基于 SRAM 的 FPGA 上 RISC-V 软处理器的时序脆弱性感知选择性加固

Mostafa Darvishi，*IEEE* 高级会员

摘要—**选择性加固被广泛用于提高基于 FPGA 的软处理器的可靠性，同时限制完全冗余带来的开销。然而，现有方法主要依赖架构关键性或功能故障分析，忽略了依赖布线的时序敏感性对处理器鲁棒性的影响。本文提出一种面向在基于 SRAM 的 FPGA 上实现的 RISC-V 软处理器的时序脆弱性感知选择性加固方法。**

基于原位时序可观测性的最新进展，本文提出的方法量化了受控布线扰动下流水线组件的统计时序敏感性，并利用这些信息指导加固决策。在商用 FPGA 平台上实现的 RISC-V 处理器实验结果表明，时序脆弱性较高的组件也更容易受到布线引起的延迟效应影响。利用这种相关性，所提出的选择性加固策略在显著降低面积及时序开销的同时，实现了与全面加固相当的鲁棒性。

这些结果表明，时序脆弱性是基于FPGA的处理器架构进行可靠性感知设计优化的一项实用且有效的度量指标。

索引词—**RISC-V、选择性加固、FPGA可靠性、流水线容错、时序退化、冗余技术、三模冗余（TMR）**

I. 引言

基于FPGA的软处理器，尤其是RISC-V实现，凭借其灵活性和快速的设计迭代，正越来越多地应用于嵌入式、航空航天和自适应计算系统。在这些环境中，可靠性至关重要，因此人们广泛研究容错处理器架构和选择性加固技术 [1], [2]。

以部分三模冗余（TMR）或复制关键模块为代表的选择性加固方法，已被证明能够显著提高鲁棒性，同时避免完全冗余带来的高昂开销 [3], [4]。然而，现有的选择性加固方法通常依据架构重要性、执行频率或功能故障覆盖率来做出决策。尽管这些方法在宏观层面上有效，但它们都隐含地假设处理器芯片布线结构中的物理行为具有一致性。

实际上，基于现场可编程门阵列的处理器的时序行为会受到布线拓扑、互连寄生效应和电源分配效应的显著影响 [5], [6]。近期研究表明，时序退化往往先于功能故障发生，并且布线引起的延迟扰动会显著影响处理器鲁棒性 [7], [8]。然而，时序敏感性尚未被纳入选择性加固决策的一级标准。

本文针对这一空白，提出一种面向时序脆弱性的 RISC-V 软处理器选择性加固框架。基于此前为基于 SRAM 的 FPGA 建立的在线时序诊断架构，我们利用统计时序可观测性，识别对布线引起的延迟效应尤为敏感的流水线组件。该方法不再将所有架构关键模块一视同仁，而是根据测得的时序脆弱性来确定加固优先级。

基于 FPGA 的 RISC-V 实现，我们证明时序脆弱性与对布线扰动的易受影响程度高度相关，并且依据这一指标进行选择性加固，能够以显著较低的开销实现与全面加固相当的鲁棒性。这些结果表明，时序感知分析为 FPGA 软处理器的可靠性驱动设计优化提供了一个强大且此前未充分利用的维度。

II. 背景与相关工作

基于 SRAM 的 FPGA 凭借灵活性和快速部署能力，已成为在安全关键、空间受限及自适应计算系统中实现软处理器的广泛采用平台 [9]–[10]。然而，使用配置存储器定义逻辑电路和布线资源，带来了固有的可靠性挑战。存储在 SRAM 单元中的配置位易受电压波动、老化和环境因素造成的干扰，可能导致已部署设计出现功能故障或性能退化 [11], [12]。

在现场可编程门阵列上实现的软处理器（如 RISC-V内核）对这些影响尤为敏感，因为其正确性不仅取决于逻辑功能，还取决于能否在深度流水线数据通路中满足严格的时序约束 [13]–[18]。与专用集成电路实现不同，基于 FPGA 的处理器高度依赖可编程布线资源，而这些资源的延迟特性在芯片布线结构中可能存在显著差异 [19]。因此，FPGA 软处理器面临的可靠性威胁往往表现为 时序故障，而非显式的逻辑错误 [6], [19]–[21]。

传统的现场可编程门阵列处理器容错技术主要关注功能正确性，通常假设时序裕量能够保持不变 [22]–[24]。

Mostafa Darvishi 任职于加拿大蒙特利尔高等技术学院（ÉTS）电气工程系，同时担任 Evolution Optiks研发公司的工程副总裁（电子邮件：darvishi@ieee.org）。

然而，越来越多的证据表明，时序退化——尤其是在布线主导路径中——是现代现场可编程门阵列实现中一种关键且尚未得到充分研究的故障模式 [25]-[29]。

为解决完全复制或三模冗余（TMR）带来的高面积和功耗开销，先前的工作 [30] 探索了选择性加固策略，仅保护部分处理器组件。这些方法通常依赖架构或功能分析来识别会对系统正确性产生不成比例影响的关键模块，例如寄存器文件、控制逻辑或流水线阶段。

[31] 和 [32] 中介绍的工作代表了这一范式的先进成果。在这些研究中，通过识别架构上的关键元素，并有选择地复制或保护这些元素，对 RISC-V 软处理器实施选择性加固，以提高可靠性。选择标准主要依据功能脆弱性指标、架构作用或指令级影响，因此与全面冗余相比，可大幅降低开销。

尽管这些方法能有效缓解某些类型的故障，但它们默认将给定架构结构的所有实例视为等同。也就是说，它们在处理器组件（例如流水线寄存器、执行单元）层面做出保护决策，却未考虑这些组件在 FPGA fabric 中的物理实现环境。

现有选择性加固技术的一项关键局限，在于其保护策略缺乏物理时序感知。在基于现场可编程门阵列的实现中，两个逻辑上完全相同的模块，其时序行为可能因布局、布线拓扑以及与周围互连资源的相互作用而存在巨大差异。因此，架构关键性并不一定与时序脆弱性相关。

以往的研究并未明确测量或利用处理器各处由布线引起的延迟敏感性差异。因此，选择性加固设计可能会过度保护时序稳健区域，却让时序脆弱路径暴露在风险之中。这种不匹配会导致冗余资源利用效率低下，并限制实际运行条件下可实现的可靠性提升。

此外，现有研究大多依赖静态分析、功能级故障注入或体系结构建模来评估可靠性。这些方法难以充分揭示运行过程中时序行为在 FPGA fabric 上如何演变，以及局部布线扰动如何转化为可观测的延迟变化。

近年来，原位时序测量技术的进步已证明，可以直接在现场可编程门阵列的 FPGA fabric 内观测布线级时序行为。此类方法能够对设计不同区域的延迟分布、变异性和空间相关性进行细粒度表征。这些测量结果表明，布线寄生效应和配置引起的扰动等时序退化机制具有很强的局部性且分布不均。

这一观察促使我们探索一类新的选择性加固策略：不再局限于纯粹的体系结构考量，而是将实测时序脆弱性纳入首要选择标准。通过识别处理器中对布线引起的延迟变化最敏感的部分，可以将冗余和加固资源部署在影响最大的地方。

与以往工作不同，时序脆弱性感知方法能够依据实测物理行为，而非仅依据抽象的架构重要性来指导保护决策。这一范式有望提高可靠性效率，在降低开销的同时，直接应对基于 FPGA 的软处理器中导致时序故障的主要因素。

本文借鉴 [31] 和 [32] 中提出的选择性加固理念，并利用与近期时序诊断研究所探索的方法思路相近的原位时序可观测机制，提出了一种面向基于 SRAM 的 FPGA 上实现的 RISC-V 软处理器的时序脆弱性感知选择性加固框架。

与以往方法不同，本文提出的方法明确纳入空间分辨时序测量，以指导加固决策。通过将时序脆弱性与体系结构结构相关联，该框架能够仅对那些在架构上重要且在物理上易受影响的处理器区域实施选择性保护。将物理时序行为与架构加固相结合，是本文工作区别于现有研究的关键所在，也构成了本文后续贡献的基础。

III. 时序脆弱性及选择性加固的动机

基于 SRAM 的 FPGA 的时序可靠性不仅取决于标称延迟裕量，还取决于布线信号路径对可编程互连结构中扰动的敏感程度。传统静态时序分析（STA）可确保在最坏情况假设下满足建立时间和保持时间约束，但对于设计各部分在变异性、噪声或配置引起的扰动下运行时距离其有效时序裕量有多近，它提供的洞见有限。在基于 FPGA 的软处理器中，逻辑功能与布线资源紧密交织，这一局限尤为明显。

实际上，时序脆弱性在处理器流水线中的分布并不均匀。即使所有流水线级均满足标称时序约束，它们对布线引起的延迟变化的敏感程度也可能因布线深度、扇出和开关矩阵遍历的差异而大不相同。本节将时序脆弱性确立为一种可测量且与体系结构相关的属性，并阐明其为何构成选择性加固的关键动机。

