# 摘 要

视频图像数据间有极强的相关性，也就是说有大量的冗余信息。视频编码就是在最大程度保持视频质量的前提下，将数据中的冗余信息去掉，以较少的数据量表示被编码的信号源。冗余可分为空域冗余、时域冗余、统计冗余等。为了去除空域的冗余信息，人们采用了帧内编码技术；为了减少时域的冗余信息，人们则采用了帧间编码技术；而为了减少统计冗余信息，则采用了熵编码技术。其中，熵编码技术是一种无损压缩技术。它对变换、量化后得到的系数和运动信息，进行进一步的压缩。

AVS 标准采用了一种自主知识产权的熵编码技术，对帧内编码和帧间编码后的语法元素和残差信息进行进一步的压缩。AVS 熵编码中采用的主要编码方式主要有：固定长度编码，K-阶指数哥伦布编码和基于内容 2D-VLC 编码。熵编码器处理各种不规整的语法元素，在每个宏块周期内处理的数据量也会随着每个宏块的编码方式的不同而变化，并且前后数据间具有严格的时序性和依赖性。设计者在设计熵编码结构时不仅需要考虑最坏情况下的处理性能，而且还要考虑设计后的运行速度和硬件资源消耗，这往往会给设计者带来很大的挑战。

首先，本文针对 AVS 标准熵编码技术提出一种高速低硬件资源消耗的适合高清视频实时编码的熵编码结构。通过将整个熵编码器分成两级流水线结构，使产生宏块头参数和编码宏块头参数处于不同的流水级，消除了等待时间，同时合理分配了编码时间。每级流水线内部采用了单路流水线硬件结构，与传统的多路并行结构相比，节省了大量硬件资源；整个变长编码模块采用了多级流水线结构，完成一级操作只需要 1 个时钟。因此，该熵编码器能够在每个时钟内完成一次变长编码操作。顺序编码宏块头信息和残差系数，复用一套哥伦布编码和码流拼接装置，节省了硬件成本，提高了资源利用率。提出了一种统一地址编排方法，对不会同时使用的表格进行编排，如变长编码码表、CBP 映射表、帧场扫描地址映射表，减少了 ROM 的使用数量。

其次，本文同时设计了一种高速多级流水线结构的熵解码器，每个时钟解码一个语法元素，能在 150MHz 的频率下实现对高清视频码流解码。支持序列级、

帧级、宏块级和块级语法元素解码，实现固定长度解码、哥伦布解码和 2D-VLC 解码。本文在设计时，为了打断变长解码过程中前后系数间的依赖性，实现一个时钟内解码一个系数，采用了一种基于查找表结构的码流移位电路。该移位电路每个时钟能够分离出一个码字，支持流水线操作，并且能在很高的频率下工作。

本文利用 Verilog HDL 对提出的熵编码器和熵解码器结构进行 RTL 描述，在 Modelsim SE6.3 下仿真通过，最终在 Xilinx Vertex5 VLX330 FPGA 上综合实现。其结果表明，本文提出的结构是有效的，完全满足 AVS 高清视频的编解码器的性能要求。

**关键词：**熵编码，VLC，VLD，AVS

# Abstract

There is close correlation among neighbouring pixels in pictures or videos, that is, much redundancy exists in video and picture data. So human has been engaged in working out efficient ways to eliminate or minimize those redundancies. As a result, high performance video encoders have been developed and serving in peple's life nowadays. Generally speaking, the redundancy can be divided into spatial redundancy, temporal redundancy, statistical redundancy and so on. Intra Prediction and Inter Prediction have been widely used to reduce spatial and temporal redundancy. For staticstical redundancy, another important technique, Entropy Coding, has been adopted to further remove redundant information. Entropy Code is a kind of lossless compression techenique and deals with various parameters and residual coefficients after transformation and quantification.

AVS (Audio and Video coding Standards) is developed in China independently and uses its own Entropy Coder to arrange syntax elements after Intra Prediction and Inter Prediction. There are about three coding methods in AVS entropy coder, Fixed Length Code, signed and unsigned Exp-Golomb Code and 2D Variable Length Code (2D-VLC). As we know, Entropy Coder deals with different kinds of irregular syntax elements. The type and number of those elements are variable in different Macroblock (MB). What's more, these elements need to be arranged serially and some of them have strong dependency on others. So, besides possible maximum number of elements, Entropy Coder designers have to take the working frequency and hardware cost into account.

This paper firstly introduces AVS Entropy Code in details and analyses overall process of Entorpy Code. Then it analyses the timing requirements of video encoder and decoder and presents several architectures for AVS Entorpy encoder and decoder. After carefully weighing factors such as requirements, throughput, speed and hardware cost, we choose to use a single-way full-pipeline architecture as the final one. In the following part, this paper provides detailed microarchitecture and main submodules in our design and gives simulation and implementation reports for each design. In general, our work can be summarized as follows.

First, an efficient architecture with high speed and low hardware cost for AVS real-time HD Video Encoder has been presented in our paper. In order to eliminate the dependency between encoding MB header syntax element and Zigzag Scan circuit, we devide the entropy coder into two pipeline stages. In each stage, we use a single way of pipeline architecture for 2D-VLC operation, which is much different from common multiple parallel structure.With a throughput of one symbol per clock, the structure surely meets the demands of HD video encoder. In this way, we can save lots of hardware resources and efficiently utilize serializability of syntax elements. Meanwhile, a very practical method for compressing VLC tables in AVS has been proposed to reduce the number of RAMs when we implement these tables in hardware.

Then, we design a high-speed multi-stage pipeline structure of entropy decoder, which supports decoding HD videos. It can decode syntax elements in sequence level, frame/field level, slice level, MB level and block level. With corresponding to entropy encoder, it can perform Fixed Length Decode (FLD), Exp-Golomb Decode and 2D Variable Length Decode (2D-VLD). The basic requirement for our architecture is to decode one coefficient in one clock. Special tables for bitstream shifter have been made to accelerate the process of extracting codewords from bitstream. The look-up table based method enables the entropy decoder to eliminate the dependency between extracting current codeword and decoding previous one. So our architecture supports pipeline operation and can function well at a high working frenquency.

The complete architectures have been described in Verilog HDL, simulated with Modelsim SE 6.3c simulator and implemented using FPGA of Xinlinx Vertex5 VLX330. They fully meet the demands of AVS HD encoder and decoder and support real-time encodeing and decoding for 1080P @ 30 frame/s or 1080i @ 60 field/s videos respectively.

***Keywords****:* Entropy Code, VLC, VLD, Architecture, AVS.