doi:10.3772/j.issn.1002-0470.2022.05.003

# 分离真伪时钟的处理器 FPGA 原型性能校准方法<sup>①</sup>

郑雅文②\* \*\* \*\*\* 吴瑞阳\*\*\*\* 陈天奇\*\*\*\* 汪文祥\*\*\* \*\*\*\* 章隆兵\* \*\* \*\*\* 王 剑③\* \*\* \*\*\*

(\*计算机体系结构国家重点实验室(中国科学院计算技术研究所) 北京 100190)

(\*\* 中国科学院计算技术研究所 北京 100190)

(\*\*\*\* 中国科学院大学 北京 100049)

(\*\*\*\* 龙芯中科技术有限公司 北京 100190)

摘要 针对现场可编程门阵列(FPGA)原型系统中内存刷新频率过高导致内存延迟变大的问题,提出了一种校准处理器 FPGA 原型系统性能的方法,搭建了一个精确的 FPGA 原型性能验证平台,可用于硅前快速准确地评估处理器系统性能。问题的根本原因是 FPGA 原型系统同时存在真实墙上时钟和由运行频率降低导致的伪墙上时钟,且在内存 系统中刷新和访问请求分别按照两个时钟进行,然而真实机器上这两种请求都是按照真 墙上时钟进行,因此 FPGA 内存系统有性能误差。本文通过将两个墙上时钟分离来实现 校准,该校准方法准确度高、通用性强,校准后的 FPGA 原型系统运行 SPEC CPU 2006基 准测试程序性能分值平均误差由 7.49% 降至 0.36%,最高误差降至 2% 以下,可快速有 效地指导硅前性能优化。

关键词 硅前验证;现场可编程门阵列(FPGA)原型;内存系统;性能评估;校准方法

# 0 引言

近年来,国产高性能处理器自主研发成为国家 战略需要,并出现了龙芯、申威、飞腾、众志等自主处 理器。随着高性能处理器设计技术的发展,处理器 是否达到预期的性能指标成为设计者所关注的问 题。硅前性能验证是流片前查找性能问题、优化性 能瓶颈的重要手段,可以极大地减少设计流片的试 错成本,加快处理器设计迭代<sup>[1]</sup>。但随着国产处理 器设计复杂度和规模不断增大,硅前验证的难度和 时间成本也随之加大,占据整个周期的 70% 以 上<sup>[2]</sup>,成为设计周期中亟待优化的一环。

如何快速而准确地评估性能是处理器硅前验证 的主要问题之一。目前基于寄存器传输级(register transfer level, RTL)的主要验证平台有仿真器<sup>[3]</sup>、硬 件仿真加速器<sup>[4-5]</sup>和现场可编程门阵列(field programmable gate array, FPGA)原型系统<sup>[6-8]</sup>。随着实 际应用变得越来越复杂,用于性能基准测试程序集 也变大,如处理器性能基准测试 SPEC CPU 2017 相 比 CPU 2006 动态指令数多了 10 倍<sup>[9]</sup>,仿真器上进 行性能评估需要数年时间。基于 FPGA 的硬件仿真 加速器可加速 RTL 的全系统仿真速度。硬件仿真 加速器平台一般运行主频高达兆赫兹级别,且具有 较好的可观测性和调试性,如龙芯 3 号处理器在 Mentor Graphics 公司的 Veloce2 系列硬件仿真加速 器运行频率为 1 MHz,且可进行波形调试<sup>[10]</sup>。而 FPGA 原型验证平台运行频率能达到数十兆赫兹, 远高于硬件加速器,大幅加快了性能评估的速 度<sup>[11]</sup>。FPGA 原型系统平台通常采用大容量 FPGA 器件母板加丰富外设接口的子板构成,支持全系统

① 国家科技重大专项(2019ZX01029101-006)资助项目。

② 女,1993 年生,博士生;研究方向: 计算机系统结构,处理器设计;E-mail: zhengyawen@ict. ac. cn。

③ 通信作者, E-mail: jw@ict.ac.cn。 (收稿日期:2021-03-02)