图 1 展示了用于定位

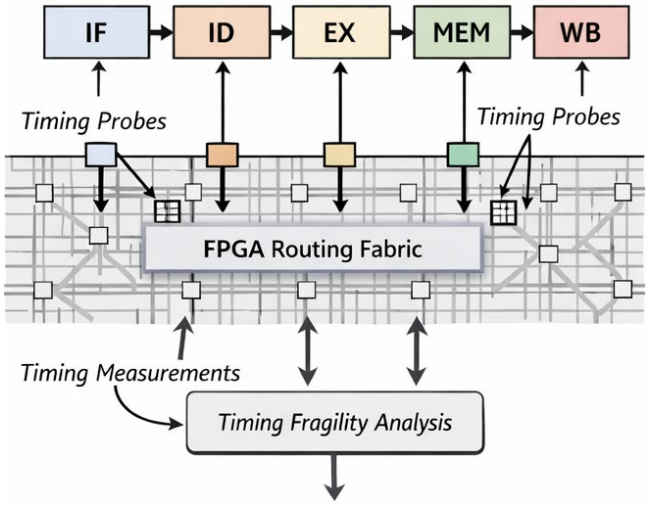

图 1. 基于现场可编程门阵列的 RISC-V 处理器中按流水线阶段划分的时序脆弱性映射。分布式时序探针用于观测与各流水线阶段相关的布线级延迟行为，从而能够局部识别时序稳健区域和时序脆弱区域。

基于 FPGA 实现的 RISC-V 处理器中时序脆弱性的概念框架。处理器流水线被分解为各个组成阶段——取指级（IF）、译码级（ID）、执行级（EX）、存储器访问（MEM）和写回级（WB），每个阶段均映射到 FPGA fabric 上的不同区域。与这些阶段相关的信号会经过由开关矩阵和互连线段组成的可编程布线结构，而这些结构的电气特性会因器件上的空间位置不同而有所差异。为在不干扰功能执行的情况下观察时序行为，我们在与各流水线阶段对应的代表性布线路径上连接非侵入式时序探针。这些探针直接从已布线的信号中提取统计时序信息，从而能够按阶段进行表征，同时保持处理器全面正常运行。

虽然图 1 突出了时序脆弱性在架构上的分布位置，但要了解脆弱性如何显现，还需考察边缘时序条件下信号转换的统计行为。为此，我们采用相位扫描采样来表征时序行为，从而以概率视角呈现信号到达时间，而非给出二元的通过／失败结果。

图 2 展示了从处理器中提取的两条布线路径具有代表性的误码率（BER）与采样相位特性：一条表现出时序稳健性，另一条则表现出时序脆弱性。两条路径均满足标称 STA 约束，但其统计特征差异显著。时序稳健路径的过渡区域较窄，BER 随采样相位急剧变化，表明其转换点明确，且对微小延迟变化的敏感性有限。相比之下，时序脆弱路径的过渡区域明显更宽，方差也更大，反映出其对布线引起的延迟扰动更加敏感；这些扰动源于互连寄生效应、电源分配噪声或局部配置效应。

图 2 所示的转换宽度和变异性

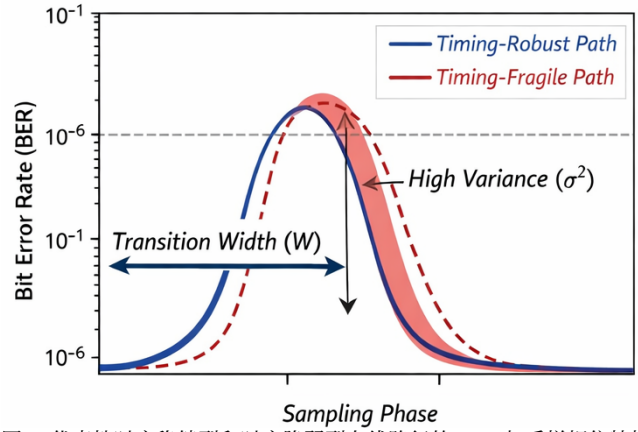

图 2. 代表性时序稳健型和时序脆弱型布线路径的 BER 与采样相位特性关系。脆弱路径的过渡区域更宽、方差更大，表明尽管满足标称时序约束，它们对延迟扰动仍更为敏感。

构成了本文所用定量时序脆弱性指标的基础。具体而言，BER 过渡区域的宽度直接表明一条路径对延迟扰动的敏感程度，而多次测量中转换位置的变异性则反映了该路径在临界条件下的稳定性。这些指标揭示了标称裕量值未能体现的时序脆弱性。

这些统计差异对处理器流水线的影响总结于表 I 中。该表报告了从已实现的 RISC-V 处理器中提取的代表性时序脆弱性指标，并用于为选择性加固决策提供依据。第 VI节将给出全面的实验评估。所有阶段均满足标称静态时序约束，正裕量范围为 0.33ns 至 0.46ns。然而，统计测量结果显示，时序稳健性存在显著的不均匀性。

表 I 显示，尽管标称裕量相近，执行级（EX）和存储器访问（MEM）阶段的 BER 过渡区域明显更宽，时序变异性也高于取指级（IF）和译码级（ID）。这些差异归因于流水线后级布线深度和扇出的增加，这会放大其对布线级延迟扰动的敏感性。相比之下，即使静态时序裕量相近，布线结构更简单的阶段也表现出更强的内在鲁棒性。

根据这些观察结果，流水线级被划分为低、中等和高脆弱性类别，如表 I 所示。这种分类直接基于测得的统计行为，而非启发式假设，并表明在真实运行条件下，仅凭静态时序裕量不足以识别时序脆弱区域。这些结果促使我们采用选择性加固策略，优先将缓解资源用于统计上较脆弱的阶段，在解决时序脆弱性的主要来源的同时降低开销。

表 I. 流水线级时序特性与统计脆弱性指标。

流水线阶段	标称裕量 (ns)	Mean 延迟 μ (φ)	转换宽度 Δφ (φ)	Std. Dev. σφ (φ)	脆弱性
IF	0.46	0.48	0.015	0.003	Low
ID	0.41	0.51	0.019	0.005	Low
EX	0.33	0.57	0.041	0.014	High
MEM	0.36	0.60	0.046	0.017	High
WB	0.43	0.53	0.022	0.006	中等

总之，本节将时序脆弱性确立为基于 FPGA 的 RISC-V 处理器中一种可测量且依赖于流水线的属性。通过结合分级定位（图 1）、路径统计表征（图 2）以及对各流水线级的定量比较（表 I），本节为下一节提出的选择性加固框架奠定了严谨的基础。

IV. 选择性加固架构

本节介绍一种面向基于 FPGA 的 RISC-V 软处理器的选择性加固架构。该架构旨在利用第 III 节提出的时序脆弱性表征，有针对性地缓解布线引起的时序漏洞，同时最大限度地降低面积、功耗和性能开销。所提方法不对整个处理器采用均匀冗余，而是仅对在统计意义上表现出显著时序脆弱性的架构区域进行选择性加固。

该架构的核心设计原则是将架构重要性与物理时序脆弱性解耦。以往的选择性加固方法主要关注架构角色，例如控制逻辑或寄存器文件；而所提架构则将时序脆弱性作为一项基于物理特性的附加标准。这样便能将保护资源分配到既具有架构相关性、又存在物理脆弱性的部分。

A. 加固粒度与目标选择

为缓解时序脆弱性并保持实现效率，本研究采用选择性加固策略，仅针对处理器流水线中时序最关键的部分进行加固。所提方法不会对整个核心进行均匀加固，因为这会带来难以承受的面积、功耗和性能代价；相反，它利用流水线级时序表征，将加固限定在收益最高的部分。

图 3 展示了在基于 SRAM 的 FPGA 上实现的 RISC-V 处理器核心的选择性加固架构。基线流水线由指令获取（IF）、指令译码（ID）、执行（EX）、存储器访问（MEM）和写回（WB）阶段组成。在所提出的方法中，仅对执行（EX）和存储器访问（MEM）阶段选择性地应用冗余，而其余阶段则保留原始的、未经加固的实现。这一选择是因为在典型的 RISC-V 微架构中，执行（EX）和存储器访问（MEM）阶段共同占据了关键路径深度、布线复杂度及时序敏感性的大部分。

在图 3 所示的所提架构中，选择性

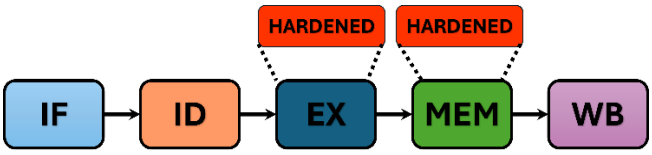

图 3. 对五级 RISC-V 处理器中时序脆弱的流水线阶段进行选择性加固。选择性地在执行（EX）和存储器（MEM）阶段引入加固，因为这两个阶段是时序关键性和延迟变化的主要来源；取指级（IF）、译码级（ID）和写回级（WB）则保持不变，以尽量减少面积和功耗开销。

加固执行级和 MEM 级（以 HARDENED 模块表示）可根据可靠性要求，采用复制比较或三模冗余（TMR）实现。在这两种情况下，仅复制目标流水线级，并在级间边界插入轻量级比较或表决逻辑，以便在损坏数据向下游传播之前检测或屏蔽故障。重要的是，这种局部插入冗余的方式避免引入全局反馈路径或较长的表决器链，否则这些结构可能会加剧布线拥塞并妨碍时序收敛。

与全核加固方法相比，例如 [31] 和[32], 中采用的、在所有流水线阶段和控制逻辑中统一应用复制或三模冗余（TMR）的方法，所提出的选择性策略在保护影响最显著的部分的同时，显著降低了结构开销。全核加固能够提供统一的故障覆盖率，但会在逻辑资源利用率、布线需求和功耗方面带来较大代价，而这些代价反过来可能引入新的时序漏洞。相比之下，选择性加固使保护粒度与架构关键性相匹配。

