下一个处理单元。



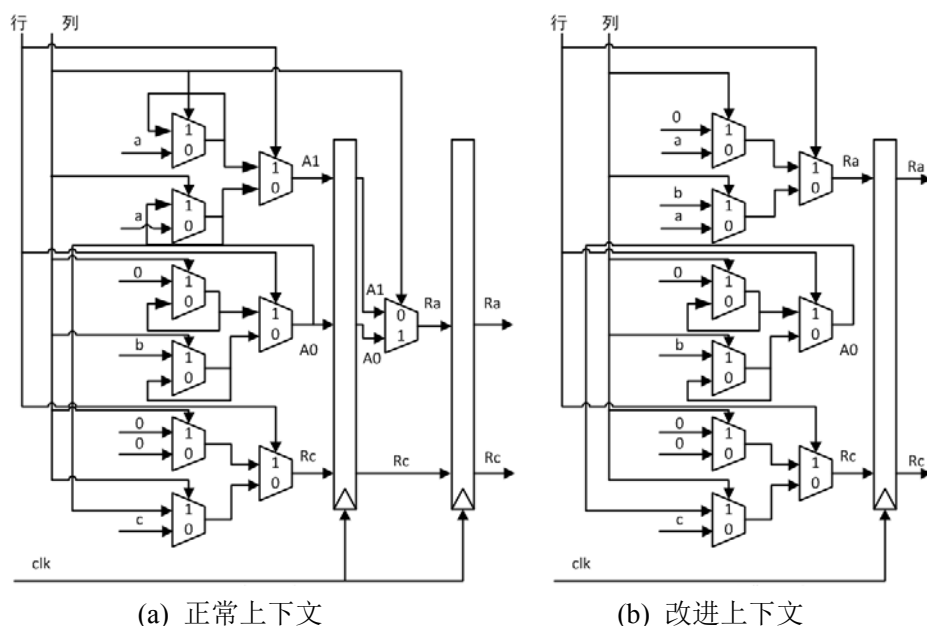


图 4.5 上下文建模

正常的上下文建模如图 4.5(a)所示。在第一个循环中，如果像素位于第一列，则  $a$  的重构值被计算并寄存在  $A0$  中，否则寄存在  $A1$  中。在第二个周期中， $A0$  或  $A1$  被分配给  $Ra$ 。在这个上下文中， $Ra$  的计算需要两个周期。重建的值  $Rb$ ,  $Rc$  和  $Rd$  需要同步到  $Ra$ ，并且需要三个寄存器用于同步。改进上下文建模如图 4.5(b)所示。改进的上下文中，如果像素位于第一列， $a$  的重构值存储在  $A0$  和  $Ra$  两者中，否则仅存储在  $Ra$  中。寄存器  $A1$  略去且  $Ra$  的计算只需要一个周期，处理延迟就会减少。 $Ra$ ,  $Rb$ ,  $Rc$ ,  $Rd$  的计算时间相同，不需要三个同步寄存器，从而节约了逻辑资源。

#### 4.1.2 预测模块设计

预测仅在正常模式下，即非常规模式下执行。预测模块主要完成功能边界检测预测、局部梯度量、梯度合并、预测误差及预测误差的计算等 HDL 的设计。预测模块编码过程示意图如图 4.6 所示。

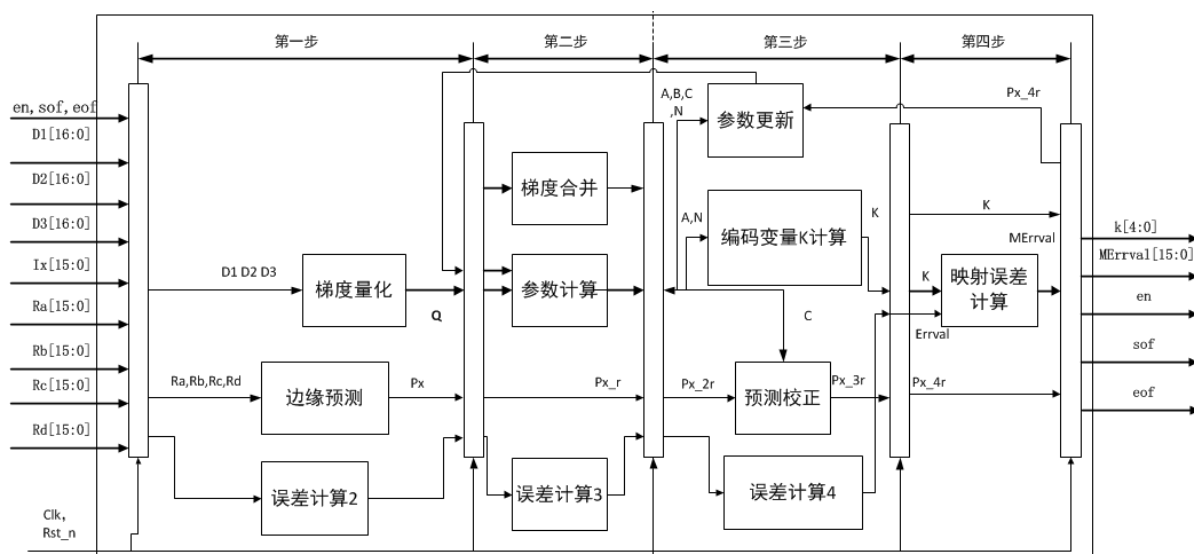


图 4.6 预测编码结构图

### (1) 边界检测预测

待编码像素的  $x$  的估计值  $P_x$  将由  $a$ ,  $b$  和  $c$  的位置的采样重建值  $R_a$ 、 $R_b$ 、 $R_c$  决定的。VHDL 中采用 function 函数实现 MAX 功能比较  $a, b$  的大小返回其中较大的一个，MIN 功能正好与之相反。

预测处理过程首先是对水平方向和垂直方向的边界检测。如果边缘预测检测出当前像素处于水平方向边界，那么预测值即为水平相邻采样像素位置  $a$  的重建值  $R_a$ 。如果边缘检测检测出当前编码像素处于垂直方向边界，那么同理预测值即为  $R_b$ 。如果没有当前检测出像素的位置以上的任何位置，那么当前采样像素位置  $x$  的预测值  $R_x$  为  $R_a + R_b - R_c$ 。如图 4.7 所示是这一处理过程。如果选择控制器条件  $R_c \geq \text{MAX}(R_a, R_b)$ ，则把  $\text{MIN}(R_a, R_b)$  赋值给  $P_x$ 。如果  $R_c \leq \text{MIN}(R_a, R_b)$ ，则把  $\text{MAX}(R_a, R_b)$  赋值给  $P_x$ 。如果两种情况都不是则  $R_a + R_b - R_c$  赋值给  $P_x$ 。