功能验证和性能评估。

但 RTL 设计在 FPGA 原型系统上进行性能评 估与真实芯片相比存在误差。由于 FPGA 相比专用 集成电路 (application specific integrated circuit, ASIC)芯片,实现同样的逻辑需要更大的面积,且布 线结构更复杂,导致导线延迟变长;另外,由于 FP-GA 资源有限,大型 RTL 设计需要级联几个 FPGA 进行实现,互联线频率限制了 FPGA 运行主频。频 率降低使得在 FPGA 原型平台上对双倍速率(double data rate, DDR)内存系统的模拟与真实机器不 同。FPGA 原型平台上内存的周期性刷新间隔须与 真实机器保持一致,但由于 RTL 设计的频率较低, 导致内存读写访问频度比真实机器低,同主频下刷 新频率过高。刷新命令会延后读写命令的发送时 机,对内存系统延迟有影响。FPGA 原型系统的内 存刷新过多,增加了内存延迟,成为性能误差的主要 来源。

本文实现了一种通用的 FPGA 原型校准方法, 校准了内存系统延迟,搭建了精确的原型性能评估 系统,为处理器的迭代改进提供了可靠的硅前性能 数据。

1 相关工作

本节介绍了几种常见的处理器 FPGA 原型验证 平台,分析存在的性能误差问题,并比较基于 FPGA 原型的相关性能校准方法。

## 1.1 处理器 FPGA 原型验证平台

FPGA 原型平台常用于处理器 RTL 的全系统验 证。Intel 搭建了几款 x86 架构处理器的 FPGA 原型 验证平台。由于处理器的设计复杂度较高,在 FP-GA 上运行频率不高,且越复杂的设计频率越低。 Intel 基于 Virtex5 LX330 FPGA 搭建了 Atom 处理器 的全系统原型验证平台,处理器主频可达 50 MHz<sup>[12]</sup>。 基于 Xilinx Virtex4 LX200 FPGA 搭建了 Pentium 处 理器的全系统平台,处理器主频为 25 MHz<sup>[13-14]</sup>。 Intel Nehalem 乱序处理器由于设计规模变大,采用 了包括 Virtex4 和 Virtex5 的多片 FPGA 平台,可运 行在 520 kHz<sup>[15]</sup>。 随着处理器设计规模逐渐变大,为每款处理器项目定制一个原型平台费时费力,而通用性 FPGA 原型验证平台集成了一片或多片高容量高性能 FP-GA 芯片以满足各种片上系统(system on chip, SoC)的验证需求,运行频率可达兆赫兹级别,例如 S2C<sup>[6]</sup>、高性能专用集成电路原型系统(high-per-formance ASIC prototyping systems, HAPS)<sup>[7]</sup>、Proti-um<sup>[8]</sup>系列。为了支持全系统模拟,这些平台都提供了丰富的外部设备接口,例如 HAPS 上的内存系统可以通过内存子卡进行搭建。

无论是定制的还是通用型的 FPGA 原型平台, 最高频率只有兆赫兹级别,而处理器芯片实际运行 频率通常在吉赫兹级别,因此面临 FPGA 运行主频 比芯片低的问题。这使得 FPGA 原型系统上外部设 备频率与处理器系统频率的关系与芯片不能完全一 致,例如 FPGA 上内存频率与处理器核频率的比例 可能远高于真实机器的比例。并且,由于在 FPGA 上 RTL 设计的内存控制器的最高频率受限于设计 时序和 FPGA 优化,只能连接可工作在较低频率的 内存,而一般内存有最低工作频率限制。处理器与 内存频率差对功能验证的影响不大,但却给性能评 估带来了误差。

#### 1.2 相关性能校准方法

校准内存系统延迟的主要方法是构建内存系统 的时序模型。文献[16-18]构建了内存的时序模型 并与内存厂商美光公司提供的 RTL 参考模型进行 校准。但由于需要模拟内存控制器的时序,时序模 型要包含事务调度器、内存状态跟踪器、内存访问调 度器等单元,使得构建精确的时序模型变得复杂。 为了简化时序模型,文献[19]使用统计得到的固定 拍数作为内存延迟的模拟,但这种模拟方法不够准 确,且对于不同平台内存延迟不完全一样<sup>[20]</sup>,而且 这种方法只能通过在原本访存通路延迟基础上增加 拍数来模拟内存延迟,而 SoC 上 RTL 内存控制器由 于频率降低,反而需要减少拍数进行时序模拟。因 此,除非实现与内存控制器完全一致的时序模型,否 则无法对设计的内存控制器进行性能验证。