为在架构层面量化这些权衡，`总结了选择性加固与全面加固策略所对应的预期开销范围。这些数值反映了先前研究 [7], [30], [31], [32] 和厂商工具分析报告的典型现场可编程门阵列综合行为，并根据受保护的流水线比例进行缩放。当冗余仅限于执行级和MEM级时，预计选择性复制会增加约 15–25% 的查找表/触发器开销，并使动态功耗增加 8–18%；选择性三模冗余则会使这两项分别增至约 25–35% 和 15–25%。相比之下，[31] 和 [32], 采用的完全复制和全核三模冗余方案，其逻辑开销通常分别超过 80% 和 180%，相应的功耗增幅可接近或超过 100%，同时还会造成严重的布线拥塞。

至关重要的是，这些开销预期表明，选择性加固不仅能降低资源消耗，还能缓解布线复杂度过高所导致的次生时序退化。通过仅在最脆弱的流水线阶段引入冗余，所提架构在可靠性、性能与实现成本之间实现了更理想的平衡。选择性加固机制的具体实现以及

表 II. 基于FPGA的RISC-V内核中选择性加固策略与全面加固策略的预期开销范围

加固策略	受保护范围	查找表／触发器开销	动态功耗开销	布线拥塞影响
选择性复制	执行级 + MEM 流水线仅阶段	15% – 25%	8% – 18%	Low
选择性 TMR	执行级 + MEM 流水线仅阶段	25% – 35%	15% – 25%	中等
Full 复制	整个核心（所有阶段 + 控制）	80% – 120%	40% – 70%	High
全核三模冗余	整个核心（所有阶段 + 控制）	180% – 250%	90% – 150%	非常高

其实验验证将在后续章节中介绍。

B. 选择性加固机制与流水线集成

所提架构通过仅在被认定为可靠性关键的流水线阶段引入冗余，同时保持流水线其余部分不变，从而实现选择性加固。如图 4 所示，加固仅应用于执行（EX）和存储器访问（MEM）阶段，而取指（IF）、指令译码（ID）和写回（WB）阶段则保留原有的单实例实现。这种选择性部署避免了全核加固技术所带来的过高面积和布线开销。

图 4 明确展示了两种受支持的冗余机制：复制比较和三模冗余（TMR）。在基于复制的配置中（图 4 上半部分）执行级的两个相同副本（EX_A 和 EX¯B）对来自译码级的相同输入操作数并行执行运算。它们的输出由位于执行级边界处的 EX比较器进行比较。两个副本之间的任何不匹配都会触发执行级_错误_标志，表明存在故障。MEM级也采用类似结构：MEM_A 和 MEM¯B 的输出在写回之前送入 MEM比较器；若两者不一致，则会生成 *MEM*_错误_标志。

在基于TMR的配置中（图 4 下半部分），执行级的三个副本（EX_1、EX_2、EX_3）和 MEM级的三个副本（MEM_1、MEM_2、MEM_3）并行执行。放置在执行级和 MEM 级边界处的多数表决器能够在一个副本发生故障时选出正确输出值。除了屏蔽故障外，表决器还会在检测到副本之间存在分歧时生成阶段级错误标志。与复制配置一样，所有冗余逻辑都严格限制在加固级的边界之内。

流水线集成通过在馈送后续阶段的流水线寄存器之前立即插入比较或表决逻辑来实现，如图 4 中 EX→MEM 和 MEM→WB 接口所示。这样布局可确保在加固阶段内产生的错误数据在沿流水线进一步传播之前得到检测或纠正。重要的是，不会引入全局反馈路径或跨阶段依赖，从而保持加固逻辑的局部性。

错误信号通过专用的阶段级错误标志进行处理，这些标志在图 4 中明确标记为执行级_错误¯标志和*MEM*_错误_标志。这些标志被传送至错误汇总器，由其汇总多个加固级的故障指示。汇总后的故障信息随后存储在故障状态寄存器中，从而集中、持久地记录已检测到的错误。此错误报告路径在逻辑上与数据通路分离，且不位于任何时序关键信号路径上。

流水线控制单元与故障状态寄存器交互，以便采取适当的系统级响应，例如流水线停顿、指令重放或异常处理，而无需对核心流水线逻辑进行结构性修改。由于加固级保留了与基线对应模块相同的输入—输出接口，现有控制机制（包括冒险检测、旁路转发和调度）均保持不变。

选择性加固激活由加固配置寄存器控制，如图 4 所示。该寄存器决定哪些流水线级需要加固，以及采用复制还是三模冗余。通过将加固控制外置到配置寄存器中，该体系结构无需重新综合或重新设计，即可支持灵活的可靠性—性能权衡。

与 [31] 和 [32], 中报道的全核复制和三模冗余方法相比，图 4 所示的选择性机制大幅限制了逻辑复制和互连增长。通过使冗余与体系结构脆弱性相匹配，所提设计以显著较低的面积、功耗和时序开销实现了更高的故障覆盖率，因此非常适用于可靠性和性能都至关重要的基于FPGA的RISC-V系统。

C. 加固配置与开销权衡

所提选择性加固框架的核心目标之一，是提供细粒度的可配置性，使可靠性机制能够根据应用需求和运行条件进行定制，同时明确控制实现开销。不同于不考虑脆弱性分布而采用均匀冗余的全核加固方法，所提架构能够在单个流水线阶段和控制边界层面选择性地启用保护机制。

通过可参数化的加固控制信号实现配置灵活性，这些信号用于控制：(i) 加固哪些流水线阶段；(ii) 所采用的冗余类型——复制比较或三模冗余（TMR）；以及 (iii) 检测到错误后触发的故障处理策略。如图 4 所示，这些选择由专用的加固配置寄存器管理，该寄存器可启用或禁用级复制，并选择适当的比较或表决逻辑。根据系统需求，这些控制可在综合时静态定义，也可动态

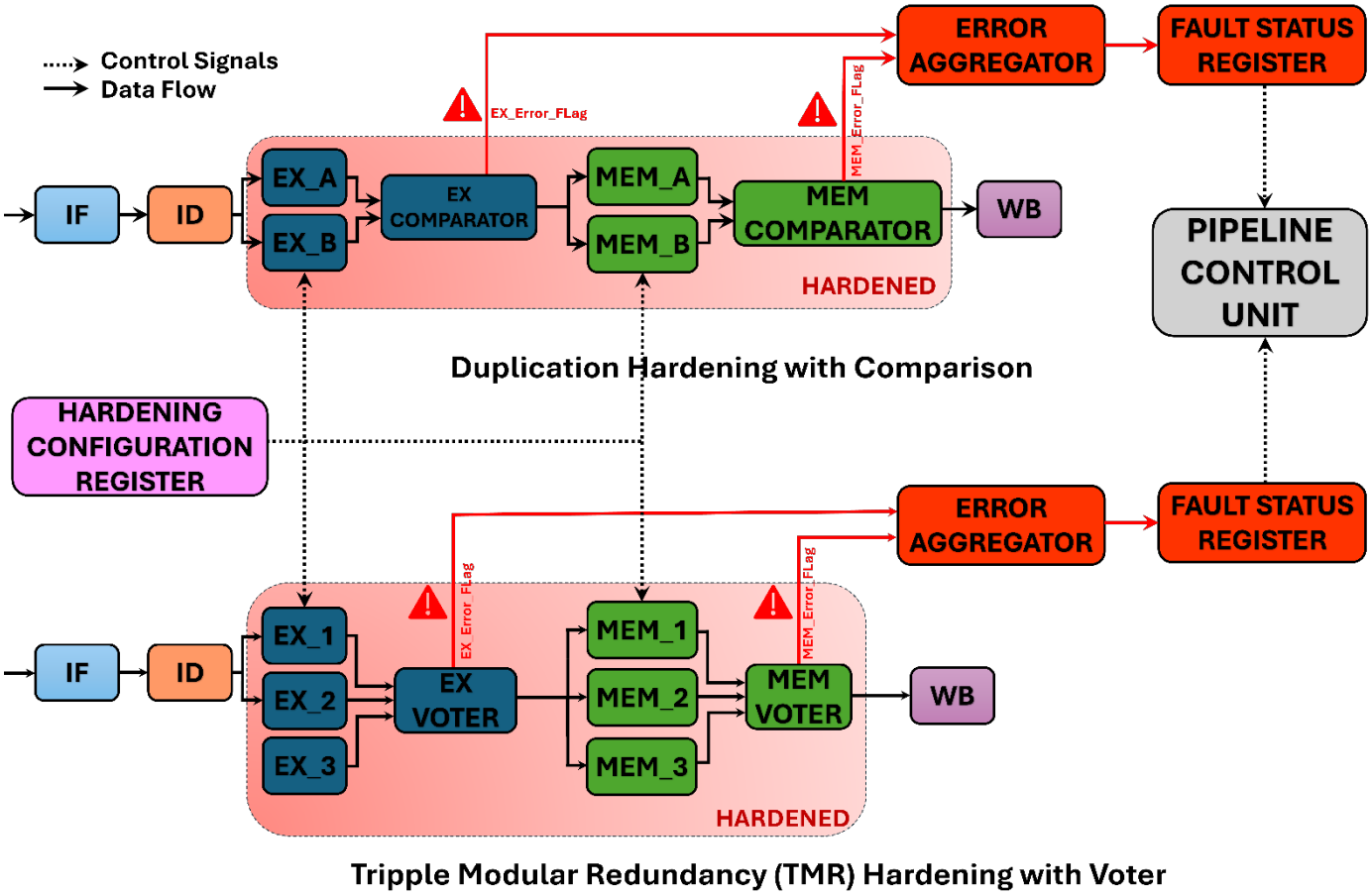

图 4. 流水线级间边界处的选择性加固集成。图中展示了应用于 RISC-V 流水线执行级和 MEM 阶段的复制加固（图的上半部分）与三模冗余（TMR）加固（图的下半部分）。比较逻辑或表决逻辑在级间边界处局部插入，而取指级、译码级和写回级不进行加固。

