

Block-SSD: A New Block-Based Blocking SSD Architecture

Ryan Wong^{†††} Arjun Tyagi[†] Sungjun Cho[†] Pratik Sampat[†] Yiqiu Sun[†]

[†]Arcana Spirit Squad ^{†††}Arcana High Leadership Council

1. Introduction

Computer science and related fields (e.g., computer engineering, computer hardware engineering, electrical engineering, electrical and computer engineering, computer systems engineering) often draw inspiration from other fields, areas, and the real world in order to describe topics in their area. One cross-domain example is the idea of a **block**. The idea of blocks comes in many flavors, including software (e.g., process control blocks, file system blocks, data blocks, basic blocks, blocking statements, blocking processes, blocker bugs) and hardware (e.g., NAND flash blocks, cache blocks, logic blocks); however, this makes it difficult to *precisely* discern what a “block” is. In this work, we make little (*negative*) effort to disambiguate these terms and propose our own set of overloaded terms to increase the complexity of this paper.

To inspire new students to join their research groups, professors often hang posters or other publications along the walls adjacent to their offices. Regrettably, Saugata does not have any posters directly covering his door, leaving prime real-estate.¹ Therefore, this underutilized space in the Siebel Center for Computer Science offers substantial opportunities for renovations. To alleviate this concern, we propose Block-SSD. Block-SSD takes a basic block, formed out of a page, and physically combines these blocks into larger blocks. Those blocks are then formed into a larger door block, which cover most of the professor’s (e.g., Saugata Ghose [3]) door. To our knowledge, we are the first to design a *block-based blocking Sabotaging Saugata’s Door* (SSD) architecture.

2. Background

Blocks. Blocks are sometimes but not always formed out of smaller units. For example, a NAND flash block is made out of smaller units called *pages*. However, this is not always the case, as operating systems often use pages as virtual blocks. To make more confusing, a cache block is often a subset of a total data page. Luckily, there are no paging statements (as opposed to blocking statements in Verilog) because what would they even do?²

For simplicity, and no generality, our work uses a hierarchy of blocking, which mirrors (but still blocks) the NAND flash block paradigm. Our most *basic block* is formed out of a basic *page* (page for simplicity). We connect these horizontally to form *blocker lines*; similarly, along the vertical dimension, we call them *blocking lines*. To distinguish Block-SSD from prior work, rather than name the larger block a *superblock*, we instead use the term, *door block*.

SSD Architecture. Prior work [1] implemented an SSD architecture in which the students **kindly** began renovations on Saugata’s door. This was a *complete misunderstanding*TM, as renovations were not set to begin.³ Unfortunately, although this work successfully implemented a blocking SSD, it was not a block-based blocking SSD. Therefore, our work SSD differs, in that we are the first block-based blocking SSD.

¹The average price of land in Urbana, IL is roughly \$4.3/sq. ft.

²Paging statements do exist in limited applications (e.g., Databases)

³Saugata was still invoiced for the cost; the balance remains outstanding.

3. Key-Idea: A Blocking SSD for Blocking Entry

When designing an SSD architecture, we consider three main metrics. First, **humor**: notably, will Saugata laugh or (threaten to) fire everyone involved.⁴ Second, **cost**: Is the absolute dollar amount minimally intrusive to Ryan’s funding. Third, **visibility**: ‘nuff said. We believe that Block-SSD satisfies all three constraints, as it is funny (**Laugh**), relatively inexpensive (< 0.1% of total operating budget), and is visible to anyone who walks by the office in Siebel 4120 (and more ☺).

4. Methodology & Results

We do not elect to use MQSim [4], a widely-used SSD simulator, as Saugata aided in its development, presenting a conflict of interest. Thus, we forgo simulation and directly fabricate an 88 cm×211 cm door block. We fold 280±20 pages of 8.5”×11” paper into basic blocks using a previously proposed technique on YouTube [5]. Block-SSD includes a convenient 20 cm×27 cm non-blocking space for door handles.

Results. We first verify the integrity of our idea by designing two non-blocking towers. Figure 1a shows a non-blocking 5×5 non-blocking tower, designed, implemented, and verified by Yiqiu. Similarly, Figure 1b shows a non-blocking 9×9 non-blocking tower produced by Arjun. From the figures, we make one surprising observation: *although they are non-blocking towers, they can not be seen through*. Therefore, they may actually be considered blocking towers...even though they fall short of a true blocking SSD architecture. End result [2].



Figure 1: Work-in-progress non-blocking towers

5. Conclusion

In this work, we present Block-SSD, the April 1, 2024, SSD architecture. Based on our analysis, we appreciate our advisor and believe the feeling is mutual.⁵ We hope that this line of work inspires future students (**WITH THEIR ADVISOR’S (rough) APPROVAL**) in new blocking architectures.⁶

References

- [1] S. Ghose, 2023. [Online]. Available: <https://x.com/SaugataGhose/status/1643304291338182678?s=20>
- [2] S. Ghose, 2024. [Online]. Available: <https://x.com/SaugataGhose/status/1774859756811043059?s=20>
- [3] S. Ghose, 2023. [Online]. Available: <https://ghose.cs.illinois.edu/>
- [4] A. Tavakkol *et al.*, “MQSim: A Framework for Enabling Realistic Studies of Modern Multi-Queue SSD Devices,” in *FAST*, 2018.
- [5] O. Twist, 2019. [Online]. Available: <https://youtu.be/5dj306CmgaM?si=ix3LNlWeIVn5Ke3t>

⁴Submissions are usually 0-way blind

⁵Ryan is the exception. Interpretation left as an exercise for the reader.

⁶The ARCANA group holds no responsibility for the firing of any student.