

一种用于并行 H.264 编码器的语法元素级分组并行算术编码器体系结构的评估

陈胜刚, 陈书明, 谷会涛, 刘 尧

(国防科学技术大学计算机学院, 湖南长沙 410073)

摘 要: 设计了一种语法元素指令流驱动的全流水 CABAC(Context-based Adaptive Binary Arithmetic Coding)熵编码 VLSI 结构, 并对提出的语法元素级分组并行算术编码器的体系结构进行了设计和开销评估. 该并行方法可以与现有符号级并行算法正交, 可同时使用, 适合大规模片上并行视频编码器; 相比标准 CABAC, 增加约 55% 的晶体管即可实现 2 倍以上的符号处理加速比和 > 1Gbin/s 的吞吐率.

关键词: 并行视频编码; 算术编码; VLSI 实现

中图分类号: TP332.2 **文献标识码:** A **文章编号:** 0372-2112 (2012) 02-0400-06

电子学报 URL: <http://www.ejournal.org.cn> **DOI:** 10.3969/j.issn.0372-2112.2012.02.031

A VLSI Architecture Evaluation of a Syntax Element Level Parallel Arithmetic Entropy Coder for Parallel H.264 Encoder

CHEN Sheng-gang, CHEN Shu-ming, GU Hui-tao, LIU Yao

(Computer School, National University of Defense Technology, Changsha, Hunan 410073, China)

Abstract: This paper proposes a new CABAC pipeline driving by syntax element instructions and a CABAC-based parallel arithmetic entropy coder for on-chip large-scale parallel H.264 video coders is also presented in this paper. Furthermore, the hardware architecture and transistor cost of the new parallel entropy coder are evaluated. This new parallel entropy coder can cooperate with the traditional symbol-level parallel algorithms and it suits the manycore platform well. Compared with the traditional CABAC hardware, the new parallel entropy coder can double its throughput to > 1Gbins/s by a cost of 55% transistors.

Key words: parallel video coder; arithmetic coder; VLSI implementation

1 引言

为完成 H.264^[1] 高清实时编码, 研究者通常采用各种并行的编码算法来加速^[2], 其中宏块级并行是一种并行度高且可扩展性好的方法. 据统计, H.264 并行高清编码器的宏块级并行度可达 4851 ~ 9169^[3]. 这对 H.264 的码流生成模块 CABAC 的吞吐率提出了较高的要求.

为提高 CABAC 的吞吐率, 研究者提出了很多用于 CABAC 专用电路的符号级并行方法. 与这些符号级并行方法正交, 我们在文献^[4]中提出了一种线程级并行算术编码算法, 对 CABAC 算法进行并行化, 形成基于语法元素分组的并行熵编码算法. 本文将通过对硬件加速器的体系结构设计, 对该并行算术编码器的硬件体系结构进行评估.

本文提出的硬件体系结构采用基于语法元素指令流驱动的全流水结构. 并行化方法从算法级提高

CABAC 的吞吐率, 硬件加速器利用电路实现高效的符号处理. 该并行算术编码器体系结构的优点是在算法级与现有的快速算法正交, 可同时在一个系统中集成; 同时硬件加速器接口规整, 指令流的驱动方式更利于与 Host 处理器实现软硬件协作. 模拟实现和 VLSI 实现结果发现, 相比标准 CABAC, 在较小的码率代价和约 55% 的晶体管开销下, 该并行算术编码器将能获得两倍左右的加速比、1Gbins/s 以上的二进制符号处理速度.

2 CABAC 相关研究

由于 CABAC 的比特级串行性和富含的分支及访存操作, 定制电路结构是实现 CABAC 的最有效方法^[5~8].