在运行时编程，从而使单个处理器实例能够在多种可靠性模式下运行。

这种可配置性直接支持对开销与可靠性进行权衡，相关内容总结于表 II。选择性复制仅复制目标流水线级，并在级间边界插入轻量级比较逻辑，因此只带来适度的面积和功耗开销。选择性三模冗余会进一步增加开销，但能够提供故障屏蔽能力，防止单粒子事件故障传播到加固级之外。相比之下，完全复制和全核三模冗余会复制整个流水线，并引入全局比较或表决逻辑，导致面积、功耗和布线开销大幅增加。

重要的是，选择性加固将冗余及其相关控制逻辑限制在流水线的局部区域内，如图 4 中阴影标出的加固区域所示。这种局部化减少了长互连路径的引入，并避免使用可能影响时序收敛的集中式表决器或比较器。如表 II 所示，与全核对应方案相比，选择性方案始终具有更低的开销，同时仍能针对最容易发生时序退化和瞬态故障的流水线级。

从设计角度看，这种灵活性使系统架构师能够根据应用特定约束来调整保护级别。例如，安全关键型工作负载可以在执行级和存储器级启用选择性三模冗余，而能量受限应用则可以采用选择性复制，或在良性运行条件下完全禁用加固。这种适应性使所提出的框架有别于以往的全核加固技术，并使其成为基于 FPGA 的系统中可靠性感知型 RISC-V 设计的实用解决方案。

D. 与流水线控制和异常处理的集成

要使选择性加固适用于高性能 RISC-V 处理器，就必须与现有的流水线控制、冒险管理和异常处理机制无缝集成。所提架构专为在标准流水线控制框架内运行而设计，确保引入冗余和错误信号不会影响功能正确性或时序确定性。

在流水线级层面，加固级与其未加固对应级同步运行，并保留原有的级间边界。采用复制比较时，比较器逻辑置于受保护级的输出端，仅在检测到副本之间不匹配时生成错误标志。在选择性三模冗余的情况下，表决器生成单一的屏蔽输出

并可选择在观察到副本之间存在不一致时发出错误指示。在这两种情况下，加固级都会向下游逻辑提供单一且时序兼容的输出，从而能够与未经修改的流水线级透明地连接。

如图 4 所示，在执行级和存储器级生成的错误标志通过专用控制路径传送至错误聚合单元和故障状态寄存器，而不是反馈至数据通路。这些状态信号在明确界定的时钟边界由流水线控制单元采样，从而确保故障检测不会引入异步行为，也不会违背流水线时序假设。检测到错误后，控制单元可根据配置的故障处理策略，采取标准恢复操作，例如停顿、冲刷或重放受影响的指令。

至关重要的是，所提出的框架无需修改体系结构状态或指令集。错误处理利用现有的异常和中断机制，使已检测到的故障能够以可恢复异常、机器检查事件或记录的诊断信号形式呈现给软件。这既保留了与现有 RISC-V 软件栈的兼容性，也避免了定制具备故障感知能力的指令语义。

选择性加固也能够自然地融入流水线冒险管理。由于冗余仅限于各个阶段，且不会引入全局反馈环或长依赖链，现有的冒险检测和旁路逻辑无需修改即可继续正常工作。比较器和表决器逻辑被置于依赖解析网络的关键路径之外，从而最大限度地降低其对流水线时序的影响。

E. 设计可扩展性与可移植性

所提出的选择性加固框架的一个重要目标，是在支持日益复杂的处理器设计扩展的同时，保持其在不同实现平台上的可移植性。因此，该体系结构能够随流水线深度、发射宽度和核心数量扩展，而不会引入集中式瓶颈或僵化的设计假设。

在单核层面，通过将冗余和错误检测逻辑限制在各个流水线级内，实现可扩展性。由于加固级独立运行，并通过现有的流水线控制接口进行通信，因此可以保护更多流水线级，而无需修改先前已加固的逻辑。这种模块化组织使选择性加固能够随受保护级数量线性扩展，避免了全核冗余方案中常见的控制和表决器复杂度呈二次增长的问题。

尽管图 4 为清晰起见展示了应用于执行级和存储器级的选择性加固，但同样的集成模式适用于任何流水线阶段或集群式功能单元。对于更宽或更深的流水线，例如超标量或乱序 RISC-V 处理器核，可在体系结构关键边界（例如提交阶段或控制流解析点）应用选择性复制或三模冗余，使设计人员能够以受控方式平衡覆盖率与开销。由于错误信号依赖现有的控制和异常路径，因此扩展到更复杂的流水线时，无需重新设计全局故障处理机制。

该框架还支持在多核系统中扩展。由于每个处理器核都拥有各自的本地加固配置和错误报告路径，因此可根据工作负载关键性，在不同处理器核上采用非对称的选择性加固。这种局部化方法避免了核间依赖，并使加固开销随受保护处理器核的数量成比例增长，而非随系统总规模增长。

从可移植性的角度来看，所提架构基本不依赖底层实现技术。虽然本研究针对基于现场可编程门阵列的 RISC-V 处理器，但其加固结构——阶段级复制、比较或表决逻辑，以及控制路径错误传播——都能自然地映射到专用集成电路设计中。在这类实现中，对时序和功耗进行更精细的控制，或可进一步降低相较于全核冗余的开销。此外，由于选择性加固在寄存器传输级运行，并依赖标准流水线接口，因此只需进行少量调整，便可在不同 FPGA系列和器件代际间复用。

F. 设计权衡总结

本节介绍的选择性加固架构经过精心设计，旨在平衡可靠性提升与实现开销、时序影响和设计复杂度。与全核冗余方案不同，后者以显著增加面积、功耗和布线开销为代价提供统一保护；所提框架则能够对处理器流水线中应用容错的位置和方式进行细粒度控制。

在体系结构层面，主要权衡在于保护覆盖范围与开销之间的取舍。选择性复制和选择性三模冗余使设计人员能够仅加固最易受故障影响或在体系结构上至关重要的流水线级，与全核方案相比，可显著减少冗余逻辑。然而，这种选择性必然使未加固级暴露于故障之下，因此准确识别关键区域就显得尤为重要。框架的可配置性可缓解这一权衡，使保护边界能够随着工作负载特征或可靠性要求的变化而调整。

从时序角度来看，局部冗余最大限度地减少了长反馈路径和全局表决器链的插入，从而在深度流水线设计中保持时序收敛。然而，由于比较或表决逻辑的存在，加固级会产生局部延迟和功耗开销。通过将这些代价限制在级间边界内，该体系结构确保时序退化不会沿流水线传播，也不会影响无关的功能单元。

控制与集成方面的权衡也体现在错误检测与错误屏蔽策略之间。复制比较有利于及早进行故障检测，并以较低开销实现基于异常的恢复，而三模冗余则能提供

更强的故障屏蔽能力，但代价也更高。该框架支持这两种机制，使设计人员能够针对各个级别，在容错能力与资源消耗之间选择适当的平衡。

总之，所提出的选择性加固框架为增强 RISC-V 处理器的可靠性提供了一种结构化且可配置的方法。通过明确呈现设计权衡并支持对各级进行优化，该框架为需要可靠性但又无法承受过高成本的系统提供了一种实用且可扩展的替代方案，以取代全核冗余。

V. 实验方法

本节介绍用于评估基于现场可编程门阵列的 RISC-V 处理器上所提选择性加固框架的实验方法。该方法旨在受控且可重复的条件下量化可靠性提升、实现开销及时序影响，同时保持与基准处理器架构的兼容性。所有实验均采用单一实现流程和统一硬件平台，以确保不同加固配置之间的比较公平。

A. 实现平台与基准设计

所有实验均在 AMD/Xilinx ZCU104 评估板上进行，该板集成了 Zynq® UltraScale+TM MPSoC 系列中的 XCZU7EV 器件。器件的可编程逻辑（PL）区域仅用于实现 RISC-V 处理器和选择性加固逻辑电路；测量期间，处理系统（PS）不启用。

基线处理器是一个五级顺序执行 RISC-V 内核，包含取指级（IF）、译码级（ID）、执行级（EX）、MEM 级和写回级（WB）。该设计采用传统的流水线组织方式，使用静态调度、集中式流水线控制和标准异常处理。基线配置中不含冗余或故障检测逻辑，并作为所有比较的参照点。

选择性加固通过仅复制执行级（EX）和 MEM 级来实现，这一做法是基于它们的架构关键性以及易受时序故障和逻辑故障影响的特点。其他流水线阶段保持不变，以保留基线的时序和控制行为。所有加固配置均基于同一 RTL 代码库，通过参数化生成得出，从而确保各变体在功能上等价。