文献[21]通过在 RTL 内存控制器后连接高速的内存控制器知识产权核(intellectual property,

IP),避免了模拟内存控制器时序的需要,而且消除 了内存的最低工作频率的限制。但该文献存在两方 面问题:(1)不支持常见的内存读写命令背靠背传 输。该文献的实现不支持图1(a)所示的背靠背内 存读写传输,只支持每次读写事务串行完成的情况, 如图 1(b) 所示。而这种背靠背传输由于能增大访存带宽,在内存访问中很常见。(2) 对内存系统只进行了功能验证,未进行性能校准。本文改进了其实现,使得支持常见的内存访问传输,并且与真实机器进行了性能校准。



2 FPGA 原型的性能误差分析与校准 方法

#### 2.1 基于 FPGA 原型的处理器性能评估方法

## 2.1.1 处理器 FPGA 原型平台

图 2 为基于 FPGA 原型平台搭建的通用处理器 原型系统结构,该原型系统由 FPGA 里的处理器、 FPGA 上的 DDR4 内存子卡和外接的桥片板卡组 成。FPGA 里的处理器包括处理器核、末级缓存、内 存控制器、互联总线以及总线控制器等模块;FPGA 上的内存子卡连接内存集成到系统中;板卡由 HyperTransport 总线与 FPGA 上的处理器相连,提供其 他外设接口,用于连接硬盘、网络等设备。





## 2.1.2 性能评估方法

由于 RTL 设计在 FPGA 上无法达到真实芯片 的运行频率,因此 FPGA 原型系统上需要整体降低 RTL 设计的运行频率,并通过主频换算得到性能评 估数据。与真实机器类似,FPGA 上各部分处于不 同的时钟域,大致可分为处理器核、内存、总线几个 时钟域。假设真实芯片中处理器核频率为1.2 GHz, 内存频率为 800 MHz,但若 FPGA 上处理器核频率 降至 24 MHz,而内存频率仍为 800 MHz,将导致内存 访问延迟周期数比真实机器中的要少很多。为了在 FPGA 原型系统上准确模拟真实情况,通常将 FPGA 上的各部分频率比例关系与真实情况保持一致,整 体将运行频率按比例调低,调低的倍数为频率缩放 倍数。

$$Perf_{sim} = \frac{1}{time_{sim}} = \frac{freq_{chip}}{freq_{FPGA}} \times \frac{1}{time_{FPGA}}$$
(1)

如式(1)所示,由于 RTL 与芯片是同一套设计,所以 同一时钟域内的拍数延迟是一致的,因此降频后 FPGA 上测得的性能分值可通过等比例缩放得到其 模拟的真实机器的性能数据。例如将 FPGA 上的核 频率调至 24 MHz,内存频率为 16 MHz,同时缩小 50 倍,则测试在 FPGA 上得到的性能分值通过乘以频 率缩放的 50 倍得到模拟的真实机器的性能分值。

但上述计算方法只有在理想情况下才能得到准

确的性能数据。在实际系统中,虽然 FPGA 上频率 可以整体调低,但某些系统事件频率没有因此降低, 且这些事件对性能有影响。例如时钟中断频率由内 核的配置决定,同一内核下 FPGA 与真机的时间片 长度相等,但由于 FPGA 上处理器频率低,同一长度 的时间片中用于处理时钟中断的时间更长,使得用 户程序的时间片占用率很低,整体测试性能变差。 为了在 FPGA 和真机上得到同样的用户程序时间片 占用率,本文通过降低 FPGA 上内核中的时钟中断 频率,使得 FPGA 上时间片变长,最终使得 FPGA 和 真机用于时钟中断处理的时间占比一致。对于计算 密集型性能测试,降低时钟中断频率对系统调度产 生的影响可以忽略不计。但并非所有事件都可以直 接降低其发生频率。典型的如内存刷新频率,其由 内存属性而定,无法随系统频率降低,且刷新频率对 内存延迟有影响,使得内存敏感的测试程序仍存在 较大的性能误差,对内存系统的校准成为整体性能 校准的主要工作。

