Area Comparison of CHERIoT and PMP in Ibex

Samuel Riedel Marno van der Maas John Thomson Andreas Kurth Pirmin Vogel

lowRISC C.I.C.
Cambridge, United Kingdom
info@lowrisc.org

arXiv:2505.08541v1 [cs.AR] 13 May 2025

***Abstract*—Memory safety is a critical concern for modern embedded systems, particularly in security-sensitive applications. This paper explores the area impact of adding memory safety extensions to the Ibex RISC-V core, focusing on physical memory protection (PMP) and Capability Hardware Extension to RISC-V for Internet of Things (CHERIoT). We synthesise the extended Ibex® cores using a commercial tool targeting the open FreePDK45 process and provide a detailed area breakdown and discussion of the results.**

The PMP configuration we consider is one with 16 PMP regions. We find that the extensions increase the core size by 24 thousand gate-equivalent (kGE) for PMP and 33 kGE for CHERIoT. The increase is mainly due to the additional state required to store information about protected memory. While this increase amounts to 42% for PMP and 57% for CHERIoT in Ibex's area, its effect on the overall system is minimal. In a complete system-on-chip (SoC), like the secure microcontroller OpenTitan Earl Grey, where the core represents only a fraction of the total area, the estimated system-wide overhead is 0.6% for PMP and 1% for CHERIoT. Given the security benefits these extensions provide, the area trade-off is justified, making Ibex a compelling choice for secure embedded applications.

***Index Terms*—Memory safety, CHERI, PMP, Trusted computing, RISC-V, Application specific integrated circuits.**

I. Memory Safety in Ibex-based Processors

Ibex is a compact, efficient, and open-source RISC-V core developed by lowRISC, specifically designed for low-power and embedded applications [1], [2]. It supports the RV32IMCB [3] instruction set and features a configurable two-stage or three-stage pipeline, making it suitable for constrained environments such as microcontrollers and security-focused processors. Ibex is best known as the core of OpenTitan, a security system-on-chip (SoC) platform equipped with a wide range of security and I/O peripherals, and the world's first commercial-grade open-source silicon root of trust.

In security domains, memory safety vulnerabilities are a major source of software security issues. Reports from both Microsoft [4] and Google [5] have shown that approximately 70% of security-related bug fixes in Windows and Chrome stem from memory safety errors. These issues are not limited to large-scale software systems; embedded systems are also highly susceptible to memory-safety vulnerabilities. Exploiting these vulnerabilities allows an attacker to compromise the confidentiality, integrity, and authenticity of data stored and processed by the system.

To enhance memory safety, Ibex supports two distinct architectural extensions. The Ibex core maintained by lowRISC [1] supports RISC-V's physical memory protection (PMP) as well as the PMP Enhancements for memory access and execution prevention on Machine mode (Smepmp), also known as enhanced PMP (ePMP). In a parallel, alternative approach, Microsoft extended Ibex with Capability Hardware Enhanced RISC Instructions (CHERI) for embedded devices in their CHERIoT-Ibex [6] implementation.

In this paper, we examine and compare the area impact of these two extensions on the Ibex core as well as a complete SoC like the secure microcontroller OpenTitan Earl Grey.

A. Physical Memory Protection (PMP)

PMP enhances memory access control and execution prevention by defining access rights for a configurable number of memory regions. These permissions are enforced according to privilege levels, helping to enforce memory isolation and to prevent unauthorised access.

RISC-V's PMP specification [7] enforces rules on all privilege modes. If at least one PMP entry is configured, all privilege modes except for machine mode are denied access to regions that do not have a corresponding rule. This, however, does not allow PMP regions to apply to machine mode without applying to all less privileged modes as well. For example, PMP cannot restrict machine mode access to memory that should only be accessible in user mode.

To provide a more flexible way to secure memory regions from machine mode, RISC-V specifies the *Smepmp* extension. This enhancement enables more flexible and expressive access control rules by providing mechanisms to restrict access even in machine mode based on PMP configuration. With Smepmp, developers can specify memory regions that are accessible to user-mode or supervisor-mode code but explicitly protected from access by machine-mode software. Smepmp helps mitigate attack vectors where attackers attempt to trick high-privileged processes into accessing or executing tampered memory from lower-privileged processes.

PMP and Smepmp are relatively simple to implement in hardware and set up in software, making them attractive security features. However, their fixed and coarse-grained number of regions can be a limiting factor, as RISC-V's specification permits at most 64 PMP regions, and misconfigurations can introduce vulnerabilities. While these mechanisms are straightforward to implement in simple cases, using PMP regions effectively in software becomes challenging when managing complex access control policies or supporting fine-grained sharing between tasks. Neither mechanism is intended to provide language-level memory safety.