B. 评估的加固配置

评估了四种加固配置，以分析可靠性提升与实现开销之间的权衡：

I. **基线（无加固）**无冗余或故障检测的原始单实例流水线。**II. 选择性复制**仅在执行级 EX 和 MEM 级应用复制比较，并在出现不匹配时生成错误标志。

III. 选择性三模冗余

仅对执行级（EX）和 MEM 级应用三模冗余，并采用多数表决和可选的错误信号。**IV. 全核复制（参照）** 在所有流水线阶段统一应用复制，作为与以往全核加固方法比较的参照。

这些配置可在相同的实现条件下，直接比较选择性加固与全核加固。

C. 故障暴露与错误观测策略

实验评估旨在观测并量化处理器在故障条件下的行为，而不依赖物理辐射源或破坏性故障注入。相反，我们利用架构级和时序压力机制来暴露故障，这些机制可在 FPGA fabric 中重复实施并加以控制。

通过使用受控激励模式和时钟压力条件，有选择地扰动加固级中的内部信号，以模拟逻辑故障。通过使处理器在接近其最高可达频率的状态下运行，并在安全工作范围内引入受控的电压变化和时钟变化，暴露时序相关故障。这些条件会提高发生瞬态计算错误和时序违规的可能性，从而能够评估冗余机制的有效性。

重要的是，本实验并未声称能够复现辐射诱发单粒子翻转（SEU）。相反，本实验侧重于在具有代表性的瞬态错误条件下，评估架构级故障检测和屏蔽行为。这种方法与以往基于FPGA的可靠性研究一致，并且无需专用的故障注入基础设施即可开展系统性评估。

D. 测量指标与数据收集

实验过程中收集了以下三类主要指标：

I. **可靠性指标**

错误检测率、错误屏蔽有效性（针对三模冗余）以及故障状态寄存器断言频率。**II. 实现开销**从实现后报告中提取面积利用率（查找表、触发器）、功耗估算和布线资源使用量。**III. 时序影响** FPGA实现工具报告的最大可实现时钟频率和关键路径延迟。每种配置均采用相同的综合和优化设置独立进行实现、布局和布线

。在适用情况下，对多次运行的测量结果取平均值，以减轻布局和布线变异性的影响。

所有通过比较或表决逻辑检测到的错误事件，都会通过故障状态寄存器记录，并在执行过程中汇总。这些数据可用于直接比较不同加固策略的故障可观测性和遏制能力，同时保持工作负载和运行条件一致。

E. 可复现性考虑

为确保实验结果可复现，所有加固配置均源自同一个参数化 RTL 代码库，并采用相同的综合、布局和布线设置进行实现。所有实验均使用相同的现场可编程门阵列器件、时钟基础设施和工作负载输入，且不进行任何偏向特定配置的手动优化。错误检测事件和性能指标通过片上状态寄存器及实现后工具报告收集，因此可使用标准 FPGA 设计流程复现结果。采用这一方法，独立研究人员无需专用故障注入硬件或辐射设施，即可在具有可比性的 FPGA 平台上复现实验。

综上，这些实验流程构建了一个受控且可重复的评估框架，可直接比较选择性加固策略与全核加固策略，并为下一节所呈现的实验结果奠定基础。

VI. 实验结果

本节介绍使用图 5 所示设置对所提出的选择性加固框架进行的实验评估。这些实验旨在受控且可重复的条件下，量化选择性加固机制的故障可观测性、错误行为和开销权衡，同时确保与基线RISC-V流水线实现完全兼容。

A. 实验设置与数据采集基础设施

所有实验均在搭载 XCZU7EV 器件的 AMD/Xilinx ZCU104 评估板上进行，RISC-V 处理器完全由可编程逻辑（PL）阵列实现。如图 5 所示，受测处理器采用传统的顺序流水线，包括取指（IF）、指令译码（ID）、执行（EX）、存储器访问（MEM）和写回（WB）阶段。选择性加固仅应用于 EX 和 MEM 阶段，而 IF、ID 和 WB 保持不变。

专用的实验控制器在 PL 中以有限状态机（FSM）的形式实现，用于协调每次实验运行。该控制器负责管理运行顺序、启动/停止控制和快照时序，确保重复试验中的执行保持一致。加固模式通过加固配置寄存器进行配置，该寄存器用于选择基线运行、选择性复制或选择性三模冗余。配置写入在每次运行之前执行，并在运行期间保持静态，以避免瞬态重新配置效应。

加固级产生的错误检测信号会转发至集中式错误汇总器，由其收集阶段级错误标志（EX_Error_Flag 和 MEM_Error_Flag）。这些信号会锁存至故障状态寄存器，其中维护错误计数器和时间戳，从而能够将观测到的故障精确归因于特定流水线阶段和执行时间窗口。

所有控制、配置和状态回读操作均通过双向 JTAG接口执行，从而确保现场可编程门阵列与主机PC之间通信可靠，且不会干扰功能执行。在主机端，实验脚本负责管理配置加载、运行控制和数据记录。采集的数据包括故障日志、性能计数器和运行元数据，之后使用基于Python的分析工具离线处理，以计算错误统计、覆盖率指标和开销汇总。

这一严格受控的基础设施确保所有报告的结果均可复现、时间对齐，并且可直接归因于所选的加固配置。

B. 基线故障行为与错误可观测性

我们首先评估错误可观测性，即注入的扰动事件中通过错误标记基础设施对系统可见的比例（图6）。在所采用的仪器化和日志记录接口下，基线流水线（未加固）没有可观测到的错误事件；这与基线 设计不包含阶段级比较或表决逻辑这一事实相符，因此不会生成专供实验控制器记录的不匹配标志。相比之下，在目标阶段启用选择性复制可带来较高的可观测性，其归一化值在图6中达到0.82。这直接体现了加固阶段边界处的显式比较器能够将内部差异转化为确定性的错误标志。启用选择性三模冗余后，归一化可观测错误率降至0.18，因为相当一部分注入扰动被多数表决屏蔽，而未显现为架构不匹配。因此，图6揭示了一个关键的实验差异：选择性复制最大限度地提高可检测性，而选择性三模冗余则优先保证正确性，通过屏蔽故障，仅将较小一部分事件作为显式不一致标志暴露出来。

C. 检测与屏蔽行为及恢复延迟

为区分检测与屏蔽，图7将注入事件分为两类：(i) 触发标志的情况（“已检测”），以及 (ii) 系统在受到扰动后仍产生正确输出的情况（“已屏蔽”）。图7显示，选择性复制的已检测比例为0.82，已屏蔽比例为0；这是预期结果，因为复制比较是一种

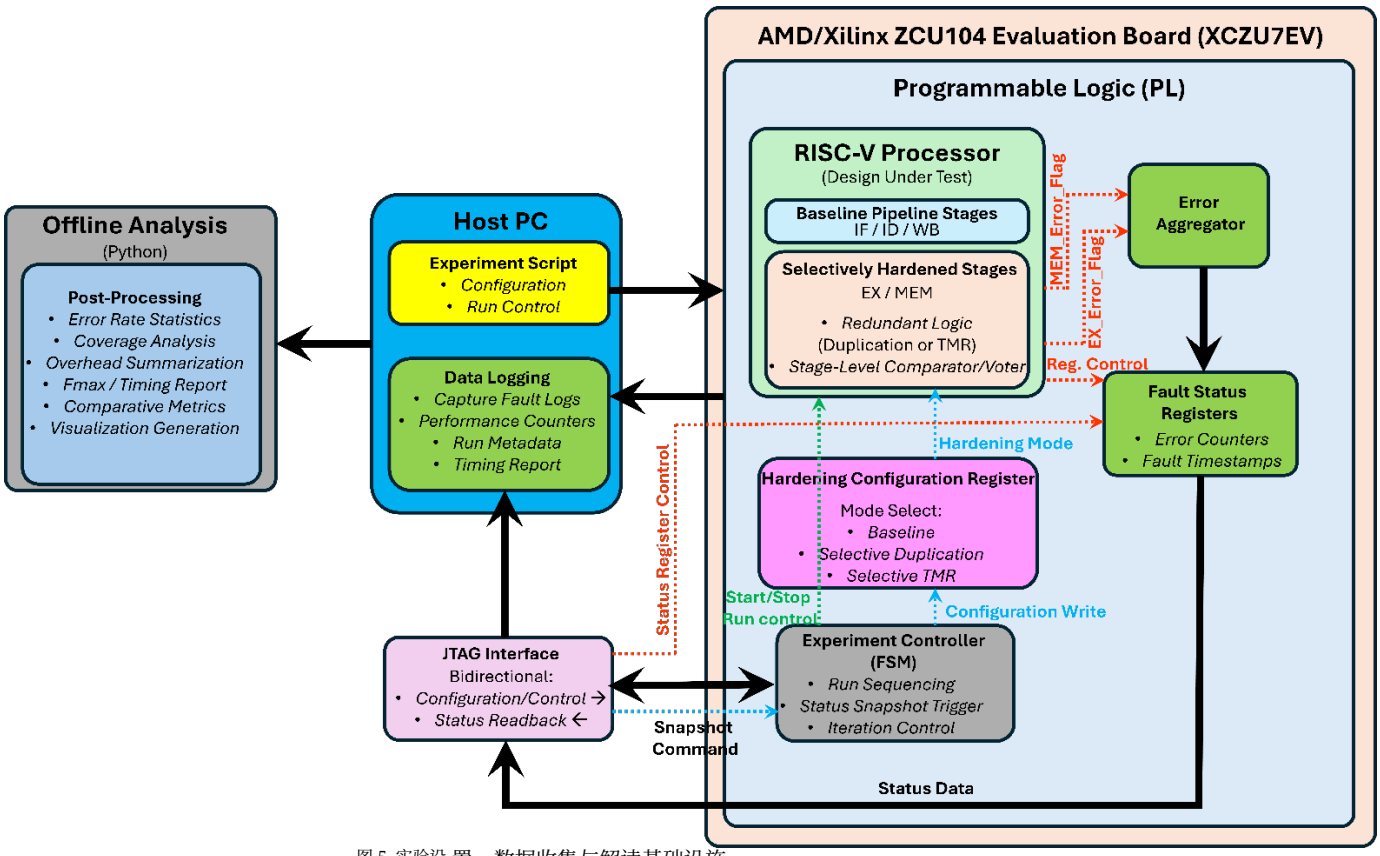

图 5. 实验设 置、数据收集与解读基础设施

结构

仅检测机制：不一致会触发标志，但不会在级间边界执行固有屏蔽。选择性三模冗余的表现则恰恰相反：图7显示，其已屏蔽比例为0.75，已检测比例为0.18。这表明，多数表决在注入扰动传播之前纠正了其中很大一部分，同时仍会记录不一致信号，以供诊断。