采用定制结构不仅可以充分的优化其逻辑结构和电路, 而且可以挖掘编码过程中存在的符号级并行. 符号级并行包含两类主要方法. 第一类方法利用等概率 (Equal Probability, EQ) 符号的特性, 在一个周期内处理

多个连续 EQ 符号和一个相邻的常规符号,实现单周期多符号的符号级并行处理^[5,6]。这种方法的优点是标准兼容,代价相对较小;缺点是 EQ 符号的出现频率不稳定,从而导致并行度不稳定。另外一类是把目前 CABAC 中的单符号概率模型扩展为多符号概率模型^[9],以实现在一个周期内处理多个符号。这种方法的优点是并行度稳定且较高,缺点是硬件逻辑开销和存储开销随每个周期处理的符号个数的增长以指数倍增加。

3 语法元素分组并行算术编码器体系结构

为完整,本文首先将对该算法进行一些简要的描述,然后对本文涉及的 CABAC 硬件流水线进行介绍,在此基础上对并行算术编码器体系结构进行分析和评估。

3.1 算法描述

H.264 中,每个宏块的预测变换编码的结果形成熵编码的输入——语法元素(Syntax Element, SE)。我们在文献[4]中提出,按照一定的规则把宏块的待编码语法元素划分为多个组,可实现基于语法元素分组的数据级多线程并行算术编码器。考虑到语义结构、开销和收益等因素,我们总共设置了三个线程,相应的语法元素分组方法如表 1 所示。根据对大多数测试序列的测试结果,表 1 的分类方法在线程负载、数据相关性等方面是最合理的。

表 1 语法元素集合划分方法

分组	简要描述
I	与宏块头信息相关的语法元素,包括宏块类型、预测信息和编码控制信息
II	与残差的位置信息相关的语法元素
III	与残差的值信息相关的语法元素

图 1 给出了在宏块级并行 H.264 编码器中,本文提出的并行熵编码器与并行的预测变换编码部分的联合执行模式。从中可以看到,预测变换编码部分并行度的提高会使得宏块重定序缓冲变得紧张;并行熵编码器可加速宏块重定序缓冲中待处理宏块的流出速度。

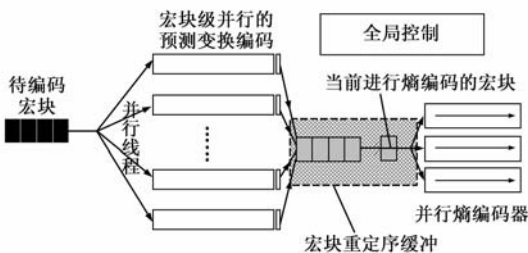


图 1 并行视频编码器执行模式

3.2 CABAC 流水线体系结构

本文采用语法指令流驱动的细分的流水线实现每个周期编码一个二进制符号。

3.2.1 流水线设置

CABAC 体系结构总共包含两条流水线(图 2):编码流水线,包括 9 级流水栈;初始化流水线,包括 7 级流水栈。

编码流水线的流水栈包括取语法元素指令、二进制化、上下文建模、微代码产生和二进制算术编码器。

二进制算术编码过程主要需要对编码区间 Range 和其下限值 Low 进行计算和更新。为了提高电路的工作频率,我们把 Range 和 Low 的计算和更新分别分配到两个流水栈中,缩短每栈流水线的逻辑长度。Marpe^[10]中提出了一种快速正规化的技术。本文利用该技术,把编码最小概率符号(LPS)时需要进行的正规化次数事先计算出来,与 CABAC 预设值一起存储在表格中。编码过程中读取预设值时可以同时得到移位次数,执行移位。编码最大概率符号(MPS)时最多需要执行一次正规化,仅需一次比较操作。

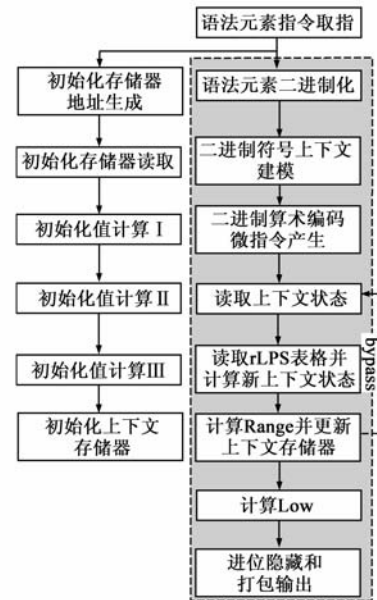


图 2 流水线设置

上下文状态初始化值计算过程中需要进行有符号乘法。本文把长有符号数拆分为两个短数据,利用 4 次短数据乘法和加法操作来代替长有符号数乘法,以缩短流水线的逻辑长度,并可以使上下文存储器初始化过程中的写存储器操作与编码流水线的更新上下文存储器操作一致。

上下文存储器采用双端口 SRAM 实现,在初始化过程中可以同时初始化两个存储器位置,节省部分初始化时间。另外,对于不同的条带类型和宏块类型,仅初始化所需部分。

3.2.2 软硬件协作

CABAC 中总共设置了二十多种语法元素用于表示预测变换编码的结果。熵编码对语法元素的处理过程

会用到多种不同类型、格式和取值的相邻信息以及编码历史信息.语法元素以及这些相关信息格式不统一,导致软硬件的功能划分及协作存在困难.