## 2.2 FPGA 内存系统误差分析

内存系统<sup>[22]</sup>通常由内存控制器、端口物理层 (physical, PHY)和内存组成。内存用电容存储数 据,由于电容存在漏电,需要不断对存储电容进行周 期性充电操作才能保存正确的数据,这种周期性数 据读写即内存的刷新操作。由于内存刷新速率与工 作频率无关,与内存本身属性有关,因此内存控制器 发送刷新请求的频率应根据内存属性要求的刷新频 率,否则可能导致数据出错。因此,虽然 FPGA 上的 内存系统运行频率较低,但由于与真实机器使用同 样的内存,刷新频率与真实机器一致,经过主频换 算,会出现刷新频率比同主频下真实机器更高的情 况。由于内存控制器在发送刷新请求时会阻塞读写 访存请求,并且刷新请求使得后续访存请求需要重 新激活,因此刷新频率变高会增加访存延迟,导致性 能下降。

图 3 反映了刷新请求对访存请求的影响。当写 请求被刷新命令阻塞时,当次写到下一次写需要经 过当次写完成、预充待刷新行、刷新该行、重新激活 下一次写和发送下一次写等阶段。其中,WL 指写 延迟,BL 指突发传输长度,tWR 指写恢复时间,用于 计算当次写完成时间;tRP 指行预充电有效周期,用 于计算预充完成时间,tRFC 指刷新周期,用于计算 刷新完成时间;tRCD 指行有效到列有效时间,用于 计算重新激活的时间。而未被阻塞情况下可直接在 当次写发送后的 tCCD 延迟之后发送下一次写。对 比正常写请求,以 DDR4 (假设速度为 2133 bps,即 tck = 0.938 ns)为例,两次刷新间周期 tREFI 是 7.8 µs, WL = 14tck, BL = 8tck, tWR = 15 ns, tRP = 15 ns, tRFC= 350 ns, tRCD = 15 ns, tCCD = 4tck, 一次刷新阻塞 访存请求带来的性能损失大约为

$$\frac{tck \times \left(WL + \frac{BL}{2}\right) + tWR + tRP + tRFC + tRCD - tCCD \times tck}{tREFI}$$



图 3 刷新阻塞访存请求示意图

由于难以准确统计被刷新阻塞的访存指令的次数,且由于访问延迟变长,各种队列的阻塞加剧,产 生其他级联反应,因此在 FPGA 原型系统上无法准 确估算访存延迟。

## 2.3 分离真伪墙上时钟的内存系统校准方法

2.3.1 内存系统校准方法

如图 4 所示, FPGA 内存系统延迟不准是由于 系统中存在两个墙上时钟,即真墙上时钟和伪墙上 时钟。这里对两个墙上时钟进行定义:真墙上时钟 是真实时间下的时钟,与真实机器时钟一致,也是内 存刷新频率依据的时钟;伪墙上时钟是由于 FPGA 运行主频降低产生的墙上时钟,比真墙上时钟慢的 倍数为频率降低的倍数,为处理器、内存控制器等运 行的时钟。由于运行于伪墙上时钟的内存控制器需 要按真墙上时钟的频率向内存发送刷新,以至刷新 命令相比访存命令频率过高,访存命令更容易被阻 塞导致整体延迟变大。而真实机器上由于刷新命令 和访存命令都是按真墙上时钟发送,因此与 FPGA 的情况不同。这使得 FPGA 上内存延迟与真实机器 相比有误差。



本文的校准方法是将内存控制器发送的访存命 令和刷新命令都调至伪墙上时钟频率,使得访存命 令被刷新阻塞的情况与真实机器一致。但内存所需 的刷新频率是按真墙上时钟计算的,将刷新频率调 至伪墙上时钟会降低内存的刷新频率。由于内存不 及时刷新会导致保存的数据出错,因此需要增加一 个运行在真墙上时钟下的内存控制器 IP 用于向内 存发送刷新请求。这使得内存系统有两个内存控制 器级联,RTL 实现的内存控制器连接处理器核,处于 伪墙上时钟域;新增的内存控制器 IP 处于真墙上时 钟域,并连接内存。两个内存控制器之间通过转接 桥将原有内存控制器的请求转发到新增的内存控制 器,并由转接桥进行跨时钟域转换。

## 2.3.2 内存系统校准实现

(1) 整体结构

如图 5 所示,本文实现的 FPGA 原型内存系统 包括 RTL 设计的内存控制器、双速率动态存储物理 层接口(DDR PHY interface, DFI)<sup>[23]</sup>到高级扩展接 口(advanced extensible interface, AXI)<sup>[24]</sup>的转接 — 466 — 桥、内存控制器 IP、DDR 内存。相比于原本结构,改 进后的内存系统将原本的 PHY 换成了新设计的伪 PHY(pseudo PHY),其中核心部件为 DFI-AXI 转接 桥,用于连接 RTL 设计的内存控制器和高速内存控 制器 IP。其中,RTL 设计的内存控制器运行在伪墙 上时钟域,而高速内存控制器 IP 和内存运行在真墙 上时钟域,二者之间通过 DFI-AXI 转接桥内的异步 先入先出队列(first in first out,FIFO)进行跨时钟域 信号转换。