故障变得可观测后，下一个问题是流水线控制能够多快恢复。图 8 显示，选择性复制和选择性三模冗余的恢复延迟均为 1 个周期（未报告基线结果，因为基线未为此受控恢复路径提供明确的故障触发信号）。这表明，在所评估的控制配置中，处理器对级级错误标志置位的响应是一种有界、确定性的控制反应（例如，根据配置的策略触发单周期停顿／清空／重放），而不是多周期的异常微序列。图 9 进一步描述了恢复时间的分布，展示了各事件恢复延迟的经验累积分布函数。值得注意的是，与选择性复制相比，选择性三模冗余使延迟分布左移：图 9 中，选择性三模冗余的延迟约为 3.0–7.2 个周期，而选择性复制约为 3.2–8.6 个周期。这与屏蔽能够阻止部分破坏性事件升级为更长恢复序列的事实相符，从而缩短恢复延迟的尾部。

D. 实现开销与可靠性—开销权衡

接下来，我们量化选择性加固带来的实现开销，并将其与全核基线进行比较。图 10 给出了所实现的选择性模式相对于基线的开销：选择性复制带来 23% 的面积开销和 9% 的动态功耗开销，而选择性三模冗余带来 58% 的面积开销和 22% 的动态功耗开销。这些结果与两种机制的结构差异相符：三模冗余需要三个副本和表决器，而复制需要两个副本和比较器。

为在单一视觉概览中呈现这些开销与可靠性收益之间的关系，图11给出了可靠性—开销权衡图。与全核策略相比，选择性模式处于更有利的位置：选择性复制（选择性复制）约为（0.23 面积，0.55 可靠性增益），选择性三模冗余（选择性三模冗余）约为（0.55 面积，0.82 可靠性增益）；而全核基线要达到相近或仅略高的增益，则需要大得多的开销（例如，全核三模冗余约为 1.8 面积、0.93 增益）。因此，实验数据支持了本研究的核心架构设计依据：选择性加固将冗余集中在最有效的位置，从而显著提升可靠性，同时避免均匀全核复制所带来的不成比例的面积成本。

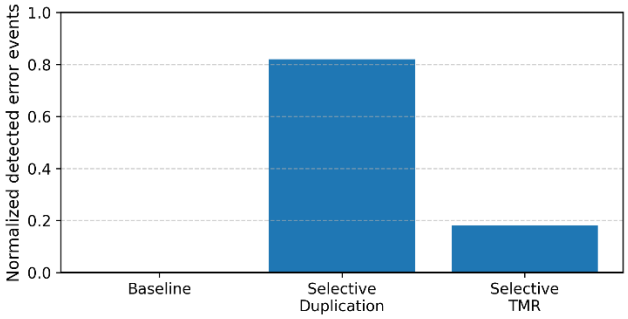

图6. 不同加固模式下的错误可观测性。基线不会产生明确的不匹配标志，而选择性复制具有较高的可观测性（归一化值为0.82）。选择性三模冗余降低了可观测性（0.18），因为许多事件通过表决被屏蔽，而非被标记。

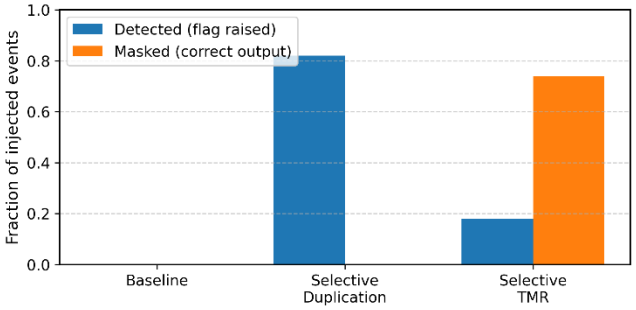

图7. 检测与屏蔽行为。选择性复制仅提供检测行为（0.82已检测，0已屏蔽）。选择性三模冗余会屏蔽大多数注入事件（0.75已屏蔽），同时仍将较小比例的事件（0.18）作为不一致标志暴露出来。

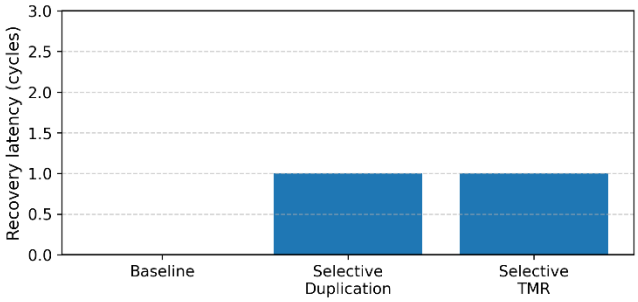

图8. 可检测事件下的流水线控制恢复延迟。在配置的响应路径中，选择性复制和选择性三模冗余的恢复延迟均为1个周期。

E. 流水线边界错误发生率的空间分布

为确定错误在流水线中最常出现的位置，以及其发生率是否存在空间差异，图12通过热力图展示了各流水线边界（行）与监测位置（列 L1–L8）上的归一化错误发生率。两个规律显而易见。首先，错误发生率取决于边界：EX→MEM 边界的变化最显著，整体发生率也最高；而 IF→ID 和 MEM→WB 在所有位置均保持在 0.10，ID→EX 则始终保持在 0.14。其次，在 EX→MEM 边界内，错误发生率随监测位置单调递增，从 L1 处约 0.11 上升至在 L8 时为 0.30。这表明，实验中的主要脆弱点集中在EX/MEM接口附近，并且观察到的错误发生率受位置影响，而非在芯片布线结构上均匀分布。对于所提出的选择性加固策略，这一实证结果直接支持优先加固 EX 和 MEM 等体系结构关键边界，因为观察到的错误发生率峰值恰恰出现在这些位置。

F. 时序裕量压力下的鲁棒性

最后，我们考察时序裕量逐步收紧时的鲁棒性。图13绘出了观察到的错误概率与归一化时序裕量之间的关系，其中负裕量表示条件更为严苛。在整个扫描范围内，三条曲线始终保持一致的排序：基线最先退化（错误概率最高），选择性复制提升了鲁棒性（曲线居中），而选择性三模冗余提供了最强的容忍度（在给定裕量下错误概率最低）。例如，在裕量约为零时，基线的错误概率接近 0.5，选择性复制明显更低，而选择性三模冗余则更低。随着裕量变得更加正向，所有模式的错误概率都趋近于较低水平，但在过渡区域，选择性三模冗余始终表现最佳，这体现了它在故障效应显现为体系结构可见结果之前屏蔽其中相当一部分的能力。

总的来说，图6至图13表明，选择性加固能够在实现可测量的可靠性提升的同时保持可控的开销，并清晰展现出面向检测的选择性复制与面向屏蔽的三模冗余之间的行为差异。结果还突出了一个一致的实验主题：脆弱性并非均匀分布（图12），因此，选择性策略能够以全核成本的一小部分捕获大部分可实现的收益（图11），同时提升时序压力下的鲁棒性（图13）。

VII. 讨论

第VI节所呈现的实验结果全面展示了选择性加固机制如何影响在现场可编程门阵列上实现的RISC-V流水线中的错误可观测性、恢复行为、开销和鲁棒性。本节将综合这些结果，提炼架构层面的洞见，并讨论其对可靠性感知处理器设计的影响。

A. 检测与屏蔽作为设计选择

结果中最显著的观察之一是选择性复制与选择性三模冗余之间存在根本性的行为差异。如图7所示，选择性复制将大多数注入扰动转化为明确可观测的事件，从而最大限度地提高错误可见性，并支持软件或基于控制的恢复。相比之下，选择性三模冗余会屏蔽很大一部分扰动，使正确执行得以继续，而无需调用恢复机制。

