

可信执行环境的 RISC-V 架构处理器安全分区方法

王健^{1,2}, 付志博^{1,2}, 明哲^{1,2}

(1. 南方电网数字电网集团信息通信科技有限公司, 广州 510670; 2. 广东省数字电网技术企业重点实验室)

摘要: RISC-V 架构处理器分区过程中, 因未处于可信执行环境中, 使得部分分区内容出现了新行内容随意替换旧行内容的问题, 导致分区结果重叠部分较多。为此, 提出基于可信执行环境的 RISC-V 架构处理器安全分区方法。根据请求指令中的虚拟机 ID 编号将虚拟机与虚拟根实例进行绑定, 为 RISC-V 架构处理器中的划分提供安全环境。将安全内存加密引擎集成到内部的 Cache 与存储控制器间, 生成单次分块区。利用多用户端密钥对称算法获取访问密文, 采用蒙特卡罗法计算当前分区解向量对应的偏移时间。根据该偏移时间序列, 采用 Cache 方法进行分区可信通道物理页着色处理, 可避免出现新行随意替换旧行的问题, 获取安全的分区结果。由实验结果可知, 使用本研究方法分区命中率最小值可达到 0.82, 能够达到理想的分区效果。

关键词: 可信执行环境; RISC-V 架构; 安全分区

中图分类号: TP301

文献标识码: A

RISC-V Architecture Processor Security Partitioning Method Based on Trusted Execution Environment

Wang Jian^{1,2}, Fu Zhibo^{1,2}, Ming Zhe^{1,2}

(1. China Southern Grid Digital Grid Group Information and Communication Technology Co., LTD., Guangzhou 510670, China;

2. Guangdong Key Laboratory of Digital Power Grid Technology Enterprises)

Abstract: During the partitioning process of the RISC-V architecture processor, it is not in a trusted security environment, which causes the problem of the new row content replacing the old row content at will in some partition content, resulting in more overlapping parts of partition results. Therefore, a secure partitioning method for RISC-V architecture processors based on trusted execution environment is proposed. According to the virtual machine ID number in the request instruction, the virtual machine is bound to the virtual root instance to provide a secure environment for the security division in the RISC-V architecture processor. Integrate the secure memory encryption engine between the internal cache and the storage controller to generate single-shot chunks. The multi-user key symmetry algorithm is used to obtain the access ciphertext, and the Monte Carlo method is used to calculate the offset time corresponding to the current partition solution vector. According to the offset time series, the Cache method is used to color the physical pages of the partitioned trusted channel to avoid the problem of replacing old rows with new rows and obtain safe partitioning results. It can be seen from the experimental results that the minimum value of partition hit rate can reach 0.82 using the research method, which can achieve the ideal partition effect.

Keywords: trusted execution environment; RISC-V architecture; security partition

0 引言

目前, 符合 RISC-V 规范指令集模块式电力学系统在调度、配置、可靠性和安全性等方面的问题越来越受到人们的重视。因此, 有必要针对 RISC-V 架构处理器安全分区进行研究。参考文献[1]提出了最优虚拟截止日期分区法, 该方法结合一种新的基于时间的可调度性测试方法以实现对关键值的有效切换。基于系统级截止时间缩短参数设计最佳虚拟截止日期分配方式。然而, 由于这种方式对任务的请求数量进行了限制, 因此, 如果任务没有

足够的物理内存, 则由于分区日期的限制, 不能对整个区域进行分区; 参考文献[2]提出了基于多核处理器的调度方法, 该方法利用遗传算法对每个任务到目标节点之间的最长路径进行排序, 并根据结果对任务进行排序, 在分配处理器内核时, 确定任务最早可执行时间。通过设置最佳匹配评价函数实现处理器安全分区, 但是这种方法牵扯到大量的数据运算, 如果根据计算结果频繁更改分区, 则会造成很大的开销, 导致分区效率较低。