通过上述设计,RTL 设计的内存控制器产生的 内存刷新请求也等比例地调整为伪墙上时钟所对应 的频率关系,因而各访存事件与其产生的性能影响 与真实机器保持比例一致。另一方面,由于内存真 正的刷新请求是由工作在真墙上时钟的高速内存控 制器 IP 发出的,与真实机器的内存刷新周期相同, 因此功能正确性不受影响。

(2) DFI-AXI 转接桥的实现

DFI-AXI 转接桥将内存控制器发送的 DFI 接口标准<sup>[23]</sup>读/写请求转换成 AXI<sup>[24]</sup>接口标准的请求, 并转发给内存控制器 IP。DFI 协议与 AXI 协议有以 下 3 个方面的主要区别。

1)地址表示方法不同。DFI协议的地址是内存 寻址地址,包括通道选择地址、bank组地址、bank地 址、行地址、列地址,而AXI协议传输的地址是物理 地址。因此,DFI接口上的地址信号不能直接用于 AXI接口的地址信号发送。

2)接口信号不同。DFI协议接口信号根据功能 可分为命令、地址等控制信号、读数据信号、写数据 信号;AXI协议接口信号可分为读地址、读数据、写 地址、写数据、写响应通道信号。

3)时序控制不同。AXI 协议各类信号相对独 立,没有时序的要求;DFI 协议的控制信号与数据信 号之间有时序要求,由内存参数控制。

DFI-AXI 转接桥将 DFI 接口上的信号根据协议 进行译码;对译码得到的读写访存请求,根据 AXI 协议转换成地址、数据位宽、数据等信号;将待发送 到 AXI 接口上的信号通过异步 FIFO 转换到内存频 率并发送。

转接桥可分为命令译码单元、地址合成单元、 读/写地址及数据状态机、读/写地址及数据异步 FIFO等模块。其中,命令译码单元负责识别 DFI 接 口上的读写访存相关命令,过滤不需要继续传递的 命令。地址合成单元将内存寻址地址重构成物理地 址。根据 DDR 协议,一个完整的访问需要先激活某 一行,再对该行的某一列进行读写。由于同时可能 有很多行被激活,因此需要将所有被激活的行存下 来,只要遇到相应的 bank 列读写命令,则将已激活 的该 bank 行地址取出,与访存命令的列地址重新拼 合成物理地址。命令译码单元产生的控制信号控制 读写状态机将地址、数据等信息通过异步 FIFO 传 递到 AXI 接口。异步 FIFO 将低频的伪墙上时钟域 信号转成高频的真墙上时钟域信号,使得输出的 AXI 接口信号与内存控制器 IP 频率匹配。

为了支持读写背靠背请求的传递,本文采用了 将地址和数据状态机分离的设计方法,对命令地址 和数据通道分别进行处理,使得可传输请求带宽更 高。与文献[21]以一次传输事务为单位的状态机 实现方法不同,分离后地址和数据状态机可以分别 接收不同传输事务的地址和数据,无需上一次的数 据传输完成后再进行下一次的地址传输,地址、数据 可以按序接续发送,实现了读写请求背靠背传输的 支持。

(3) 内存延迟的校准

内存延迟的校准主要根据 DFI 协议对读请求的 数据返回延迟进行校准。根据 DFI 协议,写请求没 有握手,读请求需要在指定内存参数拍数内返回读 数据,所以内存控制器到内存的延迟只受读数据返 回延迟的影响。因此,FPGA 和真机采用同一套内 存配置参数,只要在 FPGA 上满足在同一内存参数 指定拍数内返回读数据,就能达到内存控制器访问 内存的延迟拍数与真实机器一致的校准目标。

在内存参数指定的拍数内,内存控制器 IP 需要 完成访问内存并返回读数据。如果内存控制器 IP 需要刷新内存,将会延长返回延迟。为了保证在指 定拍数内控制器 IP 一定能返回读数据,将 RTL 设 计的内存控制器的运行频率适当调低,内存频率调 高,使得内存参数要求的返回延迟变长,进而保证即 使在内存刷新的情况下也能在该拍数内返回读数 据。而一般 FPGA 上内存控制器运行频率较低,所 以不需要额外降频也能满足返回拍数要求。