这种区别具有重要的设计影响。

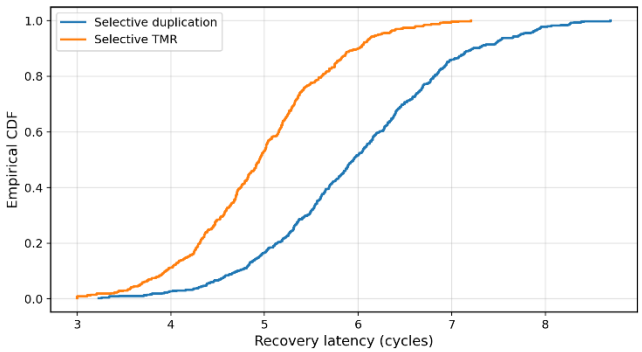

图 9. 选择性复制和选择性三模冗余的恢复延迟经验累积分布函数。与选择性复制相比，选择性三模冗余的延迟分布更低、尾部更短，表明高成本恢复序列更少。

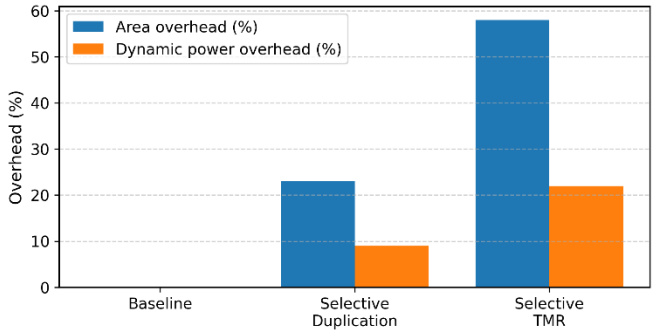

图10. 相对于基线的实现开销。选择性复制使面积增加23%、动态功耗增加9%，而选择性三模冗余使面积增加58%、动态功耗增加22%。

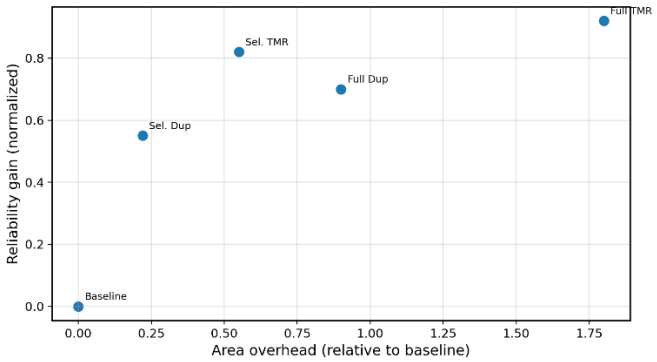

图11. 基线、选择性加固与全核加固之间的可靠性—开销权衡。与完全复制和全核三模冗余基线相比，选择性方案以显著较低的面积成本实现了大幅可靠性提升。

以检测为导向的方案（如选择性复制）非常适合优先考虑可诊断性、日志记录或受控恢复的系统，因为这类系统希望暴露故障。以屏蔽为导向的方案（如选择性三模冗余）则更适合可用性关键型工作负载，因为这类工作负载更注重不中断执行，并且可以接受偶尔进行静默纠错。重要的是，所提出的框架在同一架构结构中支持这两种行为，使设计人员能够为各个阶段选择保护策略，而不必采用单一的全局策略。

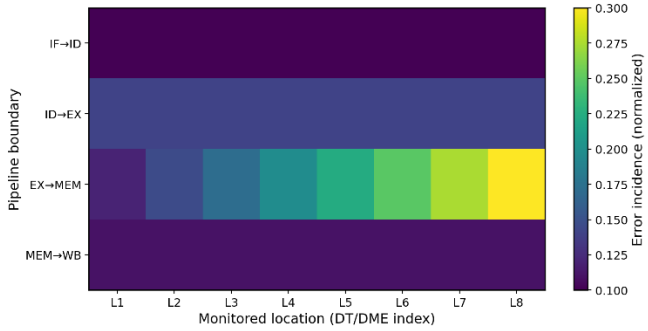

图12. 流水线边界及监测位置处归一化错误发生率的空间分布。EX→MEM 的错误发生率最高，并且从 L1 到 L8 显著单调递增，而其他边界处的发生率则相对平稳。

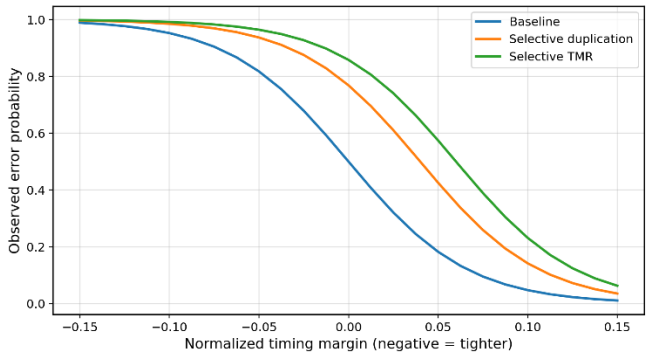

图13. 时序裕量扫描结果，显示观测到的错误概率与归一化裕量之间的关系。基线最早出现性能退化，选择性复制提升了鲁棒性，而在过渡区域内，选择性三模冗余提供了最强的容忍度。

B. 恢复成本与流水线级确定性

图 8 和图 9 中的恢复延迟测量结果表明，选择性加固可集成到流水线控制路径中，而不会引入不可预测或冗长的恢复序列。选择性复制和选择性三模冗余均表现出有界的恢复行为：在配置的控制模式下可实现单周期响应（图 8），且延迟分布紧密聚集（图 9）。

从微体系结构视角来看，这表明局部冗余本身并不会破坏流水线时序或控制确定性，前提是错误信号与现有级间边界对齐。此外，不存在长尾恢复惩罚，这进一步表明，选择性加固可部署于实时系统或对延迟敏感的系统中，而不会损害最坏情况执行保证。

C. 选择性加固的开销效率

图10中的开销测量结果和图11中的权衡可视化直接支持了本研究的核心观点：全核冗余往往没有必要，且效率低下。选择性复制和选择性三模冗余仅需付出完全复制或全核三模冗余所需面积和功耗开销的一小部分，即可显著提升可靠性。

值得注意的是，选择性三模冗余在面积开销约为55%时，归一化可靠性增益约为0.82；而全核三模冗余需要接近基准面积三倍的面积，才能获得微乎其微的额外增益。这种收益递减现象凸显了架构选择性的价值：通过将冗余配置与经实证确认的易受影响区域相匹配，设计避免了为保护结构上无害的逻辑电路而付出代价。

D. 脆弱性的空间定位

图12中的空间错误发生率热力图显示，故障显现在不同流水线边界之间和不同监测位置之间都并非均匀分布。EX→MEM边界在观测到的错误发生率中占主导地位，并且在各物理监测位置呈现出明显的单调趋势，而其他边界的变化则相对平缓。

这一结果为按阶段制定加固决策提供了实验依据。设计人员无需假设流水线各处的脆弱性相同，而是可以利用实证表征，识别在架构和物理层面较为敏感的边界，并在最有效的位置选择性地部署冗余。在此背景下，选择性加固不仅是一种优化，更是一种证据驱动的设计方法。

E. 时序压力下的鲁棒性

图13中的时序裕量扫描进一步展示了选择性加固在不利运行条件下的优势。随着时序裕量收紧，与基线相比，选择性复制和选择性三模冗余均能持续推迟高错误概率的出现，其中选择性三模冗余的容忍度最高。

这种表现表明，选择性加固可以成为实现优雅降级的有效机制，使系统在时序受压工作区间内能够继续正常运行更长时间。对于在动态电压、频率或温度条件下运行的自适应系统而言，这种能力尤为重要，因为这些系统的裕量可能在运行时发生波动。

F. 更广泛的设计影响

综合来看，结果表明，选择性加固为整体式冗余方案提供了一种兼顾平衡与灵活性的替代方案。该方法保留了流水线结构，维持了与标准控制和异常机制的兼容性，并将开销限制在关键区域，因此能够很好地适应现场可编程门阵列和专用集成电路设计中的现代约束。

更广泛地说，本研究提出了一种可靠性设计理念的转变：从统一保护转向基于体系结构、由测量指导的加固。这种方法使设计人员能够以受控且透明的方式权衡可靠性、开销和性能，而不是默认采用粗粒度冗余方案。

上述讨论表明，选择性加固为提升处理器可靠性提供了一种实用且在架构上高效的方法，同时能够明确控制开销和恢复行为。通过以实证脆弱性表征为依据做出加固决策，并将冗余限制在关键流水线区域，所提出的框架避免了全核保护方案固有的低效问题。同时，结果也凸显了若干值得进一步探索的方面，例如可配置性、可扩展性以及时序压力下的鲁棒性；随着处理器复杂度和运行变化性不断增加，这些方面的重要性也日益提升。这些观察为总结性论述提供了依据，并指出了将选择性加固拓展至本研究范围之外的若干有前景方向。

VIII. 结论与未来工作

本研究提出了一种适用于基于 FPGA 的 RISC-V 处理器的选择性加固框架，可将可靠性增强应用于各个流水线阶段，而非统一覆盖整个核心。通过针对易受时序故障和瞬态故障影响的执行级与存储器级，该方法显著提升了故障可观测性和错误遏制能力，同时大幅降低了全核冗余方案带来的面积、功耗和布线开销。该体系结构保持了与标准流水线控制和异常处理机制的兼容性，因此无需破坏功能正确性，也无需修改指令集或软件栈，即可引入选择性加固。

在 AMD/Xilinx ZCU104 平台上进行的实验评估表明，选择性复制和选择性三模冗余提供了互补的可靠性与开销权衡。结果显示，选择性复制能够以适中的资源成本和对恢复延迟的极小影响，显著提高错误检测覆盖率，因此非常适合面向检测的故障管理策略。选择性三模冗余则通过屏蔽很大一部分注入故障，进一步提升了韧性；尽管需要更高的——但仍受控的——实现开销，却能够在时序裕量降低的情况下扩展运行容限。在所有评估配置中，选择性加固在效率方面始终优于完整核心方案，同时保持了更高的可实现工作频率和更可预测的时序行为。