上述这两种方法出现问题主要是因为执行环境不可靠, 为此提出基于可信执行环境的 RISC-V 架构处理器安

全分区方法。该方法在可信指令执行环境下建立可信执行环境的分区可信通道,结合蒙特卡罗法进行安全分区。

1 基于 TPCM 虚拟化架构的分区可信指令执行

基于 TPCM 虚拟化架构的分区可信指令执行框图如图 1 所示。

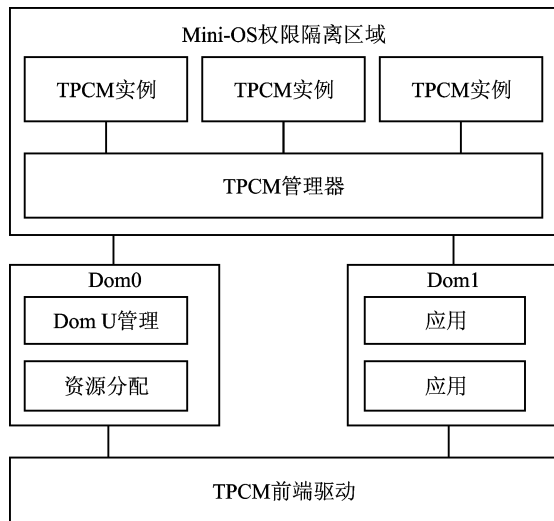


图 1 基于 TPCM 虚拟化架构的分区可信指令执行框图

根据图 1,在 Mini-OS 权限隔离区域,TPCM 管理程序完成了对系统的“暂停”、“删除”和“迁移”等操作。在 Dom0 创建新的客户机时,需要 VirtualRoot 创建一个可信 Root 实例,该实例与客户机相关^[3-5]。该方法将虚拟机和虚拟根实例结合,在虚拟机运行期间,如果需要使用虚拟的可信根实例,就会对 RISC-V 体系结构的前端驱动发送一个可信根应用请求^[6]。基于此,提出一种基于 Mini-OS 的虚拟信任根系统框架,将其与其他领域分离,增强安全性。以 RISC-V 架构处理器为基础,采用虚拟技术将 RISC-V 架构处理器中的信任源代码引入云环境中,为 RISC-V 架构处理器中的安全划分提供了保证^[7]。

2 基于可信执行环境的处理器安全分区

2.1 可信执行环境的分区可信通道建立

在分区可信指令执行环境下,在 RISC-V 架构处理器上建立空实例模型,锁定源端的虚拟根实例状态,并将其状态信息传递给空实例,在 RISC-V 体系结构的处理器中,从源平台中移除一个实例,实现虚拟可信根的迁移^[8-9]。基于此,可信执行环境的分区可信通道建立步骤如下:

① 双向身份认证。双向身份认证采用了“握手”协议,即在验证服务端发起请求,用户端进行响应。

用户根据客户端的要求对认证服务器进行访问,并将认证证书发送给服务器。将该部分信息与数据库服务器

中的信息进行对比分析,判断注册用户是否合法,若合法,则证明双向身份认证成功,否则失败^[10]。

② 协商会话密钥。在双方身份认证之后,两方计算会话密钥如下:

$$\begin{cases} k_{ab} = k_{ba} \\ k_{ab} = g^{r_a r_b} \\ k_{ba} = g^{r_a r_b} \end{cases} \quad (1)$$

式中, g 表示加密元; r_a 、 r_b 为两方产生的随机数。在 RISC-V 架构处理器结构基础上,利用建立会话密钥的指令在源平台上产生一种对称密钥,并将该密钥传送给 RISC-V 体系结构的处理器,使该密钥由 RISC-V 架构处理器接收。

③ 为了保证系统的可靠性,采用 RISC-V 架构处理器实现对处理器的远程验证。源侧的系统将利用一个随机量和一个请求给目标侧的系统提供一条验证信息,然后利用一个会话密钥对请求和该随机量加密。这样就可以获得一条加密生成消息,公式如下:

$$M = \{k_{ab}, k_{ba}\} [R(d) \parallel n] \quad (2)$$

式中, $R(d)$ 表示签名请求; n 表示随机数。将加密生成消息及其哈希值传送给 RISC-V 架构处理器,通过 PIK 算法消去 PCR 数字特征,从而生成一个验证源平台的信息。根据类似流程,在 RISC-V 架构处理器下完成了对来源系统的全面检验,并最终生成了一个用于信息传输的安全信道。

2.2 基于蒙特卡罗法与 Cache 的处理器安全分区

AES 模块是一种面向数据访问的加解密技术,通过复合迭代处理后,以随机产生数据的适配性为出发点,求出各个适配数据在可 Cache 方法进行分区可信通道物理页着色处理信息通道中所占的比重,并通过下式表示:

$$\lambda = \frac{\text{Fit}(f(M_i))}{\sum_{\gamma=1}^i \left(\frac{cf(M_i)}{tM_i} \right)} \quad (3)$$