B. CHERI and CHERIoT

Capability Hardware Enhanced RISC Instructions (CHERI) [8], originally developed by the University of Cambridge and SRI International, takes a different approach to memory safety by extending each memory pointer into a *capability*. Instead of accessing memory directly through addresses, memory operations must go through these capabilities—which not only store a memory address but also enforce strict limits on which parts of memory can be accessed and how. CHERI's design ensures that these capabilities cannot be modified in unsafe ways, and it includes features for securely compartmentalising different parts of a program, making software more resistant to attacks. This enables fine-grained control over memory access and significantly improves security.

Although the initial CHERI work focused on memory safety for application-class cores, embedded applications often require tailored solutions for microcontrollers. To address this, Microsoft developed *Capability Hardware Extension to RISC-V for Internet of Things (CHERIoT)* [9], [10], successfully bringing the benefits of CHERI to embedded devices without significantly increasing complexity or resource demands. On top of that, CHERIoT provides the primitives to create strong compartmentalisation models without the burden of backwards compatibility with application-class software, as well as efficient temporal safety enabled by simpler core designs. This makes CHERIoT a compelling choice for embedded systems that need both efficiency and strong security guarantees.

In contrast to the more coarse-grained isolation mechanisms provided by RISC-V's PMP, which defines memory access permissions over a limited number of fixed regions, CHERIoT enforces fine-grained restrictions specific to each individual memory access instead of general restrictions that apply to all memory accesses. PMP can provide segmentation and isolation but is limited by the number of regions it has. CHERIoT allows C and C++ code to be fully memory safe in a way that would be infeasible in a PMP setup with limited regions. This memory safety includes deterministically guaranteeing both spatial and temporal safety. This results in a more scalable solution for memory safety and fine-grained compartmentalisation, albeit at the cost of additional metadata in memory.

II. Area Comparison of Extensions

A. Evaluation Methodology

To evaluate the area overhead of different extensions designed to mitigate memory safety issues, we synthesised Ibex using a commercial tool targeting the open-source FreePDK45 technology [11]. In our synthesis, we included only the processor and its instruction cache logic, excluding data memories and any surrounding SoC infrastructure. We applied reasonable timing constraints to the input and output ports and used timing models from OpenRAM [12] for all SRAM macros. However, we did not include the SRAM macro of the instruction cache in our analysis, as it is highly technology-dependent and independent of the evaluated extensions. Therefore, we report only the core area to provide a detailed view of the changes within Ibex itself. In Section II-D, we will provide insights into the area impact of a surrounding system.

We evaluated three different Ibex implementations:

Ibex Baseline (RV32EMCB)

- 16 32-bit registers
- Support for the M (multiply/divide) extension with a single-cycle multiplier
- Support for the B (bit manipulation) extension
- Writeback (WB) pipeline-stage (3-stage pipeline)
- Dedicated Branch Target arithmetic logic unit (ALU)
- Instruction cache error correcting code (ECC)
- No PMP, CHERIoT, or dual-core lockstep (DCLS) support

Ibex PMP

- Same features as the Baseline
- Additional PMP support for 16 regions, including the *Smepmp* extension

Ibex CHERIoT

- Same features as the Baseline
- *CHERIoT* support, which includes a CHERI execute stage, checks for all memory accesses, the background revocation engine (TBRE), capability load filter, and expanded register file. The core does not include the optional stack zeroisation engine (STKZ).

TABLE I
IBEX CORE AREA AND OVERHEAD WITH DIFFERENT EXTENSIONS.

Configuration	Area (kGE)	Area Overhead (%)
Ibex	57	Baseline
Ibex+PMP	81	Baseline +42%
Ibex+CHERIoT	90	Baseline +57%

B. Results

The areas, measured in kilo gate-equivalents (kGE)[1], are reported in Table I. Note, in the FreePDK45 technology, one gate (NAND2_X1) corresponds to 0.798 µm^2 [13]. In its embedded configuration, Ibex itself is relatively small. To enable protection against memory vulnerabilities, both the PMP and CHERIoT extensions introduce an overhead roughly equivalent to half the core size. The CHERIoT-enabled core is 11% larger than the PMP-enabled core in the configurations we investigated.

C. Area Breakdown

To analyse the overhead sources in detail, we provide an area breakdown in Table II and Fig. 1, comparing the Baseline, PMP, and CHERIoT extensions. In the figure, each block is

[1]Under the tooling used in this work, specifically targeting the FreePDK45 process and using the OpenRAM memory macro compiler, a 4 KiB SRAM macro (1024 rows, 32-bit words, byte-wise write enable, 1 read/write port) corresponds to roughly 63 kGE, which translates to a density of 1.94 GE/bit. The size of the baseline Ibex would therefore correspond to roughly 3.6 KiB of SRAM memory, with CHERIoT adding another 2 KiB on top. This is intended as a rough rule of thumb to aid comparison. It is important to note that this translation factor is highly dependent on the technology node, the SRAM implementation, and the SRAM macro size and configuration.