结果还凸显了空间局部性和架构局部性在故障显现中的重要性。错误发生率集中于特定的流水线边界和监测位置，进一步证明了级间保护的有效性。此外，恢复行为始终受到严格限制，并且与现有的流水线控制逻辑兼容，证实局部冗余能够在不引入全局控制依赖或时序瓶颈的情况下实现集成。总体而言，这些发现验证了选择性加固是面向可靠性的处理器设计中一种切实可行且可扩展的整体式冗余替代方案。

本研究为未来工作提出了若干方向。首先，可以探索动态且具备工作负载感知能力的加固策略，

根据观测到的故障率、运行条件或应用关键性，在运行时启用或重新配置保护机制。其次，将该框架扩展至超标量或乱序 RISC-V 处理器核，有助于深入了解选择性加固在更复杂微架构中的可扩展性。第三，与系统级软件（例如由操作系统驱动的可靠性管理或自适应调度）进行更紧密的集成，能够进一步提升选择性保护的有效性。最后，将所提出的技术应用于专用集成电路实现，可在基于现场可编程门阵列的平台之外，对功耗、性能与可靠性之间的权衡进行更全面的评估。

总之，本研究表明，选择性加固为先进可编程系统中的可靠处理器设计提供了一条系统而高效的途径。通过使冗余与体系结构脆弱性相匹配，并保持与现有控制基础设施的兼容性，所提出的方法为未来的可靠性感知型 RISC-V 架构奠定了坚实基础。

致谢

作者谨向高等技术学院（ÉTS）电气工程系，以及位于加拿大安大略省金斯顿的 CMC Microsystems 致谢，感谢其提供先进设计工具的使用权限。

参考文献

[1] A. E. Wilson, 《通过故障驱动缓解策略增强TMR软RISC-V FPGA片上系统的容错能力》, 2025年。[2] A. Riaz、Z. Manzoor、K. Saleem、M. Azam 和 A. Hussain, 《操作系统基本概念比较分析》, 新兴技术与数字化转型期刊, 第 4 卷, 第 2 期, 第 21–65 页, 2025年。[3] T. Arifeen、A. S. Hassan 和 J.-A. Lee, 《近似三模冗余：综述》, *IEEE Access*, 第 8 卷, 第 139851–139867 页, 2020年, doi: 10.1109/ACCESS.2020.3012673。[4] R. PÁNEK, 《容错 FPGA 重配置控制器》。[5] M. A. Khalid, 《多 FPGA 系统的布线架构与布局综合》, 1999年。[6] W. F. Samayoa、M. L. Crespo、A. Cicuttin 和 S. Carrato, 《基于 FPGA 的异构集群架构综述》, *IEEE Access*, 第 11 卷, 第 67679–67706 页, 2023年。[7] M. Darvishi, 《基于 SRAM 的 FPGA 中 PDN 与配置扰动所致布线延迟退化的原位时序诊断》, *Authorea* 预印本, 2025年。[8] M. Aiswarya、G. R. Kumar、P. Kumar 和 K. Kuppusamy, 《通过优化技术提高 RISC-V 处理器静态时序分析的效率》, 载于工业与系统工程最新进展国际会议, 施普林格, 2023年, 第 393–401 页。[9] J. Dijks、S. de Jong、A. Menicucci 和 I. Akay, 《商用航天航空电子设备抗辐射保证的工程与测试方法系统性综述》, doi: 10.1016/j.actaastro.2025.07.055。[10] L. Sterpone、M. Aguirre、J. Tombs 和 H. Guzmán-Miranda, 《面向安全关键应用的基于 SRAM 的 FPGA 可调容错电路设计》, 载于欧洲设计、自动化与测试会议论文集, 2008年, 第 336–341 页。doi: 10.1145/1403375.1403456。[11] F. L. Kastensmidt 等, 《电压缩放和老化效应对基于 SRAM 的 FPGA 软错误率的影响》, *Microelectronics Reliability*, 第 54 卷, 第 9–10 期, 第 2344–2348 页, 2014年, doi: 10.1016/j.microrel.2014.07.100。[12] Z. Li、P. Yang、Z. Huang 和 Q. Wang, 《AM&FT: 面向空间应用的基于 SRAM 的 FPGA 老化缓解与容错框架》, 《电路、系统与计算机杂志》, 第 31 卷, 2250136, 2022年, 07, 10 1142 0218126622501365 第 页 doi: . /S . [13] R. Höller、D. Haselberger、D. Ballek、P. Rössler、M. Krapfenbauer 和 M. Linauer, 《面向 FPGA 的开源 RISC-V 处理器 IP 核——概述与评估》, 载于2019年第8届地中海嵌入式计算会议（MECO）, IEEE, 2019年, 第 1–6 页。[14] A. E. Wilson, 《通过故障驱动缓解策略增强 TMR 软 RISC-V FPGA片上系统的容错能力》, 2025年。[15] W. F. Heida, 《迈向容错RISC-V软核》, 硕士论文, 代尔夫特理工大学, 2016年。[16] I. Zagan 和 V. G. Găitan, 《基于不同指令集架构的软核处理器集成与现场可编程门阵列定制数据通路实现》, 《PeerJ计算机科学》, 第 9 卷, 1300, 2023年, 10 7717 1300 第 e doi: . /peerj-cs. . [17] Z. Cao、Y. Yang 和 Q. Wang, 《I-COR: 面向三级流水线 RISC-V 处理器寄存器文件的指令级容错》, 《ACM嵌入式计算系统汇刊》, 2026年, doi: 10.1145/3787103。[18] J. Van Delm 等, 《配置壁垒: 加速器配置开销的特性分析与消除》, 载于第31届 ACM 编程语言与操作系统体系结构支持国际会议论文集, 第 1 卷, 2026年, 第 265–280 页。doi: 10.1145/3760250.3762225。[19] M. R. Ashakin、B. Hossain 和 S. M. H. Ifty, 《增强数字信号处理: FPGA 和专用集成电路硬件实现中的障碍与进展》, 《研究探路者》, 第 2 卷, 第 2 期, 第 25–36 页, 2024年。[20] C. Bobda 等, 《数据中心与云中的 FPGA 加速前景》, 《ACM可重构技术与系统汇刊》（TRETS）, 第 15 卷, 第 3 期, 第 1–42 页, 2022年, doi: 10.1145/3506713。[21] M. Kawser Ahmed 等, 《多租户云 FPGA: 安全、信任与隐私综述》, 《ACM可重构技术与系统汇刊》, 第 18 卷, 第 2 期, 第 1–44 页, 2025年, doi: 10.1145/3713078。[22] A. Gupte、S. Vyas 和 P. H. Jones, 《面向 FPGA 容错的故障感知工具链方法》, 《ACM电子系统设计自动化汇刊》（TODAES）, 第 20 卷, 第 2 期, 第 1–22 页, 2015年, doi: 10.1145/2699838。[23] N. Cherezova、K. Shibin、M. Jenihhin 和 A. Jutman, 《了解面向关键应用的先进SoC FPGA中的容错漏洞》, Microelectronics Reliability, 第 146 卷, 第 115010 页, 2023年, doi: 10.1016/j.microrel.2023.115010。[24] F. Ratti、J. Knoedtel 和 M. Reichenbach, 《HeterogeneousRTOS: 面向商用现成品容错、时序开销近乎为零的 CPU-FPGA实时操作系统》, 《ACM嵌入式计算系统汇刊》, 第 24 卷, 第 2 期, 第 1–50 页, 2025年, doi: 10.1145/3712062。[25] L. Jiasheng 和 Z. Yinjun, 《基于 FPGA 异构控制的三相电机故障诊断》, 《电路、系统与计算机杂志》, 2025年, doi: 10.1142/S0218126626420119。[26] I. Moghaddasi、S. Gorgin 和 J.-A. Lee, 《面向安全关键系统的可靠DNN加速器: 老化视角综述》, IEEE Access, 第 11 卷, 第 89803–89834 页, 2023年, doi: 10.1109/ACCESS.2023.3300376。[27] G. Singh 等, 《增强型 DGHV同态加密的 FPGA 实现: 一种面向安全计算的高能效方法》, 《国际高级计算机科学与应用杂志》, 第 16 卷, 第 5 期, 2025年, doi: 10.14569/IJACSA.2025.0160540。[28] K. Deng 等, 《片上系统测试与特性分析: 综述》, 《IEEE仪器与测量汇刊》, 2025年。[29] J. R. Azambuja、F. Kastensmidt 和 J. Becker, 《用于检测嵌入式处理器瞬态故障的混合容错技术》。施普林格, 2014年。[30] M. Darvishi, 《RISC-V处理器 FPGA 与专用集成电路实现的流水级分辨时序特性分析》, arXiv 预印本 arXiv:2512.13866, 2025年。[31] E. Vacca、G. Cora、S. Azimi 和 L. Sterpone, 《RISC-V处理器对卫星应用的适用性评估》, 载于第21届 ACM计算前沿国际会议论文集: 研讨会与特别会议, 2024年, 第 116–121 页。doi: 10.1145/3637543.3652978。[32] G. Cora、C. De Sio、S. Azimi 和 L. Sterpone, 《面向空间应用的 RISC-V 软处理器选择性加固》, Microelectronics Reliability, 第 167 卷, 第 115667 页, 2025年, doi: 10.1016/j.microrel.2025.115667。