为解决此问题,本文设计了一种较为规则的语法元素指令,用于描述语法元素及其编码相关信息,并驱动流水线,如图3所示.

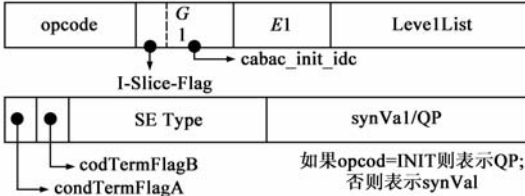


图3 语法元素指令格式

采用语法元素指令流驱动流水线的另外一个好处是利于软硬件的功能划分.熵编码的工作可以分为两类,一是语法元素指令的生成,二是语法元素的执行.Host 负责控制整个编码系统的运行,便于收集宏块相邻相关信息和历史信息,并生成语法元素指令.CABAC 硬件流水线专为执行语法元素指令而优化.Host 处理核与 CABAC 硬件流水线之间通过一个指令缓冲进行协作,如图4所示.

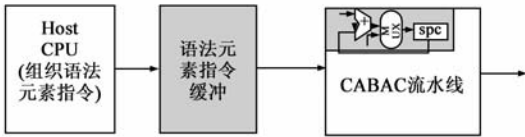


图4 Host与CABAC硬件加速器之间的协作关系

4 并行算术编码器体系结构

本文所提出的并行熵编码器的每一个语法元素的编码过程与其在标准 CABAC 中一致,因此并行方案的硬件结构实现较为简单.只需要在以上工作的基础上,进行根据语法元素的分组进行功能单元划分和复制即可,如图5所示.显然,并行算术编码器由三条子 CABAC 流水线组成,最后的码流合并和缓冲部分则可根据具体的应用场合增删.也正是这个特点,使得本文提出的并行算术编码器体系结构与已有的快速 CABAC 技术正交.这些快速技术可直接应用到并行算术编码器体

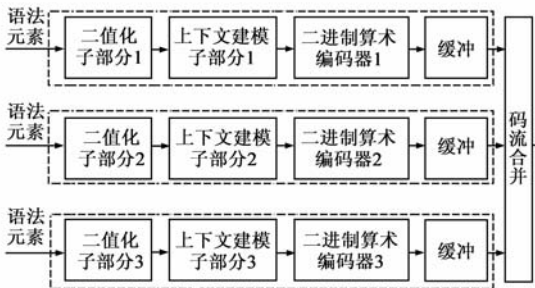


图5 并行算术编码器框架结构

系结构的每一条子 CABAC 流水线中,进一步提升流水线吞吐率.

表2对并行算术编码器的硬件开销进行了分析.从中可知,相比 CABAC 流水线结构,并行算术编码器结构的硬件开销主要来自两部分:二进制算术编码器和 FL1 二进制化逻辑、rLPS 表格以及下一状态存储器.占用面积比较大的存储器部分,特别是上下文存储器、初始化存储器,需要在三个线程间进行拆分,而不需要额外增加存储位置.

表2 并行算术编码器硬件结构代价分析

功能和结构	GROUPI	GROUPII	GROUPIII	说明
二进制化	查表, FLx, UEG3, U, TU	FL1	UEG0	FL1 结构出现冗余
上下文建模	根据语法元素类型进行选择	查表	公式在线计算	选择项之间不存在重叠
二进制算术编码器	需要完整结构	需要完整结构	需要完整结构	出现重叠
上下文存储器	入口集合1	入口集合2	入口集合3	存储位置不存在重叠
初始化存储器	入口集合1	入口集合2	入口集合3	存储位置不存在重叠
rLPS 表格	需要完全备份	需要完全备份	需要完全备份	rLPS 值以及正规化次数
nState 表格	需要完全备份	需要完全备份	需要完全备份	下一状态存储器

注: FLx: Fixed Length with cMax = x; U: Unary; TU: Truncated Unary; UEGk: Concatenated unary/k-th order Exp-Golomb

5 软件模拟与硬件实现结果及分析

本文对所提出的并行化方案进行了软件模拟,对其码率开销和加速比进行了简要分析;同时采用逻辑综合和物理实现的方式对其硬件开销进行了分析.