3 实验与结果分析

本文基于龙芯3号处理器搭建的 FPGA 原型平 台进行实验。通过实验对比了内存校准前后的 FP-GA 原型平台上 RTL 设计的性能分值,并与该设计 流片后的真实机器的性能分值进行对比。

本文使用 Synopsys HAPS-80 S52 作为 FPGA 原 型平台。HAPS-80 S52 原型验证平台提供了 2 个 Xilinx Virtex UltraScale XCVU 440 FLGA 2892 FPGA 芯片和丰富的接口用于扩展。使用 HAPS-80 S52 搭 建了原型系统,连接了 DDR4\_HT3 作为内存子卡, 外接自定制板卡提供南北桥和 L/O 资源。

实验原型系统基于龙芯 3 号处理器搭建,由 1 个乱序 4 发射 64 位 GS464 系列处理器核、1 级 AXI 交叉开关总线、片上共享高速缓存、2 级 AXI 交叉开 关总线、内存控制器以及 IO 控制器等模块构成的 FPGA 原型,并由 HyperTransport 总线连接桥片提供 其他外设接口用于连接串口、硬盘、网络等设备。

为了尽量减少性能测试程序的运行环境差距, 在 FPGA 原型系统与真实机器上均采用 Linux 4.19.73内核,测试程序使用同一二进制,均测试单 核单线程性能。并且,FPGA 与真机使用同一套 RTL设计和配置。如表1所示,真实机器与 FPGA 系统上 RTL处理器主频和内存控制器频率的比例 关系一致,FPGA 系统上的性能分值经过频率换算 后与真实机器进行比较。另外,真实机器和 FPGA 原型平台的各级 cache 和内存大小一样,两者使用同样的内存参数。

	平台	真实机器	FPGA 原型
频率	处理器	1200	24
/MHz	RTL 内存控制器	400	8
容量	L1 cache	64 kB	64 kB
	L2 cache	256 kB	256 kB
	L3 cache	2 MB	2 MB
	内存	8 GB	8 GB

表1 真机与 FPGA 原型平台的配置

本文使用 lmbench 3. 19 中的访存延迟测试 lat \_ mem \_ rd<sup>[25]</sup>、访存带宽测试 stream<sup>[26]</sup>程序、处理器 性能基准测试集 SPEC CPU 2006<sup>[27]</sup>作为性能测试 程序集,SPEC CPU 2006 测试使用 train 输入集进行 测试。

图 6 显示了对各级访存层次的访问延迟的测试,在越过每级访存层次的容量时都会有访问延迟的陡增,分别显示了各级 cache 和内存的访问延迟。对比各级访问延迟发现,对于各级 cache,FPGA 校准前后误差均不大。但 FPGA 校准前对内存访问延迟的模拟误差较大,为18.26%,校准后基本接近真实机器的访问延迟,误差仅为-0.72%。这表明经过校准,内存访问延迟基本准确。



图 6 FPGA 校准前后 lat \_ mem \_ rd 测试结果比较

如图 7 所示, FPGA 校准前由于内存延迟偏大, 因此测得内存带宽比真实机器低 20% 以上;经校准 后 FPGA 原型系统运行 stream 测试程序与真机相比 误差均小于2%。测试本身存在一定的波动范围误 差。实验结果反映了本文提出的内存校准方法是有 效的。



如图 8 所示, FPGA 校准前 SPEC CPU 2006 测试 部分误差较大,误差最大的测试 459. GemsFDTD 高 达 40%,其余测试如 433. milc、437. leslie3d、470. lbm 误差在 20% 以上, 429. mcf、410. bwaves、450. soplex 误差在 10% 以上, 平均误差为 7. 49%。经校准后 FPGA 上运行 SPEC2006 测试误差最高仅为 2%,大 部分误差小于 1%, 平均误差为 0. 36%。由于测试 本身有一定的波动范围误差,因此校准后 FPGA 原 型平台可以获得较准确的 SPEC CPU 2006 性能分 值。