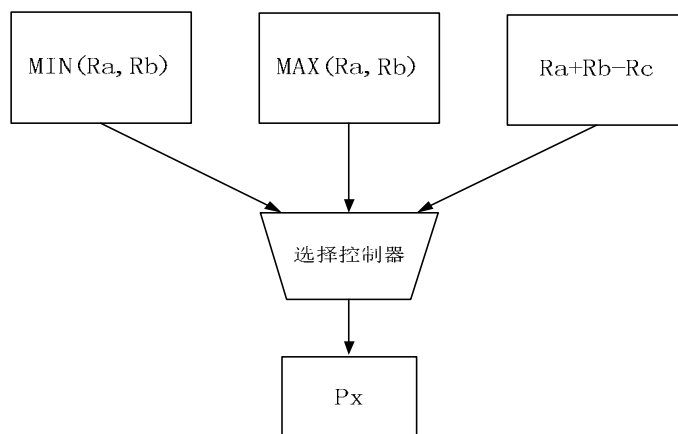


图 4.7 边缘检测预测器

## (2) 梯度量化与合并设计

局部梯度量化的设计同样采用 VHDL 的 function 函数 GETQI 来描述对梯度 D1、D2、D3 量化，其中 T1、T2 和 T3 为局部梯度的量化阈值，取值为编码参数初始化处理过程给出的值分别为 3、7 和 21。接下来我们就要对量化梯度 Q1、Q2 和 Q3 的符号进行校正，使得后续模块在产生上下文是正整数输入。具体的实现过程：通过量化后梯度之间的关系，使用 Qi 最高位作为标志 SIGN，如果最高位为“1”则 SIGN=-1，如果最高位为“0”则 SIGN=1，使用这样的方法是考虑到负数的二进制表示最高位为 1 的补码。然后通过 SIGN 去校正量化梯度的符号。梯度合并就是把梯度量化后得到的 Q1、Q2 和 Q3 一对一的映射到 Q 的具体函数，得到一个不大于 365 的整数且该映射对(Q1、Q2、Q3)所有值都有定义。实现逻辑采用简单的比较器即可完成。

在局部梯度量化模块中，经常根据图 4.8 中指定的过程对 D1，D2 和 D3 进行量化。为了实现这样的效果，使用非负阈值 T1，T2 和 T3。根据它们与阈值的关系，得到一个区域号 Qi（分别为 Q1，Q2 和 Q3）。包含 8 个判断等待时间和一个等待时间等。其中 Di 的最高位（MSB）置‘1’时所映射的 Qi 为负数。

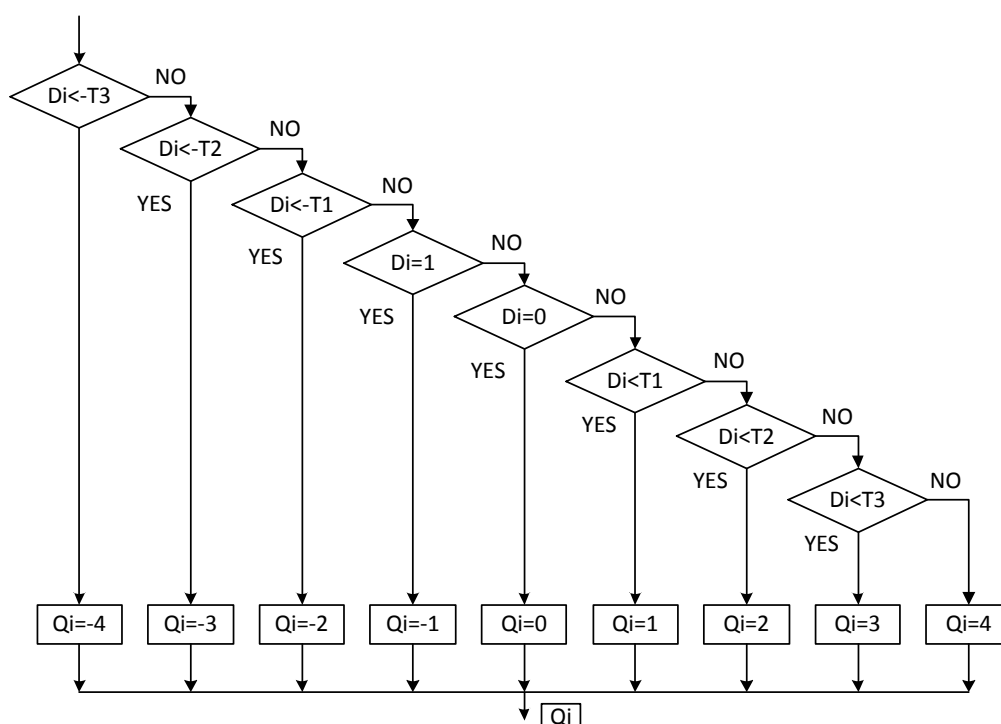


图 4.8 局部梯度量化

## (3) 预测校正及预测误差计算设计

在计算出 Px 后，预测值必须根据过程图 4.9 所示进行校正，此后的 SIGN 须根据上一节中的 SIGN 正负号变化来设计，所得到新的 Px 值必须在[0...MAXVAL]范围内。预

测校正值  $C[Q]$  由变量更新来推导。如果  $SIGN=1$  时，选择控制器就会选通  $P_x+C[Q]$  赋给  $P_x$ ，否则  $P_x-C[Q]$  赋给  $P_x$ 。MAXVAL 会选择当  $P_x>MAXVAL$  时赋值给  $P_x$ ，否则  $P_x=0$ 。

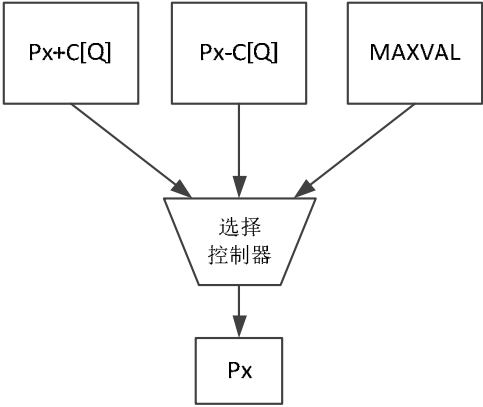


图 4.9 预测校正

利用上述预测校正后得到的  $P_x$  的预测值后，预测误差 Errval 的值通过计算我们就能得出，在计算前我们还得讨论下编码变量符号的问题，如果上述编码变量  $SIGN\_r2$  指延迟两个时钟周期后给出的符号为“1”，相应的预测误差也要取反。位置  $x$  值为  $I_x$  采样预测误差计算过程及预测误差按模减过程。其中， $RANGE0$  表示预测误差表示的范围， $HALF\_RANGE$  代表是  $RANGE0$  的二分之一。计算的得到的预测误差必须减去与编码相关的范围。Errval 的最高位 '1' 表示 Errval 为负值，否则表示为正值。