5.1 并行编码器效率分析

我们把提出的并行算术编码器集成到 JVT 提供的测试模型 JM8.6^[11]中.编码器主要设置选项为 IPPP 结构/简化的率失真优化/30fps/主要档次编码器.

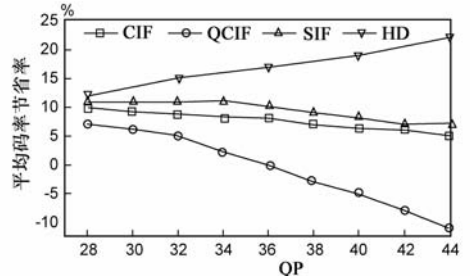


图6 相对于CAVLC的码率节省率

图 6 给出了本文对所提出的并行算术编码器与基于上下文自适应的变长编码(CAVLC)的编码效率的对比情况.对于图像尺寸较大的视频序列,特别是 HDTV 格式的大尺寸图像序列,并行编码器能充分保持算术编码相较于变长编码的优势.这说明本文提出的并行编码器从码率开销上来讲是有效的.

表 3 平均加速比

QP	平均负载加速比			
	QCIF	SIF	CIF	HD
28	2.54	2.24	2.44	2.68
30	2.52	2.35	2.45	
32	2.44	2.51	2.40	2.54
34	2.41	2.51	2.37	
36	2.32	2.41	2.30	2.30
38	2.24	2.42	2.23	
40	2.17	2.37	2.16	2.16
42	2.13	2.22	2.10	
44	2.11	2.20	2.07	2.09

表 3 给出了序列在不同 QP 设置下的平均负载加速比的统计结果,其中加黑的数字表示序列加速比的最大值.CABAC 的处理耗费周期数与待处理的二进制符号的个数往往成线性正比关系.因此,本文用二进制符号的数目来表征每个线程的负载.

从数据上看,并行方案加速比较小,但本文所提出的并行方案的最大优势是其可以与符号级并行和其他快速编码技术直接结合;协同成倍地提升熵编码吞吐率.

5.2 硬件实现结果及比较分析

5.2.1 CABAC 硬件实现结果及比较

我们采用自顶向下的标准单元设计方法对并行熵编码结构进行设计,整体设计采用简单的时序和面积约束.我们用 Verilog 硬件描述语言对第四节所提出的结构进行了逻辑实现,并采用 SMIC 0.13um LVT 工艺进行了逻辑综合.所有使用到的存储器均采用 ArtisanTM Memory Compiler 自动综合产生.表 4 给出本文和其他 CABAC 相关文献的实现结果,其中这 4 篇参考文献的工作代表了目前国际上最先进的几类研究.

文献[12]实现了基本的 CABAC 电路,采用动态流水线方案来改善 CABAC 编码器性能,并利用旁路机制来消除可能的存储器冲突.但是其动态流水线方案需要在编码过程中插入流水线气泡,因此电路在 150MHz 工作频率下仅能达到 80Mbins/s 的吞吐率.

文献[13]仅针对 CABAC 中的二进制算数编码器部分进行优化,而对于二进制化、上下文建模以及初始化等工作均不予关注.这样做的好处是可以集中精力关注关心的部分,但缺点是对系统的忽视容易造成整个系统性能的下降.同时文献[13]的研究要求 Host 处理

器负责为每一个二进制比特生成流水线驱动指令,这对主处理器造成过大的负担.

文献[6]要求二进制化能同时产生多个符号,从而导致逻辑开销增大.由于我们的目标是设计片上大规模并行视频编码器的 CABAC 加速器,面积将是一个重要的约束.另外,文献[6]仅关注 DCT 系数,而其他的则由主处理器负责处理;同时上下文存储器的初始化工作也由处理器来负责.这种类型的协作不彻底,将严重影响系统性能.