实验结果表明,经过校准后的 FPGA 原型系统 对于计算密集型和内存敏感型的测试都能获得较为 准确的性能数据。

# 4 结论

本文基于分离真伪墙上时钟提出了一种通用的 FPGA 原型系统性能校准方法,解决了 FPGA 原型系 统上内存延迟不准和用户程序时间片占用率过低的 问题,使得在 FPGA 原型设计平台上性能评估时误 差在 2% 以下。本文提出的方法理论上适用于任何 真墙上时钟和伪墙上时钟并存的系统校准,具有一 般性意义。并且,本文提出校准方法的实现与设计 耦合度很低,因此对于不同设计及平台不需要重新



图 8 FPGA 校准前后 SPEC CPU2006 测试结果比较

进行性能校准,具有通用性。校准后的 FPGA 原型 验证平台为硅前性能分析提供了更快速准确的解决 方案,相对于硬件仿真加速器达到数十倍的速度提 升,成本数十倍降低,使得性能评估可以运行更大的 测试程序,获得更完备且可信的评估结果。

虽然本文实现了在 FPGA 原型平台上对计算、 访存密集型测试的准确评估,但降低时钟中断频率 实际上会对 IO 访问延迟带来影响。后续工作将进 一步评估降低时钟中断频率对 IO 访问密集型测试 性能产生的影响,进一步完善 FPGA 原型性能验证 平台。

#### 参考文献

- [ 1] EECKHOUT L. Computer Architecture Performance Evaluation Methods [ M ]. Williston: Morgan and Claypool Publishers, 2010
- [2] DEMPSTER D, STUART M. Verification Methodology Manual Techniques for Verifying HDL Designs [M]. Winchester: Teamwork International, 2002: 1-5
- [ 3] Synopsys. VCS: industry's highest performance simulation solution[EB/OL]. https://www.synopsys.com/verification/simulation/vcs.html: Synopsys,[2021-01-02]
- [ 4] Mentor Graphics. Veloce emulation platform [EB/OL]. https://www.mentor.com/products/fv/emulation-systems/: SIEMENS, [2021-01-02]
- [5] Synopsys. ZeBu server ASIC emulator [EB/OL]. https: // www. synopsys. com/verification/emulation/zebu-server. html: Synopsys, [2021-01-02]
- [ 6] S2C Inc. Design SoC Using FPGA-Based IP[M]. San Jose: S2C Inc, 2005: 3-10
- [7] Synopsys. HAPS-52 Virtex-5 Motherboard Datasheet [M].
  Mountain View: Synopsys Inc, 2010: 1-3

- [8] Cadence Design Systems. Protium S1 desktop prototyping platform [EB/OL]. https://www.cadence.com/en\_ US/home/tools/system-design-and-verification/fpgabasedprototyping/protium-s1-fpga-based-prototyping-platform.html;Cadence Design Systems, [2021-01-02]
- [9] AKRAM A, SAWALHA L. A survey of computer architecture simulation techniques and tools[J]. *IEEE Access*, 2019, 7: 78120-78145
- [10] 王朋宇,吕旭东. Mentor Veloce2 硬件加速器在 LOONG-SON 3 A4000 片上多核处理器设计验证中的应用[EB/ OL]. https://www.mentor.com/products/fv/resources/ overview/mentor-veloce2-loongson-3a4000--36ad9178f694-46f1-92c4-3a7b0e21b569: Mentor Graphics,[2021-01-02]
- [11] FAROOQ U, MEHREZ H. Pre-silicon verification using multi-FPGA platforms: a review[J]. Journal of Electronic Testing, 2021, 37: 1-18
- [12] WANG P H, COLLINS J D, WEAVER C T, et al. Intel atom processor core made FPGA-synthesizable[C] // Proceedings of the 17th International Symposium on Field-Programmable Gate Arrays, New York, USA, 2009: 209-218
- [13] LU S L L, YIANNACOURAS P, KASSA R, et al. An FPGA-based Pentium R in a complete desktop system[C] // Proceedings of the 15th International Symposium on Field-Programmable Gate Arrays, New York, USA, 2007: 53-59
- [14] LU S L L, YIANNACOURAS P, SUH T, et al. A desktop computer with a reconfigurable Pentium R[J]. ACM Transactions on Reconfigurable Technology and Systems, 2008, 1(1): 1-15
- [15] SCHELLE G, COLLINS J, SCHUCHMAN E, et al. Intel Nehalem processor core made FPGA synthesizable[C] // Proceedings of the 18th International Symposium on Field-Programmable Gate Arrays, New York, USA, 2010: 3-12
- [16] KARANDIKAR S, MAO H, KIM D, et al. FireSim: FP-