JPEG-LS 算法存在两个反馈逻辑环路，第一个是上下文参数组的更新是前后像素具有较强依赖关系的；第二个是近无损压缩模式下重建值  $R_a$  和  $I_x$  是强依赖关系的， $R_a$  是上一个  $I_x$  计算所得的重建值。假设 FPGA 不对图像进行帧缓存，信源发送过来的像素一般都是前后像素紧密相关的，那么意味着需要在一个时钟周期内就必须完成大量的串行逻辑计算，显然这样会使得整个 JPEG-LS 硬件编码模块的最高时钟频率大大受约束。在使用 xilinx 的 FPGA 做了测试，不做优化情况下按照 C 实现的算法串行逻辑进行设计，最高时钟频率大约在 36MHz 左右。严重制约了硬件模块的可用性。

针对上述问题，使用前向预测的方式来打破串行逻辑并且在资源占用上更少。前向预测机制，是针对预测误差  $P_x$  修正时的计算需要使用  $C[Q]$  的值，而  $C[Q]$  的值在一次像素编码时，只可能有三种变化情况：-1、0、+1。那么使用三个分支，将这三种情况全部进行计算，然后在下级流水线进行选择，可以理想的打破这个串行逻辑，提高综合主频。如下图 4.10 所示为  $C[Q]$  更新进行改进处理的过程示意图。

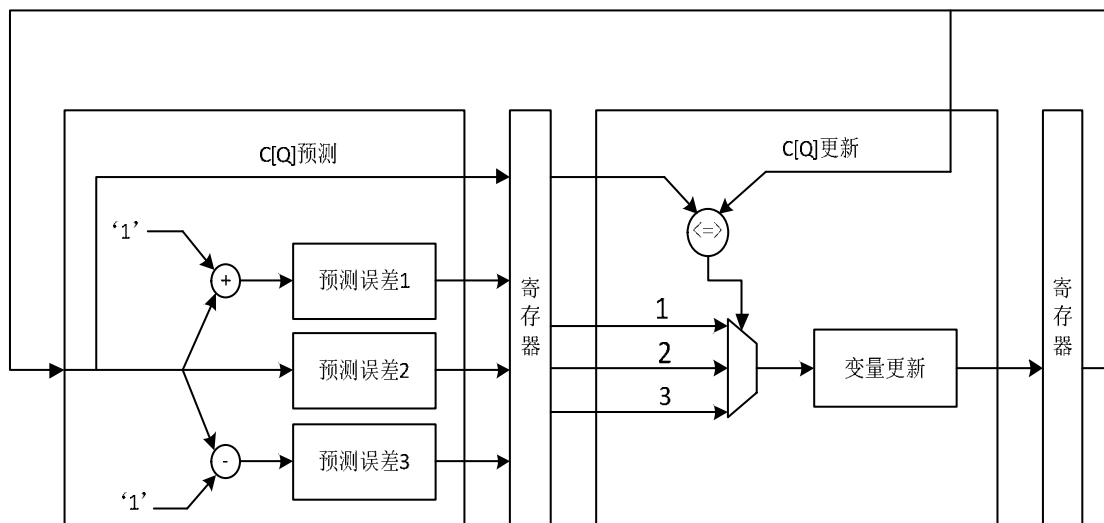


图 4.10 C[Q]更新的处理图

最终 FPGA 的硬件模块设计运行频率可达 41MHz, 能够对摄像头输入的  $1920 \times 1080$  的图像数据进行实时压缩, 无损压缩率大约在 2.5 倍左右。

#### (4) 变量的更新及 $K$ 值生成设计

预测编码的最后一步是在正常模式下对变量 A、B、C 和 N 的更新。值得注意的是, 变量的更新必须在编码过程的最后, 在变量 K 和 MErrval 被计算后更新。在本设计中, 采用 FPGA 内部的 Block Memory 用代码生成 4 个可读写的片内 RAM 分别给 A、B、C 和 N, 通过操作对 RAM 控制器的来完成对每个 RAM 的读写操作, 从而实现在编码过程中对变量数组的更新。实现方法: 将生成的四个 RAM 定义为和四个变量数组同样的深度, 本文中 RAM 的地址深度为 8, 数据位宽为 32 位; RAM 的地址和变量数组的下标与上下文 Q 一一对应; 这样, 对变量数组的使用读取和使用后的更新写入 RAM 中, RAM 的写地址作为当前上下文 Q 的索引值, 使用时读取相应 RAM 的地址去寻址该存储位置即可。实现流程如图 4.11 变量更新过程示意图。

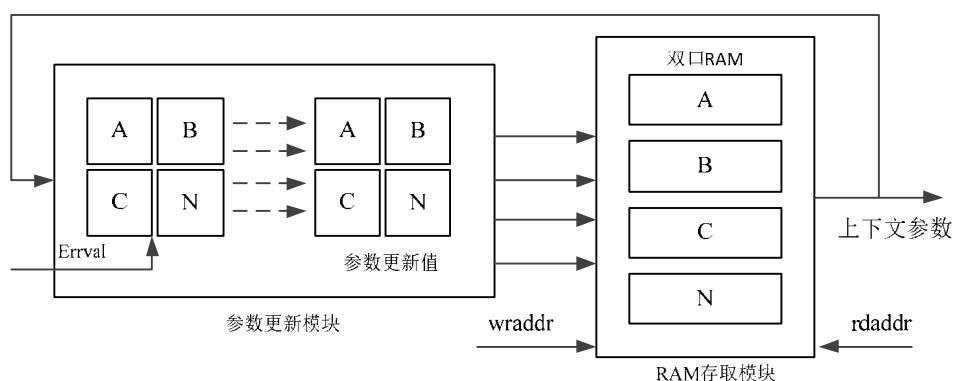


图 4.11 变量更新过程

Golomb 编码中编码变量  $K$  使用  $N[Q]$  和  $A[Q]$  来计算来查找。 $A[Q]$  和  $N[Q]$  移位比较图如图 4.12 所示。

