# Strategic planning for a Scientific Article on Dual-Core Glitch Protection for a Niche Audience

Jarl Magnus Sæbø

October 2023

Today micro-controllers and other embedded electronics are in almost everything we use. Because of this, ensuring the secure operation of these devices is critical. While much attention is directed towards software and firmware security, hardware security often goes overlooked. Particularly, hardware fault-injection (glitching) har repeatedly demonstrated its capability to bypass conventional security measures. Examples include bypassing cryptographic signature validation of firmware binaries [Boo20], re-enabling hardware debug functionality on production/fused processors[Lim20] and bypassing input validation of data that crosses trust boundaries between privilege levels[Rae21]. To stop these glitch attacks, a number of countermeasures can be introduced. This project looks at the ones implemented by the OpenHW Group for their 'CV32E40S' RISC-V core[Gro20], and how these security measures can be avoided entirely by instead running a dual-core RISC-V setup with lockstep.

The RISC-V architecture has emerged as a significant point of discussion within the realm of computer engineering, particularly in the context of embedded systems[SNH20]. Interestingly, despite the relevance of glitch protection, it only started gaining attention after practical glitch attacks first made their appearance in 2002 through an optical fault attack[TBC19]. In the world of computer architecture, many aspects remain closely guarded secrets. This secrecy extends to preventive measures against such attacks, making them often inaccessible to the public. As a result, the primary sources of information about preventing these types of attacks are other research papers, rather than from the actual producers of the products. Given the nature of the topic, the primary audience for this project comprises:

- **Embedded System Engineers:** Professionals and researchers who specialize in designing, testing, and implementing embedded systems will find this project directly relevant. The focus on glitch protection using a dual-core approach can offer insights into designing more robust systems.

- **Computer Architecture Enthusiasts:** The project delves into the intricacies of the RISC-V architecture, making it appealing to those who are keen on understanding the nuances and advancements in computer architecture.

However, it's important to delineate those within the field of computer engineering who might not find this project as relevant:

- **Low-level Hardware Engineers:** Professionals who focus on transistor-level design, analog circuits, and foundational logic gate designs might find the content too high-level, as the project emphasizes system-level solutions rather than micro-level components.
- **Software Developers:** While computer engineering as a broad field encompasses both hardware and software realms, this project leans heavily towards hardware considerations. Hence, software engineers, especially those who aren't involved in firmware or low-level programming, might not find the content directly applicable.

In summary, while the project is of large importance to a subset of computer engineers and researchers, its specialized nature may limit its appeal to those beyond the realms of embedded systems and computer architecture. Nonetheless, the primary aim of this project is to contribute to the existing body of knowledge in the area of glitch protection. Despite its growing importance, glitch protection remains a mystified and under-explored domain within computer engineering. Through my research, I aspire to shed light on this niche area, offering insights, methodologies, and perhaps innovative solutions that can be useful for both academia and industry.

In this research paper, I will undergo a comprehensive examination of current glitch detection methods—highlighting both their strengths and weaknesses. Leveraging an available open-source RISC-V processor with several built in security measures, my aim is to shed light on the inherent trade-offs of existing systems: they function but often at the expense of increased power consumption, system complexity and delayed execution of programs. Moreover, the current techniques often fall short in pinpointing the exact nature and origin of faults.

My core argument is straightforward: there's a viable alternative in the form of a dual-core layout. I wish to argue that this proposed system not only matches but potentially surpasses the efficiency of current solutions, offering ease of implementation as a major benefit, as well as the possibility of giving more precise feedback about the nature of detected faults. To substantiate my claims, this paper will present rigorous testing and simulations, ensuring that the reader walks away with a robust understanding and trust in the proposed solution's viability. In order to portray my claims in a good way I intend on using a lot of graphs and tables to display data. For instance, when comparing the power usage, area and performance of both solutions, good visuals are necessary.