式中, $\text{Fit}(f(M_i))$ 表示 i 个经过加密的适应度函数; $\frac{cf(M_i)}{tM_i}$ 表示通过复合迭代处理表示的数学形式; t 表示迭代符号; γ 表示适应度。

利用多用户端密钥对称算法,结合物理页着色处理原理计算所有加密结果,公式为:

$$\epsilon = i \cdot \lambda \cdot c_i \quad (4)$$

式中, c_i 表示 i 个随机信息的对称密钥。通过分析输入量与输出量之间的关系,在明文加密前需要进行异或计算,公式为:

$$W = z \oplus c_i \quad (5)$$

式中, z 表示明文。基于此,得到的访问密文可表示为:

$$h_i = (W, \{l_i'\}) \quad (6)$$

式中, $\{l_i'\}$ 表示访问密文集, 该集合包含访问密文地址以及密文计数器值。

采用蒙特卡罗法以求解, 步骤为:

- ① 随机产生分区次数的解向量, 设置计数器数值为 1。
- ② 判断计数器数值与解向量之间的关系, 如果计数器数值小于等于解向量长度, 那么转至步骤③; 否则, 重新分析计数器数值与解向量之间的关系。

- ③ 计算当前分区解向量对应的偏移时间, 计算公式为:

$$t' = \text{Mod}(S \cdot t_s) \quad (7)$$

式中, S 表示帧长度, t_s 表示协议单位时间。

- ④ 判断当前分区偏移时间与截止时间之间的关系, 如果当前分区偏移时间小于截止时间, 那么转至步骤⑤; 否则, 重新分析当前分区偏移时间与截止时间。

- ⑤ 计数器值加 1, 输出解向量和分区偏移时间序列。

⑥ 根据该偏移时间序列, 采用 Cache 方法进行分区可信通道物理页着色处理。Cache 分区利用页面上的颜色原则, 降低了与共享缓存之间的矛盾。在组相联映射的 Cache 中, 要被映射到的 Cache 组是以物理地址中间的特定位置为基础来确定的, 而用这些物理页数和 Cache 组数之间的公共位所表示的就是页面的颜色。

⑦ 通过对分配给一个过程的实体页面的颜色进行控制, 可以将一个过程的数据按页面大小进行对应, 并将其存储到缓冲区。

- ⑧ 将 Cache 分区大小以矩阵乘示例, 表示最终分区结果, 如图 2 所示。

由图 2 可知, 处理器模块 1 被分配到寄存器 1 中, 处理器模块 2 被分配到寄存器 2 中, 处理器模块 3 被分配到寄存器 3 中。当确定各个目标位置之后, 其间的分区关系

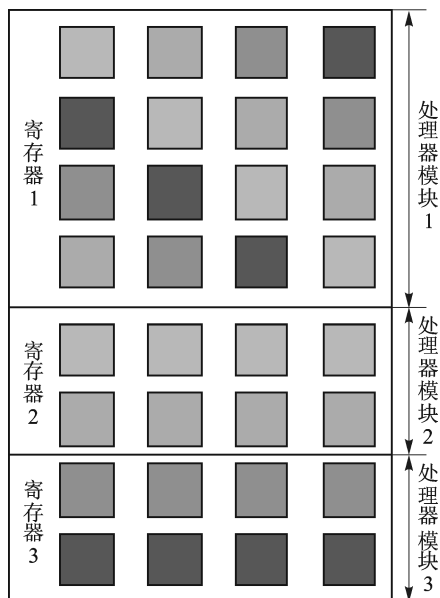


图 2 矩阵乘中处理器安全分区结果

也被确定, 不会出现新行随意替换旧行的情况, 由此获取安全的分区结果。

3 实验

3.1 实验环境

采用 C++ 语言对基于可信执行环境的 RISC-V 架构处理器安全分区方法应用可行性进行验证分析, 实验数据集为符合分区参数的实验数据。

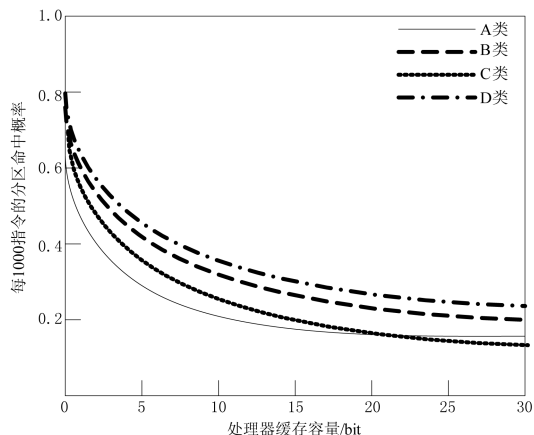
设置 4 类测试程序, 分别为:

- ① A 类: 处理器缓存分区对分区容量变化特别敏感。
- ② B 类: 处理器缓存分区对分区容量变化较敏感。
- ③ C 类: 处理器缓存分区对分区容量变化敏感程度一般。
- ④ D 类: 处理器缓存分区对分区容量变化不敏感。

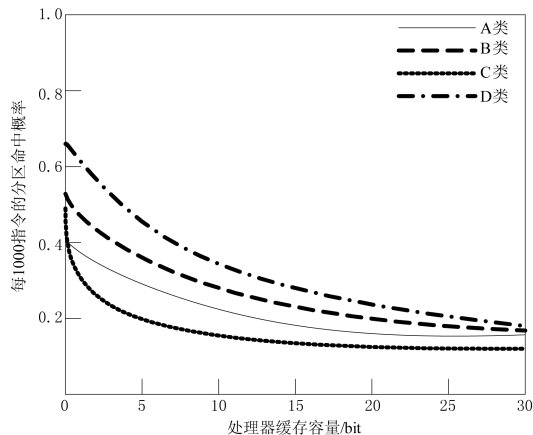
在这 4 类测试程序中, 分别使用基于最优虚拟截止日期多处理器的分区调度方法、基于多核处理器的调度方法和基于可信执行环境分区方法对分区效果进行对比分析。

3.2 处理器分区命中概率分析

使用 3 种方法对比分析 4 类测试程序对应的分区命中概率, 结果如图 3 所示。

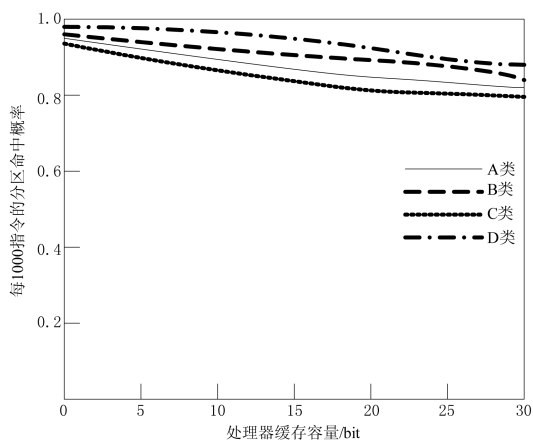


(a) 基于最优虚拟截止日期多处理器的分区调度方法



(b) 基于多核处理器的调度方法

图 3 不同方法分区命中概率



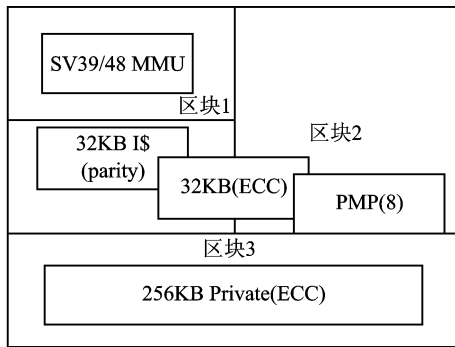
(c) 基于可信执行环境分区方法

图3 不同方法分区命中概率(续)

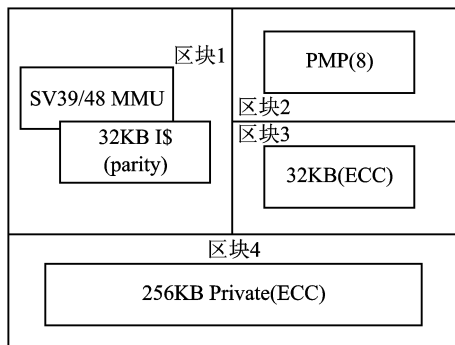
由图 3(a)可知,对于 A、B、C、D 类测试程序,对应的分区命中率最小值分别为 0.18、0.23、0.17、0.27;由图 3(b)可知,对于 A、B、C、D 类测试程序,对应的分区命中率最小值分别为 0.18、0.20、0.16、0.23;由图 3(c)可知,对于 A、B、C、D 类测试程序,对应的分区命中率最小值分别为 0.86、0.88、0.82、0.90。由该分析结果可知,使用基于可信执行环境分区方法分区命中率较高。