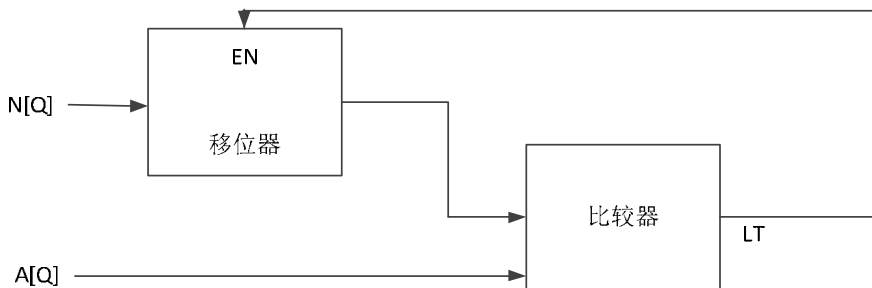


图 4.12  $A[Q]$  和  $N[Q]$  移位比较图

变量  $K$  的计算使用 `when` 语句  $N[Q]$  移位计算和  $A[Q]$  进行比较然后得出  $K$  值如图 4.13 所示  $K$  值产生的状态机。当“00000000000000000000”& $N[Q] \geq A[Q]$  时  $K$  取值为 00000，同理  $N[Q]$  左移 1 位和  $A[Q]$  比较状态机就跳转到下一个状态。所以  $K$  值随着时钟的上升沿在转移到下一个状态，相应  $K$  的取值也会随状态的改变而相应的改变。

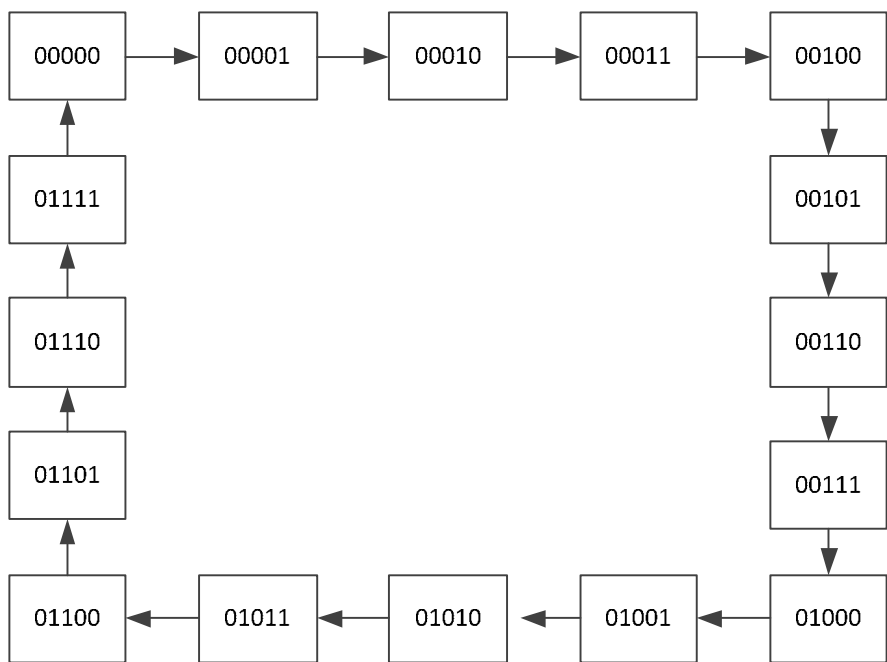


图 4.13  $K$  值产生状态图

### 4.1.3 Golomb 编码模块设计

Golomb 熵编码由误差映射  $MErrval$ 、 $K$  值输入、映射误差  $MErrval$  编码模块三部分

组成。误差映射模块和  $k$  值产生器实现比较简单，重点介绍 Golomb 编码模块的实现过程。该模块主要根据级预测编码模块输出的 MErrval 值以及  $k$  值进行 Golomb 编码，同时对 Golomb 编码输出的码流进行封装操作，当输出码流拼满 32bits 后，将该数据写入 FIFO 中，只要该 FIFO 非空，则从其中从低到高读出数据，作为最终输出码流。VHDL 设计实现按照如图 4.14 所示 Golomb 编码的示意图。

图 4.14 Golomb 编码示意图

具体的编码规则如下:

(2) 如果进入了最长码字编码方式, 先编移位寄存器中当前存放值的大小进行计数, 然后输出该计数值大小个码元‘0’; 然后输出一个码元‘1’; 最后将  $k$  比特寄存器中的数据减去 1, 扩展成 qbpp 位二进制表示的形式输出。

## 4.2 仿真与验证

本节针对 VHDL 实现的 JPEG-LS 无损压缩算法编写测试文件，主要作为输入信号的激励源，将图像的数据用软件 HexEditcansave 打开并复制图像数据，其中图像数据以十六进制的方式显示。将图像数据复制到 source.txt 文件中以供仿真测试调用，此仿真测试采用随机测试数据填写到 source.txt 文件。仿真测试使用 ISE 14.7 和 Modelsim 10.4 联合仿真测试。仿真测试如图 4.15 所示。采用 Xilinx ISE 环境下的 XST 综合工具对顶层模块进行综合图 4.16 所示为 ISE14.7 软件的综合报告，报告内容包含工程状态信息，逻辑资源的使用情况和详细报告等情况。