In order to effectively communicate my message to the audience I propose the following structure of my text:

- Abstract: Provide a brief summary of the entire paper that encapsulates the main objectives, methods and findings.
- Introduction

- Problem Definition
  - Statement of the problem
  - Scope and boundaries of the problem
  - Significance: Why is this problem worth solving?
  - Existing solutions and their limitations
  - Relevance to the target audience
  - Objective statement
- Background
  - Introduction to RISC-V
  - The need for glitch protection: Showcase how glitches have been used previously to bypass security measures in embedded systems.
  - State of the art / Existing research: Current technologies and comparative analysis.
  - Previous approaches to glitch protection and their limitations
  - Proposed Dual-Core Solution
  - Rationale for this study: Convince reader that this research is necessary and will contribute positively to the field.
- Methodology and Experimentation
  - Methodology and Theory: How do I want to compare the performance of the existing solutions and the proposed dual core solution? (Power, Area and Performance)
  - Experimental Setup: What software is used for simulation and synthesis
  - Limitations
- Results and Analysis
- Conclusion or Concluding Remarks

Given the specialized nature of my topic, the intended audience, primarily researchers and engineers, will have an inherent familiarity with the subject. Because of this I do not think that the use of metaphors are strictly needed. However, instead I will try to use technical analogies relevant to the field, as these can aid in clarifying complex concepts or introducing innovative ideas. For instance, the idea of using dual cores to perform glitch protection can be compared to a car's dual braking system: the primary brakes serve everyday stopping needs, but in case they fail, there is a handbrake as a safety redundancy. In addition to this I will also need to use graphical representations to illustrate concepts, especially if they're complex. Perhaps most importantly, I will need to effectively do comparative analysis where I will compare a new concept against existing ones to emphasise the benefits of my research. In order to do all these things I will use the following resources:

- **Writing Scientific Research Articles: Strategy and Steps (Cargill and O'Connor)** [CO09]: This book holds a lot of information about how to structure a scientific research article. Specifically it describes the *AIMRaD* structure which I plan to base my article on. This structure focuses on having a broad introduction to catch the interest of the reader before narrowing the scope of the article to talk about the specific problem we aim to solve. After describing the method and results, the discussion at the end will again broaden the scope to show how my work fits into the "bigger picture".

- **They Say / I Say: The Moves That Matter in Academic Writing (Graff and Birkenstein)**[Gra18]: This book "Shows that writing well means entering a conversation". This is an important thing to remember in my report as I am going to be making arguments as to why the methods that have been used earlier by other engineers are not necessarily the best, and that what I propose could be an improvement. To do this it will be important to use the "They say / I say" method that this book introduces.

- **Writing for Scholars: A Practical Guide to Making Sense & Being Heard (Nygaard)**[Nyg15]: This book explains how to "reach other scholars and convince them that you have something important to say". This book will be important for all communicative aspects of the reports as it describes how to make precise figures, form the core argument and structure the paper as well as much more.

As glitches and their associated threats become more pronounced, the need for innovative and robust protection mechanisms becomes increasingly important. The research proposed in this paper seeks to address an existing gap in the defences that are currently available to the public. By drawing on technical analogies, graphical illustrations and comprehensive comparative analysis, this paper aims to shed light on the possible benefits of the proposed solution. The main takeaway for engineers and researchers reading this paper is clear: as glitch attacks continue to evolve, so too must our strategies for combating them. The dual core proposition is a step in that direction.

## References

[Boo20] Jeremy Boone. There's a hole in your soc: Glitching the mediatek bootrom, 2020. Accessed: Sep 25, 2023.

[CO09] M. Cargill and P O'Connor. Writing scientific research articles: Strategy and steps. *Oxford UK: Wiley-Blackwell*, 2nd Ed:11–16, 2009.

[Gra18] Birkenstein C Graff, G. They say / i say: The moves that matter in academic writing. 4th Edition, 2018.

[Gro20] OpenHW Group. *OpenHW Group CV32E40S User Manual*, 2020. Accessed: Sep 25, 2023.

[Lim20] LimitedResults. nrf52 debug resurrection (approtect bypass) part 1, 2020. Accessed: Sep 25, 2023.

[Nyg15] P Nygaard, Lynn. Writing for scholars: A practical guide to making sense  being heard. 2nd Edition, 2015.

[Rae21] Raelize. Qualcomm ipq40xx: Breaking into qsee using fault injection, 2021. Accessed: Sep 25, 2023.

[SNH20] R. Ueno S. Nashimoto1, D. Suzuki and N Homma. Bypassing isolated execution on risc-v with fault injection, 2020.

[TBC19] T. Trouchkine, G. Bouffard, and J. Clediere. Fault injection characterization on modern cpus – from the isa to the micro-architecture. In *WISTP 2019*, Paris, France, 2019.