3.3 分区结果分析

基于此,使用 3 种方法分区处理器,结果如图 4 所示。

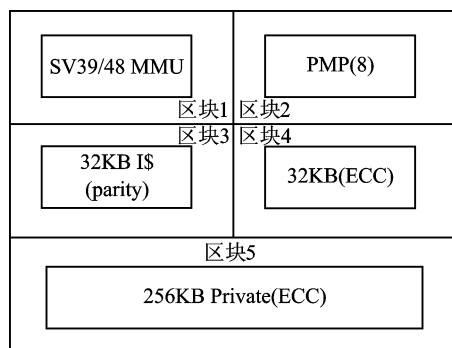


(a) 基于最优虚拟截止日期多处理器的分区调度方法



(b) 基于多核处理器的调度方法

图4 不同方法分区结果



(c) 基于可信执行环境分区方法

图4 不同方法分区结果(续)

由图 4 可知,使用基于最优虚拟截止日期多处理器的分区调度方法只能得到 3 个分区块,其中第 2 区块 3 个部分重叠;使用基于多核处理器的调度方法只能得到 4 个分区块,其中第 1 区块两个部分重叠;使用基于可信执行环境分区方法能得到 5 个分区块,且分开的各个部分未重叠,与处理器未分区模块所包含的内容一致。

4 结 语

现有的分区方法在不安全环境中越来越显出不足,对 RISC-V 架构处理器分区造成不可忽视的负面影响。为此,提出了基于可信执行环境的 RISC-V 架构处理器安全分区方法并得到如下结论:①通过设计分区可信指令执行框图能够为 RISC-V 架构处理器中的安全划分提供安全可执行环境;②通过建立可信执行环境的分区可信通道能够为安全分区提供协商会话密钥;③通过使用蒙特卡罗法和 Cache 物理页着色方法能够实现 RISC-V 架构处理器安全分区。

参考文献

- [1] 高启明,吴莉莉.基于最优虚拟截止日期多处理器混合时序调度[J].计算机工程与设计,2021,42(5):1376-1382.
- [2] 梁秋玲,张向利,张红梅,等.基于多核处理器的关联任务并行感知调度算法[J].计算机工程,2021,47(7):212-217.
- [3] 成元虎,黄立波,崔益俊,等.基于 RISC-V 的嵌入式多指令集处理器设计及实现[J].电子学报,2021,49(11):2081-2089.
- [4] 牛世权.轻量级安全内存:RISC-V 嵌入式微处理器安全增强[J].计算机工程与科学,2021,43(8):1360-1365.
- [5] 姜冕,张海金,杨靓,等.基于 Chisel 语言的 RISC-V 处理器设计技术[J].微电子学与计算机,2021,38(3):51-55.
- [6] 徐子晨,崔傲,王玉峰,等.基于 RISC-V 架构的强化学习容器化方法研究[J].计算机工程与科学,2021,43(2):266-273.
- [7] 刘有耀,潘宇晨.基于 RISC-V 参数化超标量处理器的优化设计[J].计算机工程与应用,2022,58(5):66-74.

现每个 CPU 的 1 GHz 操作,本地时钟偏斜在 150 ps 左右进行优化。

3.4 时钟延迟调整器

过小的时钟偏斜可能会增加峰值噪声,为了减小动态峰值电流,必须使用时钟延迟调整器来调整每个 CPU 的时钟路径延迟。时钟延迟调整器(Clock Delay Adjuster, CDA)由延迟元件和选择器组成,其分辨率为 78 ps。CDA 微调每个处理器的时钟相位,使用这个电路可以有效抑制动态峰值电流和噪声。图 4 显示了 3 个芯片的 CDA 评估结果,在这个评估中,7 个 CPU 的时钟相位被对齐,只有 1 个 CPU 的 CDA 设定值被改变,水平轴显示设定值,垂直轴显示目标 CPU 的最大工作频率,该 CPU 具有不对齐的时钟,当重叠消除时,电压下降减小,目标 CPU 的频率增加。目前的结果显示,通过调整时钟相位,频率可以提高约 4%。图 4 还显示了两个操作频率较低的点,一个大约在 0 ps 附近,另一个大约在 500 ps 附近。在前者点,所有 CPU 的时钟相位匹配,因此电压下降很大;在后者点,由于该 SoC 中的 SRAM 在时钟负边沿附近消耗大量电流,因此在该 CDA 调整方法的帮助下有效提升了可以实现的频率。