文献[14]提出了一种双模式的二进制编码器,把 CAVLC 和 CABAC 的逻辑适当地混合在一起以共享某些功能部件,达到缩减硬件开销的目的(表 4 给出了其中 CABAC 模式下的面积开销,存储器开销是 62.625 Kbits (ROM + RAM + FIFO).由于本文的应用场景硬件资源紧俏,尤其是存储资源占据绝大部分面积,所以这种复杂的功能融合机制是不合适的.在满足性能要求的前提下,面积开销是关键因素.同时,文献[14]的设计要求较大的数据供给带宽,这难以用在与 CPU 协作的场合.

表 4 CABAC 硬件实现结果对比

来源	工艺	吞吐率 (Mbins/s)	面积 (门数)	频率 (MHz)	存储器 (比特)
本文	SMIC0.13um	~ 590	3.21K	590	3.5K(RAM) 34.375K(ROM)
文献[12]	ROHM0.35um	80	4.57K	150	4K(RAM) ROM(未给出)
文献[13]	0.13um	/	4.42K	476	未给出
文献[6]	AMS0.35um	342 ~ 414	19.43K	180	未给出
文献[14]	TOWER0.18um	~ 308	43.88K	140	10.5K(RAM) 51.625K(ROM) 0.5K(FIFO)

本文设计采用 32 比特宽度的规则的语法元素流指令驱动,接口数据宽度要求小,数据流平缓.本文的设计包含了 CABAC 所有的功能模块,并通过细分流水栈的方法提高吞吐率.本文通过 CABAC 各个模块之间的协作,可以实现每个时钟周期处理一个二进制符号.本文针对上下文初始化内容的计算、上下文建模的方案、二进制算术编码器和码流输出等进行了与算法有关的优化,达到了提高系统吞吐率,缩小晶体管开销的目的.本文最后的实现结果表明,相对于其他相关研究的设计而言,本文的设计具有简单高效的特点.

5.2.2 CABAC 的 VLSI 实现

在硬件综合的基础上,本文采用 SOC Encounter 对 CABAC 流水线进行了 VLSI 实现,其版图如图 7 所示. CABAC 的面积总开销为 0.6mmx0.6mm,并满足时序要

求.从图中可以看出,ROM 和 RAM 存储器占据了大部分的面积开销,而逻辑部分的面积占用量较小.

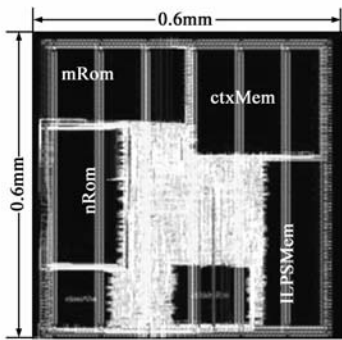


图7 硬件结构版图

5.2.3 并行算术编码器吞吐率及硬件开销评估

根据表 2 和表 3 的结果,本文提出的并行算术编码器的吞吐率将大于 1Gbins/s,这将能够充分的满足实时高清编码的要求;如果与其他的符号并行方案并用,所能够达到的吞吐率将能够满足更高分辨率的视频编码要求,将在很长一段时间内解决并行视频编码其中算术熵编码的瓶颈问题.

根据本文第 4 节的分析可知,本文提出的并行算术编码器相对于 CABAC 的硬件开销主要包括:二进制算术编码器和 FLI 的逻辑开销、rLPS 表格和 nState 存储器.

表 5 给出了在 VLSI 物理实现版图中各个部件所占的面积开销百分比.根据逻辑综合的结果可知,二进制算术编码器和 FLI 的逻辑开销占总面积的 8.08%,rLPS 表格和 nState 存储器分别占 13.99% 和 5.49%.因此,相比 CABAC 流水线,本文所提出的并行算术编码器的最终逻辑版图约需要花费 55% 额外的晶体管面积.

表 5 VLSI 实现结果的面积开销评估

功能单元	相对 CABAC 所占面积百分比
逻辑开销	26.95%, 其中 BAC + FLI 占 8.08%
rLPS 存储器	13.99%
nState 表格	5.49%
其他存储器	53.57%
并行熵编码器	155.13%

6 结论

本文根据微处理器技术发展方向,针对实时大规模并行视频编码所面临的熵编码瓶颈问题进行了分析,提出了一种以 CABAC 为基础的语法元素级分组并行熵编码器,并在 CABAC 硬件加速器的实现基础之上,对其硬件实现复杂度和代价进行了充分的讨论.该方案在软件的协同下,采用语法元素指令流进行驱动并产生最终码流.