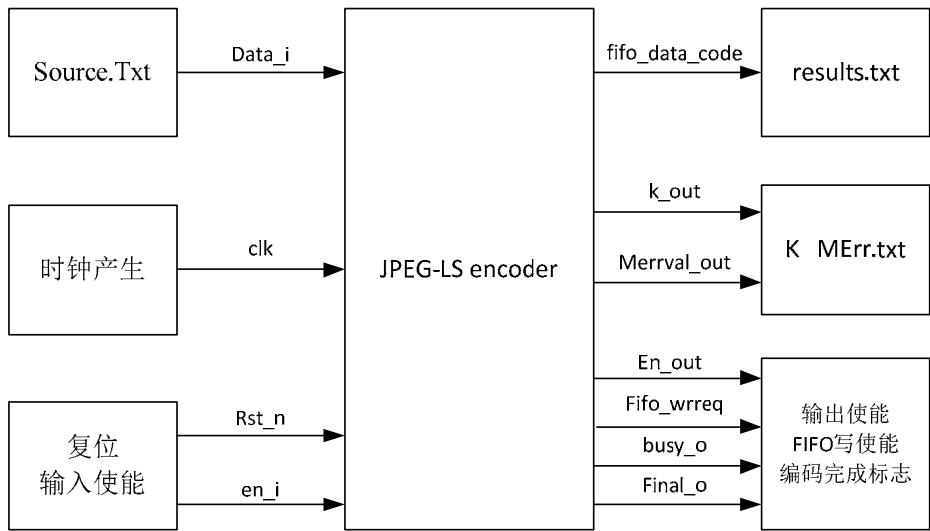


图 4.15 仿真测试框图

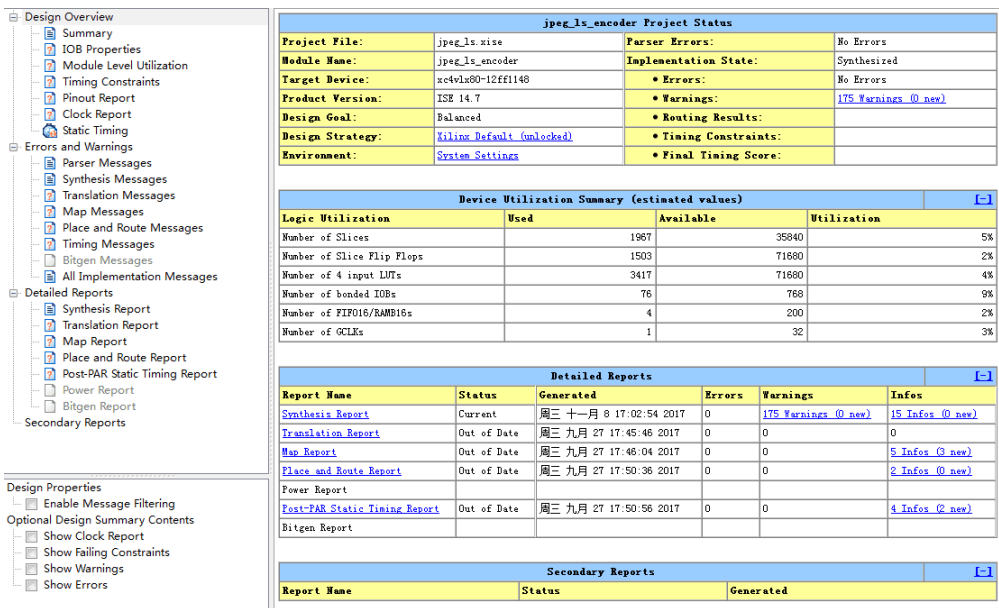


图 4.16 ISE 综合报告图



为了验证 VHDL 设计的 JPEG-LS 无损压缩算法编码的正确性, 采用如图 4.17 测试数据测试 JPEG-LS 编码器的顶层模块。将编码得到的数据写入到 results.txt 文本文件中, 参数 K 值和映射误差 MErrval 写入到 k\_MErr.txt 文本文件中。Xilinx ISE14.7 和 Modelsim 10.4 联合仿真结果如图 4.18 所示。仿真结果的数据如图 4.20 所示。采用测试时钟的时钟周期为 10ns 即 100MHz, 从图 4.18 可以看出完成编码所用时间仅为 0.028ms, 能够满足对图像的高速实时压缩。

图 4.17 仿真测试数据

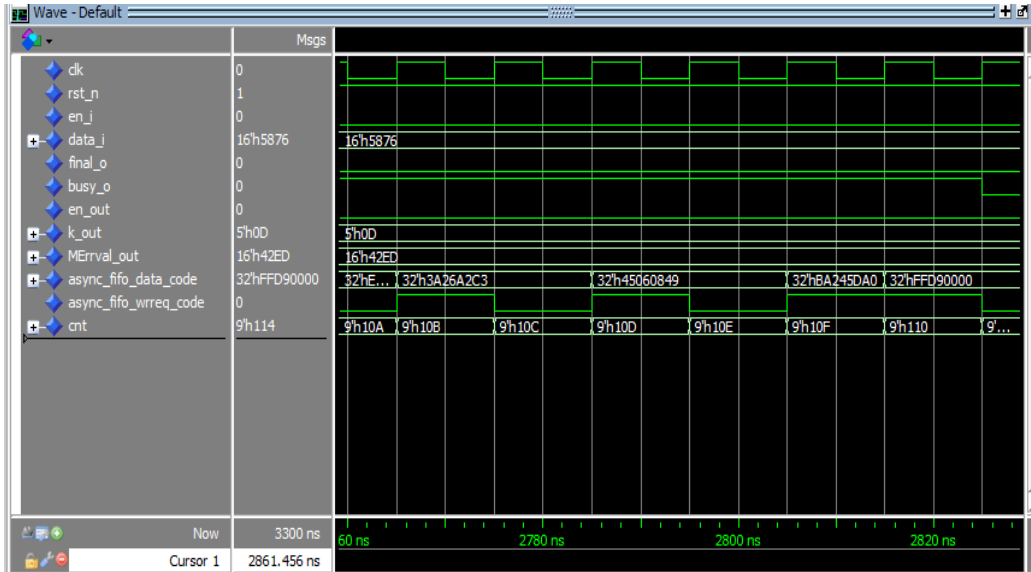


图 4.18 modelsim 10.4 仿真时序图

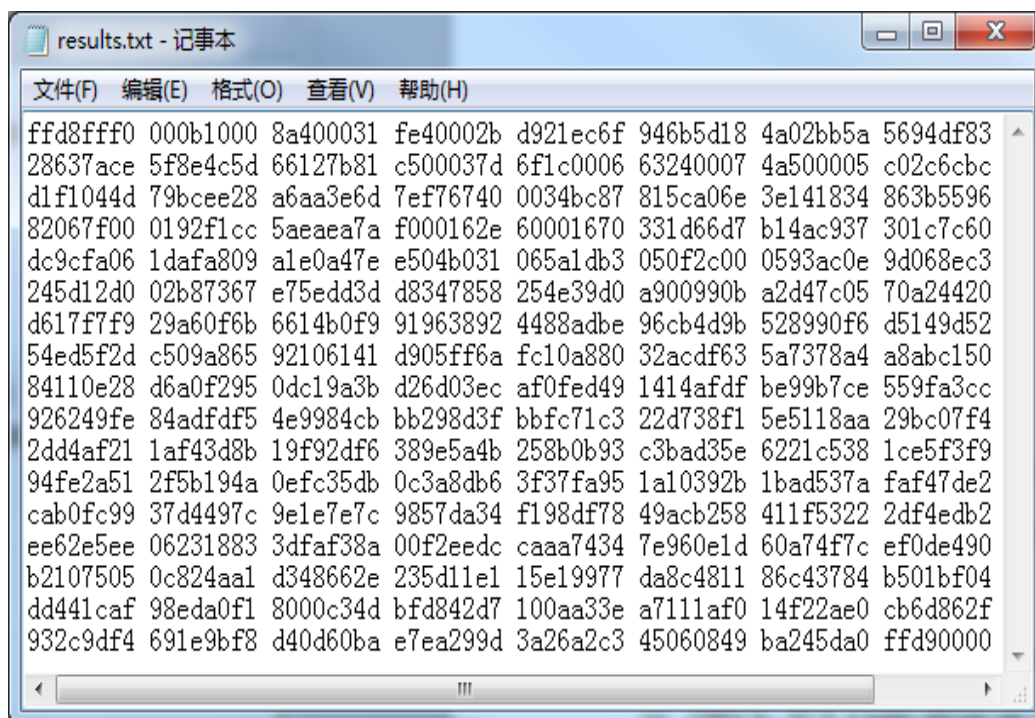


图 4.19 仿真结果

为了验证仿真结果的正确性需要对压缩的数据进行解压，解压的方式通过软件来解压。通过软件解压后的图像数据如图 4.20 所示。通过用软件比较工具 Beyond Compare 3 来比较两组数据结果正常。说明编码器压缩编码图像的正确可靠性。