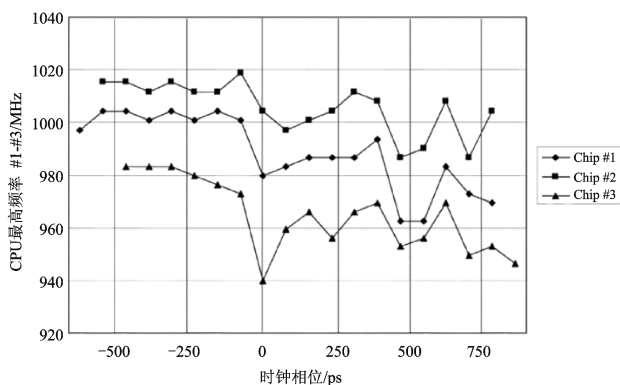


图 4 时钟延迟调整器的评估

4 硅芯测试结果与总结

测试芯片采用 90 nm 通用 CMOS 工艺制造,具有 8 层铜互连。芯片采用 729 引脚的翻转芯片球栅阵列封装。图 5 显示了芯片时钟频率的 shmoo 图。

本文开发了一款集成了 3 种可合成处理器的高速多核 SoC,可配置的异构多核 SoC 相比传统的对称多 CPU 系统降低了 45% 的功耗。1 GHz 的 CPU 和 76 GFLOPS 的 MBMX (500 MHz 运行)通过使用微小的延迟监测器和微控制器来实现“任务跳转”。这是一种在未来应用或工艺技术中有潜力的技术。

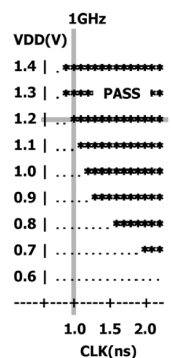


图 5 CPU shmoo 图

参考文献

- [1] 陈诚,皮志勇,赵英龙,等.基于自适应灾变遗传-循环神经网络的锂离子电池 SOC 估计[J].电气工程学报,2022,17(1):86-94.
- [2] 季永康,景乃锋,王琴.面向异构 SoC 的串并混合总线结构设计与研究[J].信息技术,2020,44(6):41-45.
- [3] 闫瑾.SOC 技术与发展预测[J].信息技术,2011,35(9):188-190.
- [4] 雷思磊.RISC-V 架构的开源处理器及 SoC 研究综述[J].单片机与嵌入式系统应用,2017,17(2):56-60,76.
- [5] 张发荣.面向算法的 SIMD 计算机数学模型及其应用[J].电子技术,2022,51(3):48-49.
- [6] 陈梦清,陈洋,陈志环,等.路网约束下异构机器人系统路径规划方法[J].自动化学报,2023,49(4):718-730.
- [7] NODA H, NAKAJIMA M, DOSAKA K, et al. The design and implementation of the massively parallel processor based on the matrix architecture[J]. IEEE Journal of Solid - State Circuits, 2006, 42(1):183-192.
- [8] KANEKO S, KONDO H, MASUI N, et al. A 600 - mhz single - chip multiprocessor with 4.8 - gb/s internal shared pipelined bus and 512 - kb internal memory[J]. IEEE Journal of Solid - State Circuits, 2004, 39(1):184-193.
- [9] 左超,陈钱.计算光学成像:何来,何处,何去,何从[J].红外与激光工程,2022,51(2):158-338.

潘斌(高级工程师),主要研究方向为电力系统自动化。通信作者:潘斌,mvqjre@163.com。

(责任编辑:薛士然 收稿日期:2023-05-22)

- [8] 孙彩霞,郑重,邓全,等.DMR:兼容 RISC-V 架构的乱序超标量通用处理器核[J].计算机研究与发展,2021,58(6):1230-1233.
- [9] 黄正伟,刘宏伟,徐渊.用于 ITof 传感器的极低功耗 RISC-V 专用处理器设计[J].计算机工程,2022,48(9):146-154.
- [10] 杨凯峰,张雪倩,冯娇,等.基于遗传进化策略的处理器验证方案[J].微电子学与计算机,2021,38(5):25-29.

王健(高级工程师),主要研究方向为通信工程、网络安全;付志博(工程师),主要研究方向为信息安全;明哲(高级工程师),主要研究方向为电力自动化、网络安全。通信作者:王健,yixile6833@163.com。

(责任编辑:薛士然 收稿日期:2023-04-14)