实验结果表明,对于高清晰度的视频序列,在增加 55% 的晶体管面积和较小的码率代价下,本方案可以获得 2 倍以上的加速比和大于 1Gbins/s 的吞吐率;并能充分兼容符号级并行技术和其他 CABAC 快速算法,进一步提高符号处理速度,充分满足实时高清的编码需求.

参考文献

- [1] ITU-T Rec. H.264 (E) and ISO/IEC 14496-10, Advanced Video Coding for Generic Audiovisual Services[S].
- [2] 孙书为,陈书明.高效的 H.264 并行编码算法[J].电子学报,2009,37(2):357-361.
SUN Shuwei, CHEN Shuming. An efficient parallel algorithm for H1264/AVC encoder [J]. Acta Electronica Sinica, 2009, 37(2):357-361. (in Chinese)
- [3] Meenderinck C, et al. Parallel scalability of H.264[A]. First Workshop on Programmability Issues for Multi-Core Computers, Held in Conjunction with the 3rd International Conference on High-Performance Embedded Architectures and Compilers (HiPEAC)[C]. Goteborg, 2008.
- [4] Chen S, et al. P3-CABAC: A nonstandard tri-thread parallel evolution of CABAC in the manycore era[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2010, 20(6):920-924.
- [5] Sze V, et al. Parallel CABAC for low power video coding[A]. Gabbouj M. Proceeding of 15th IEEE International Conference on Image Processing[C]. San Diego: IEEE Press, 2008. 2096-2099.
- [6] Osorio R R, Bruguera J D. High-throughput architecture for H.264/AVC CABAC compression system[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2006, 16(11):1376-1384.
- [7] Jian-Wen C, et al. A high-throughput fully hardwired CABAC encoder for QFHD H.264/AVC main profile video[J]. IEEE Transactions on Consumer Electronics, 2010, 56(4):2529-2536.
- [8] Nunez-Yanez Y L, et al. Hardware assisted rate distortion optimization with embedded CABAC accelerator for the H.264 advanced video codec[J]. IEEE Transactions on Consumer Electronics, 2006, 52(2):590-597.
- [9] Lin J-H, Parhi K K. Parallelization of context-based adaptive binary arithmetic coders[J]. IEEE Transactions on Signal Processing (see also IEEE Transactions on Acoustics, Speech, and Signal Processing), 2006, 54(10):3702-3711.
- [10] Marpe D, et al. A fast renormalization technique for H.264/MPEG4-AVC arithmetic coding[A]. Seidel H-U. Proceeding of 51st International Scientific Colloquium(IWK)[C]. Ilmenau: Druckerei Kretzschmar, 2006.
- [11] Joint Model (JM)—H.264/AVC Reference Software[OL].

<http://www.iphome.hhi.de/suehring/tml/download/>

- [12] Li L, et al. A CABAC encoding core with dynamic pipeline for H.264/AVC main profile[A]. Wang Chua-Chin. Proceeding of IEEE Asia Pacific Conference on Circuits and Systems (APCCAS'06)[C]. Singapore: IEEE Press, 2006. 760 – 763.
- [13] Nunez-Yanez J L, Chouliaras V A. Design and implementation of a high-performance and silicon efficient arithmetic coding accelerator for the H.264 advanced video codec[A]. Stamatis Vassiliadis. Proceeding of 16th International Conference on Application-Specific Systems, Architecture and Processors (ASAP'05)[C]. Samos: IEEE Computer Society Press, 2005. 411 – 416.
- [14] Pastuszak G. A high-performance architecture of the double-mode binary coder for H.264/AVC[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2008, 18(7): 949 – 960.

作者简介



陈胜刚 男, 1981 年生于四川邻水. 国防科学技术大学计算机学院助理研究员, 博士, 主要研究方向为视频数字信号处理与体系结构.

E-mail: shgchen@nudt.edu.cn



陈书明 男, 1961 年生于安徽六安. 国防科技大学计算机学院教授, 博士生导师, 主要研究方向为高性能微处理器设计和超深亚微米 VLSI 设计理论与技术.

E-mail: smchen@nudt.edu.cn