GA-accelerated cycle-exact scale-out system simulation in the public cloud [C]//2018 ACM/IEEE 45th Annual International Symposium on Computer Architecture (ISCA), Los Angeles, USA, 2018: 29-42

- BIANCOLIN D, KARANDIKAR S, KIM D, et al. FASED: FPGA-accelerated simulation and evaluation of DRAM
   [C] // In Proceedings of the 2019 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays, New York, USA, 2019: 330-339
- [18] STEINER L, JUNG M, PRADO F S, et al. DRAMSys4.0: a fast and cycle-accurate system C/TLM-based DRAM simulator [C] // International Conference on Embedded Computer Systems, Samos, Greece, 2020; 110-126
- [19] 朱英,陈诚,许晓红,等. 一款多核处理器 FPGA 验证
  平台的设计与实现[J]. 计算机研究与发展, 2014, 51
  (6): 1295-1303
- [20] VERDEJO R S, ASIFUZZAMAN K, RADULOVIC M, et al. Main memory latency simulation: the missing link [C] // Proceedings of the International Symposium on Memory Systems, Washington, USA, 2018: 107-116
- [21] MAYEKAR P A. Design and Verification of a DFI-AXI

DDR4 Memory PHY Bridge Suitable for FPGA Based RTL Emulation and Prototyping[D]. New York: Rochester Institute of Technology, 2019: 0-122

- [22] JACOB B, NG S, WANG D. Memory Systems: Cache, DRAM, Disk [M]. San Francisco: Morgan Kaufmann Publishers, 2007
- [23] Cadence Design Systems, Inc. DDR PHY Interface (DFI) Specification 5.0[S]. Austin, USA: 2018
- [24] ARM Limited. IHI0022H AMBA AXI and ACE: Protocol Specification [S]. Cambridge, England: ARM, 2020
- [25] McVOY L, STAELIN C. Lmbench: portable tools for performance analysis[C] // In Proceedings of the 1996 Annual Conference on USENIX Annual Technical Conference, Berkeley, USA, 1996: 23-23
- [26] McCALPIN J D. Memory bandwidth and machine balance in current high performance computers [J]. IEEE Computer Society Technical Committee on Computer Architecture Newsletter, 1995(2): 19-25
- [27] Standard Performance Evaluation Corporation. SPEC CPU<sup>®</sup> 2006[EB/OL]. https://www.spec.org/cpu2006/: Standard Performance Evaluation Corporation, [2021-01-02]

# A performance calibration method of processor FPGA prototyping by separating real and pseudo wall clock

ZHENG Yawen<sup>\*</sup> \*\*\* \*\*\*\*</sup>, WU Ruiyang<sup>\*\*\*\*</sup>, CHEN Tianqi<sup>\*\*\*\*</sup>, WANG Wenxiang<sup>\*\*\*\*</sup>, ZHANG Longbing<sup>\*</sup> \*\*\*\*\*\*, WANG Jian<sup>\*</sup> \*\*\*\*\*\*</sup>

(\*State Key Laboratory of Computer Architecture, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190)

(\*\* Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190)

(\*\*\* University of Chinese Academy of Sciences, Beijing 100049)

(\*\*\*\*\* Loongson Technology Corporation Limited, Beijing 100190)

#### Abstract

Aiming at the problem that memory refresh rate is too high in field programmable gate array (FPGA) prototype system, which leads to large memory latency, a method to calibrate the performance of FPGA prototype system is proposed in this paper, and an accurate FPGA prototype performance verification platform is built, which can be used to quickly and accurately evaluate processor system performance before silicon. The root cause of the problem is that the FPGA prototype system has both a real wall clock and a pseudo wall clock caused by the decrease of running frequency, and the refresh and access requests in the memory system are carried out according to two clocks respectively. However, the two requests in the real machine are carried out according to the real wall clock, so the FPGA memory system has performance errors. In this paper, two wall clocks are separated to achieve the calibration. The calibrated FPGA prototype system running SPEC CPU2006 benchmark program is reduced from 7. 49% to 0.36%, and the maximum error is reduced to less than 2%, which can quickly and effectively guide the pre-silicon performance optimization.

Key words: pre-silicon verification, field programmable gate array (FPGA) prototyping, memory system, performance evaluation, calibration method