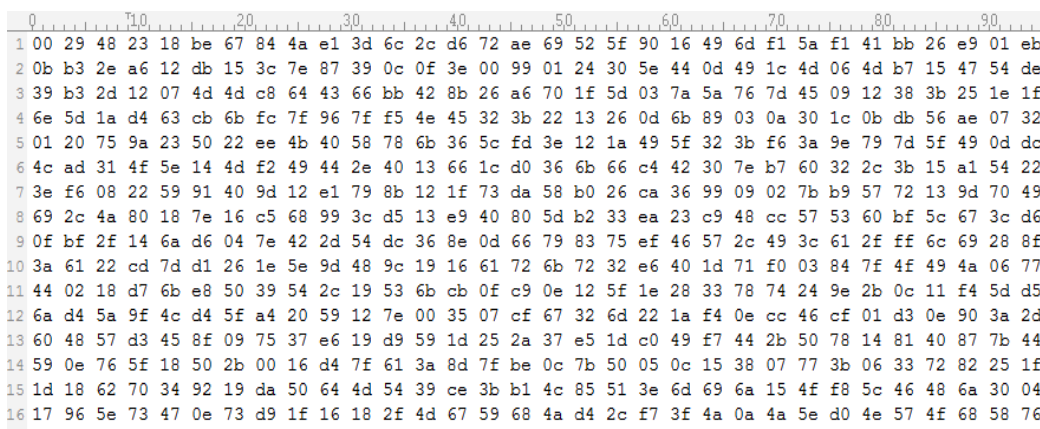


图 4.20 解压后的图像数据

为了更直观对比原始图像和压缩图像，采用 4096×3600 的 SAR 图像如图 4.21(a)所示加载到图像激励源中，通过 Cameralink 图像采集处理单元进行测试验证 JPEG-LS 无损压缩编码器。无损压缩率大约在 2.5 倍左右，无损压缩的图像效果如图 4.21(b)所示。



(a) SAR 图像原图像

(b) SAR 图像压缩图像

图 4.21 原始图像与压缩图像

通过对比图 4.21 两幅图像可以看出，JPEG-LS 无损编码器达到了无损压缩效果。从图像质量上可以看出压缩图像和原始图像质量保持一致。说明了 JPEG-LS 无损压缩编码器实现了高保真度和实时无损压缩从而使低复杂度高性能有效统一。因此本章采用 FPGA 设计 JPEG-LS 无损压缩编码器经过仿真和实际工程的测试验证都符合既定要求。

### 4.3 本章小结

本章介绍了使用 FPGA 设计实现 JPEG-LS 无损压缩的编码的全过程。首先介绍了 FPGA 设计开发的流程。其次，介绍了通过 VHDL 设计 JPEG-LS 无损压缩算法的各个模块设计实现过程。最后，对 VHDL 设计的 JPEG-LS 无损压缩编码器进行验证，通过 Xilinx ISE14.7 软件和 modelsim 10.4 联合仿真测试编码器的时序仿真图验证了该编码器编码的正确性和压缩效率说明了编码器可工程化使用。结果表明，压缩数据和解压缩数据无异常，且编码比特流稳定，恢复图像与原图像无异常，运行频率达到 41MHz。

## 结 论

随着空间应用技术的发展,航空航天领域和医学领域上都会使用图像的压缩处理等功能。JPEG-LS 无损图像压缩算法具有硬件实现复杂度低、保真度高和高压缩效率等特点,对于实际应用都具有很重要意义,因而广泛应用于卫星遥感、工业测量、医学图像等领域。

本文首先对 JPEG-LS 背景意义及国内外的研究发展的现状做了简单的介绍。然后对 JPEG-LS 无损压缩算法理论进行了研究和分析为后续的算法 FPGA 实现做了理论铺垫。在对 Cameralink 协议进行分析后基于 FPGA 硬件设计 Camerlink 图像采集处理系统,针对系统要求做出详细的系统方案设计并且针对系统内的各个模块进行 FPGA 设计实现。实际验证效果表现系统的稳定性好,图像采集处理效率高、鲁棒性高和对图像实时压缩等特点,说明了系统的可实际工程使用性。最后,设计了基于 FPGA 实现了 JPEG-LS 无损压缩算法,在设计中使用思路灵活、且稳定的 VHDL 语言进行模块化设计。在基于 FPGA 设计 JPEG-LS 无损压缩编码器里面包括上下文建模模块、预测模块和 Golomb 模块的设计,对上下文建模的逻辑改进设计和 C[Q]更新时的改进处理把编码器的时钟频率提高到 41MHz,同时 JPEG-LS 无损压缩编码器也是 Cameralink 图像采集处理里面的图像的无损压缩处理实现方法。最后进行了仿真验证说明 JPEG-LS 无损压缩编码器的编码效率很高、实时数据压缩处理以及低复杂度易于硬件实现等特点。

虽然本文实现了基于 FPGA 实现 JPEG-LS 无损压缩算法的设计,但是还有很多不足之处。

(1) JPEG-LS 无损压缩编码器还没实现游程编码设计,虽然在实际过程中游长编码的有无对编码器的影响不会很大,但是有游长编码的效果比没有要稍好一点。

(2) 对很多场合而言压缩图像需要控制压缩比和码率控制等要求,本文基于 FPGA 设计 JPEG-LS 无损压缩编码器还是有待改进和完善的。

## 附录 I

DDR3A ADD[15:0], 控制线和时钟等长。  
最大走线误差10MILS。

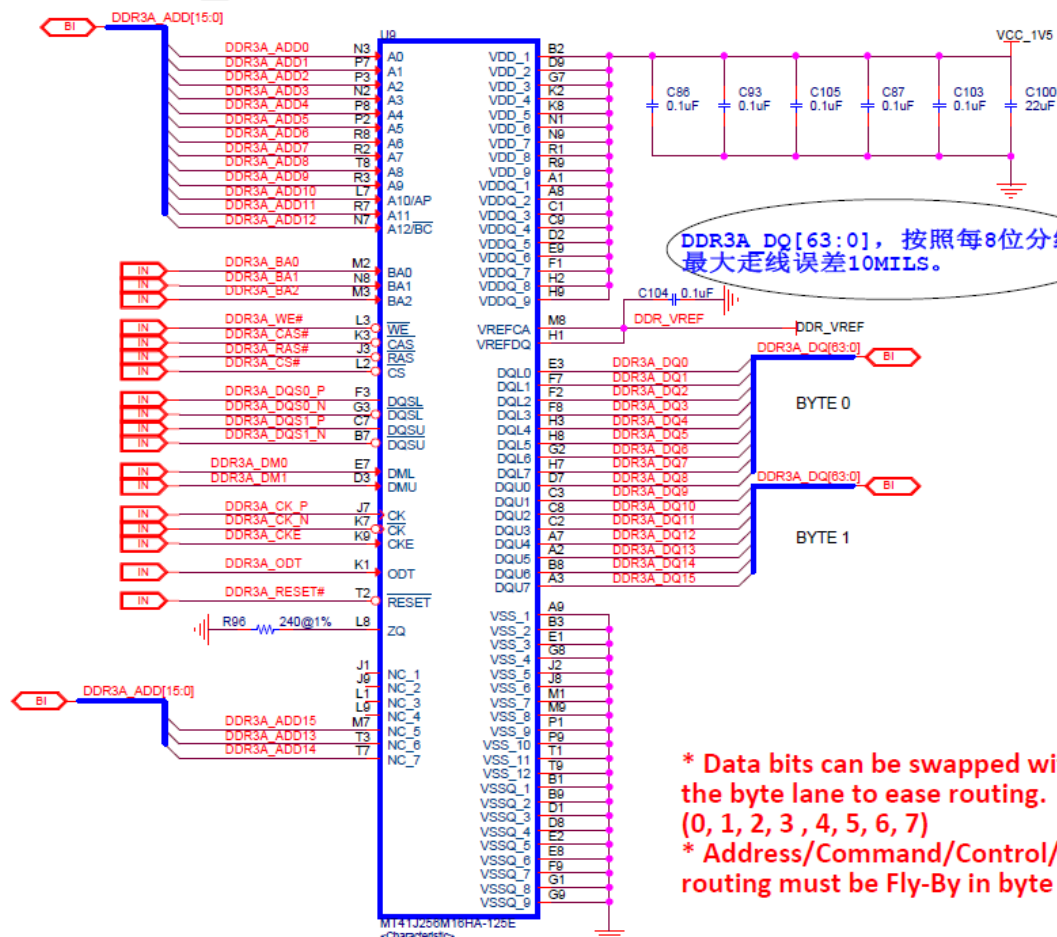


图 I.1 DDR3 电路设计原理图

## 参考文献

- [1] 吴乐南,徐孟侠.数据压缩的原理与应用[M].北京:电子工业出版社,1995.
- [2] 吴乐南.数据压缩(第三版)[M].北京:电子工业出版社,2012
- [3] ISO/IEC,Information technology-Coded representation of picture and audio information- progressive bi-level image compression,International Standard ISO/IEC 11544 and ITU-T Recommendation T.82, Mar.1993.
- [4] ISO/IEC, Information technology-Lossless and near-lossless compression of continuous-tone still images, International Standard ISO/IEC 14495-1 and ITU Recommendation T.87, 1999.
- [5] Itu-T I. Information technology-JPEG 2000 image coding system: Core coding system[J]. ITU-T Recommendation T.800 and ISO/IEC International Standard, 2002, 21(1):27-51.
- [6] MIAOU S G,KE F S,CHEN S C. A lossless compression method for medical image sequences using JPEG-LS and interframe coding[J].IEEE Trans. Information Technology in Biomedicine, 2009, 1(5): 818-821.
- [7] 徐燕凌,刘蓓.JPEG-LS 图像压缩动态码率控制策略[J].计算机工程,2008,34(7):238-239.
- [8] 李晓飞.Huffman 编解码及其快速算法研究[J].现代电子技术,2009,32(21):102-104.
- [9] Weinberger M J, Seroussi G, Sapiro G. LOCO-I: a low complexity, context-based, lossless image compression algorithm[C]// Conference on Data Compression. IEEE Computer Society, 1996:140.
- [10] 陈益刚,邓家先,谢凯明.基于 LOCO-I 算法的图像压缩比控制[J].电视技术,2015,39(02):57-60.
- [11] 田耘,徐文波,胡彬.Xilinx ISE Design Suite 10.x FPGA 开发指南,逻辑设计篇[M].北京:人民邮电出版社,2008.
- [12] 张天序,邹胜,曾永慧.基于 FPGA 的图像无损压缩算法的实现[J].系统工程与电子技术, 2004,26(10):1340-1343.
- [13] M. Weinberger, G. Seroussi, and G.Sapiro. "The LOCO-I lossless image compression algorithm: Principles and standardization into JPEG-LS,"IEEE Trans. Image Processing, vol. 9, pp. 1309-1324, Aug.2000.
- [14] Shantanu D. Rane and Guillelmo Sapiro, "Evaluation of JPEG-LS, the New Lossless and Controlled-Lossy Still Image Compression Standard, for Compression of High-Resolution Elevation Data", IEEE Transactions on Geoscience and Remote Sensing, vol. 39, pp. 2298- 2306, October, 2001.
- [15] Savakis A, and Piorun M. "Benchmarking and hardware implementation of JPEG-LS," 2002 International Conference on Image Processing.IEEE, 2:11-949-11-952, vol. 2, 2002.
- [16] D. Brunello, G. Calvagno, G. Mian, and R. Rinaldo."Lossless compression of video using temporal information," Image Processing, IEEE Transactions on,vol. 12, no. 2,pp. 132-139,Feb. 2003.
- [17] L.-J. Kau and Y.-P. Lin, "Adaptive lossless image coding using least squares optimization with edge-look-ahead," IEEE Trans. Circuits and Systems II, Vol. 52, No. 11, pp. 751-755, Nov. 2005.
- [18] M. Papadonikolakis, V. Pantazis, and A. P. Kakarountas, "Efficient high-performance asic implementation of jpeg-ls encoder," in Proceedings of the conference on Design, automation and test in Europe, pp. 159-164, EDA Consortium, 2007.Kunt M, Ikonomopoulos A, Kocher M. Second-generation image-coding techniques[J]. Proceedings of the IEEE, 2005, 73(4):549-574.
- [19] M. E. Papadonikolakis, A. P. Kakarountas, and C. E. Goutis, "Efficient high-performance 15 implementation of jpeg-ls encoder," Journal of Real-Time Image Processing, vol. 3, pp. 303-310, 2008.NURBS reconstruction of digital terrain for hydropower engineering based on TIN model[J].Progress in Natural Science,2008(11):1409-1415.
- [20] Pierantonio Merlino, Antonio Abramo; "A Fully Pipelined Architecture for the LOCO-I Compression Algorithm,"IEEE Transactions on Very Large Scale Integration (VLSI) Systems,2009,17(7):967-971.
- [21] S. Wahl, H. A. Tantawy, Z. Wang, et al. "Exploitation of context classification for parallel pixel coding in JPEG-LS", Image Processing (ICIP) 2011 18th IEEE
- [22] Wang Z, Klaiber M, Gera Y, et al. Fast lossless image compression with 2D Golomb parameter adaptation based on JPEG-LS[C]// Signal Processing Conference. IEEE, 2012:1920-1924.
- [23] Sheorey S, Firl A, Wei H, et al. Adaptive Prediction with Switched Models[C]// Data Compression

Conference. IEEE, 2015:471-471

- [24] 程子敬.遥感图像高保真高速实时压缩技术研究[D].北京:北京航空航天大学,2001.
- [25] 曹青,吴乐南.利用 JPEG-LS 高效无损压缩气象卫星云图数据[J].应用气象学报,2002(03):380-382.
- [26] 吴美建,林行刚.一种改进的遥感图像无损压缩 JPEG-LS 算法[J].中国图象图形学报, 2003,8(5):596-600.
- [27] 王晋,张晓玲,柴焱等.一种基于自适应预测的高光谱图像近无损压缩方法[J].计算机应用研究, 2007(05):305-307.
- [28] 孟宪元,钱伟康.FPGA 嵌入式系统设计[M].北京:电子工业出版社,2007.
- [29] 吴镇扬.数字信号处理基础[M].北京:高等教育出版社,2016.
- [30] 马秋芳.关于离散无记忆信源的最佳编码问题[J].江汉石油学院学报,1987(02):51-57.
- [31] Majid Rabbani, Paul W. Jones, "Digital image compression techniques", Volume TT 7, Donalds C. O'Shea, Series Editor Georgia Institute of Technology, 1991, pp 49-92.
- [32] Katherine Bouman, Vikas Ramachandra, Klain Atanassov, et al. "RAW camera DPCM compression performance analysis", SPIE-IS&T Electronics Imaging, SPIE Vol.7867, 78670N, 2011.
- [33] D.G Daut, D.Zhao and J.C.Wu "Double predictor differential pulse code modulation algorithm for image data compression", Optical Engineering, Vol. 32, no. 7, pp.1523 1993.
- [34] 邹云伟,李冰.动态查找表设计方案研究[J].电子与封装,2007,No.5612:15-18.
- [35] CCSDS 121.0-B-1, Lossless Data Compression, CCSDS,1997.
- [36] JPEG2000: Image Compression Fundamentals, Standards and Practice, D. Taubman, M. Marcellin, Kluwer Academic Publishers, 2002.
- [37] A. Said and W. Pearlman, "A New, Fast, and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees," IEEE Transactions on Circuits and Systems for Video Technology, vol. 6, no. 3, pp. 243-250, June 1993.
- [38] M. Rabbani, R. Joshi, "An overview of the JPEG2000 still image compression standard", Signal Processing Image Communication, Eurasip, Volume17, No. 1, pp. 3- 48, January 2002.
- [39] DING J J, CHEN H H, WEI W Y. Adaptive Golomb code for joint geometrically distributed data and its application in image coding[J]. IEEE Trans. Circuits and Systems for Video Technology, 2013, 23(4): 661-670.
- [40] 胡栋.静止图像编码的基本方法与国际标准[M].北京:北京邮电大学出版社,2003.
- [41] 吴乐南.静止图像压缩标准新进展[J].电脑应用技术,1998(42):8-11.
- [42] 胡华,徐端颐,胡恒.光存储系统中的游程长度受限编码[J].光学技术,2006,32(3):4-7+10.
- [43] 张德联,张帆.基于 CameraLink 的高速数据采集压缩系统[J].科学技术与工程,2008,18:5253-5255.
- [44] 王智,罗新民.基于乒乓操作的异步 FIFO 设计及 VHDL 实现[J].电子工程师,2005,31(6):13-16.
- [45] 荆旭.基于 FPGA 的 SATA 协议分析及收发控制器设计[D].西安:西安电子科技大学,2014.
- [46] 卢毅,赖杰.VHDL 与数字电路设计[M].北京:科学出版社,2002.
- [47] 魏亚辉.基于 FPGA 的遥感图像 JPEG-LS 压缩算法的研究与实现[J].信阳农林学院学报, 2016,26(2):107-110.
- [48] J.-C. Wu and D.G Daut "Adaptive nonstationary DPCM image coding with variable blocksize", SPIE Proceedings of the 1997 Visual communications and image processing, vol. 3024, pp.447-458 1997.
- [49] Nazar F, Murugan S. Implementation of JPEG-LS compression algorithm for real time applications[C] // International Conference on Electrical, Electronics, and Optimization Techniques. IEEE, 2016: 2772-2774.
- [50] 戴鑫.JPEG-LS 图像无损压缩 IP 核的 FPGA 设计[D].太原:太原理工大学,2010.

## 致 谢

随着时间的飞速流逝，我们即将迎来毕业季。在短短的两年半时间里我收获了人生中最难得可贵的经历。我们在这个阶段收获了来自老师的知识，同学之间的友谊，朋友的勉励等，因此我在这里表达感谢之意。

首先，感谢张雷老师对我学术上的指导和生活工作上的关心，能跟着张老师一起完成我的研究生学业是一件非常值得骄傲的事。张老师在学术上的造诣值得我认真的学习，认真负责的态度是我今后从业的楷模。张老师在我论文这方面一直很有耐心和悉心的指导，生活工作的事也一直在帮助我，在这里衷心的感谢您张雷老师。

其次，感谢我们班的同学对我学习上的帮助，生活上的关心。跟你们在一起学习和生活是永远值得怀念的事。我会永远记得这份同学情谊，谢谢你们我的同学们。

最后，感谢我的父母对我学业上的支持，没有你们的支持就没我的今天。我会一直记在心里。

李长兴于沈阳航空航天大学

2018 年 01 月 10 日



## 攻读硕士期间发表（含录用）的学术论文

- [1] 张雷, **李长兴**. 基于 JPEG-LS 低复杂度近无损算法的码率研究. 计算机应用研究.(已录用)