TABLE II
AREA BREAKDOWN OF THE BASELINE, PMP, AND CHERIOT-ENABLED IBEX CORES. THE TABLE SHOWS THE AREA OF EACH BLOCK IN kGE, ALONG WITH THE PERCENTAGE OVERHEAD COMPARED TO THE BASELINE CONFIGURATION.

Block	Ibex Baseline	Ibex Baseline+PMP		Ibex Baseline+CHERIoT	
	Area (kGE)	Area (kGE)	Overhead (%)	Area (kGE)	Overhead (%)
Ibex	57.3	81.4	42.1%	90.3	57.5%
|-Core	36.0	60.2	67.2%	62.7	74.2%
| |-CSRs	7.4	13.7	84.2%	14.5	95.1%
| |-IF Stage	10.5	10.7	2.1%	10.8	2.8%
| |-ID Stage	2.8	2.8	0.4%	3.3	18.0%
| |-EX Block	13.5	13.5	0.0%	13.8	2.2%
| |-LSU	1.1	1.1	2.1%	2.1	99.1%
| |-WB Stage	0.7	0.7	0.4%	1.5	130.8%
| |-PMP	-	17.5	∞	-	-
| |-CHERI EX Block	-	-	-	12.3	∞
| |-CHERI TBRE	-	-	-	3.2	∞
| |-CHERI Load Filter	-	-	-	1.1	∞
|-ICache Data Control	9.6	9.5	–0.7%	9.5	–1.1%
|-ICache Tag Control	5.9	5.9	0.1%	5.9	0.3%
|-Register File	5.7	5.7	–0.2%	12.2	112.5%

annotated with the area in kGE and a percentage increase to the baseline version if the module changed more than 1%, while black boxes represent modules specific to the respective extension. The coloring of the blocks follows the hierarchy in the source code, with blue boxes making up the *core* module.

We first focus on the comparison between the PMP-enabled core and the baseline core. The PMP core's area increase comes primarily from two modules:

1) The **PMP block**, which implements PMP checking for 16 regions in Ibex's pipeline stage, is approximately 18 kGE in size—comparable to Ibex's ALU.
2) The **control and status registers (CSRs)** also contribute significantly. To store PMP configurations and protected address ranges, multiple CSRs must be added, depending on the number of regions. Supporting 16 address regions increases the CSR size by a factor of 1.8×.

The CHERIoT-enabled core exhibits a similar overhead pattern to the PMP one:

1) New instruction logic for the CHERI Execution Stage, TBREs, and the load filter contributes 16 kGE.
2) CSRs increase due to additional registers tracking system state, enlarging the CSR block by a factor of 1.9×.
3) Register file size doubles to accommodate the additional capability registers.
4) The load store unit (LSU) and WB stages grow to handle the extra functionality. While they double in size, their absolute contribution to the total overhead remains small, as these modules are inherently small.

Finally, comparing the PMP and CHERIoT-enabled cores directly reveals that both require additional execution blocks and CSRs, resulting in similar core sizes. The key difference lies in CHERIoT's extension of the register file, which ultimately makes the CHERIoT core 11% larger than the PMP core.

D. System Impact

So far, we have primarily discussed the impact of memory safety extensions on the Ibex core. However, in a complete system, the core typically occupies only a small fraction of the total chip area. Components such as the instruction and data paths, interconnects, peripherals, and accelerators quickly take up significantly more space—especially with respect to small processors like Ibex. As a result, even a doubling of core size would have only a minor impact on the overall chip area.

1) Earl Grey Area Breakdown: To assess the overall impact on the system, we synthesised the OpenTitan Earl Grey [14] top level, i.e., the discrete chip implementation of OpenTitan. A block diagram of Earl Grey is shown in Fig. 2, with the synthesised top level discussed here coloured in purple.

We use the successfully taped-out v1.0.0 version of the design, which is publicly available on GitHub [15], and follow the same methodology as in previous experiments, i.e., using the open-source FreePDK45 technology and targeting a main clock frequency of 100 MHz. Technology-specific and proprietary memory macros, including embedded Flash, one-time programmable (OTP) fuses, ROM, and all SRAM macros, are excluded from the logic area breakdown. As a result, the area data shown in Fig. 3 and Table III only includes synthesisable logic, such as the memory controller, and not the area occupied by memory macros. For reference, the number of memory bits each module would contain is annotated in Fig. 3, providing a rough estimate of the additional area that would be required for memory. In a realistic tape-out scenario, these memory macros typically contribute an area roughly equal to that of the logic, accounting for about 50% of the total SoC area. We will discuss their impact in more detail in Section II-D2.

The Ibex core in Earl Grey v1.0.0—featuring 32 registers, the PMP extension with 16 ranges, and a DCLS mechanism—

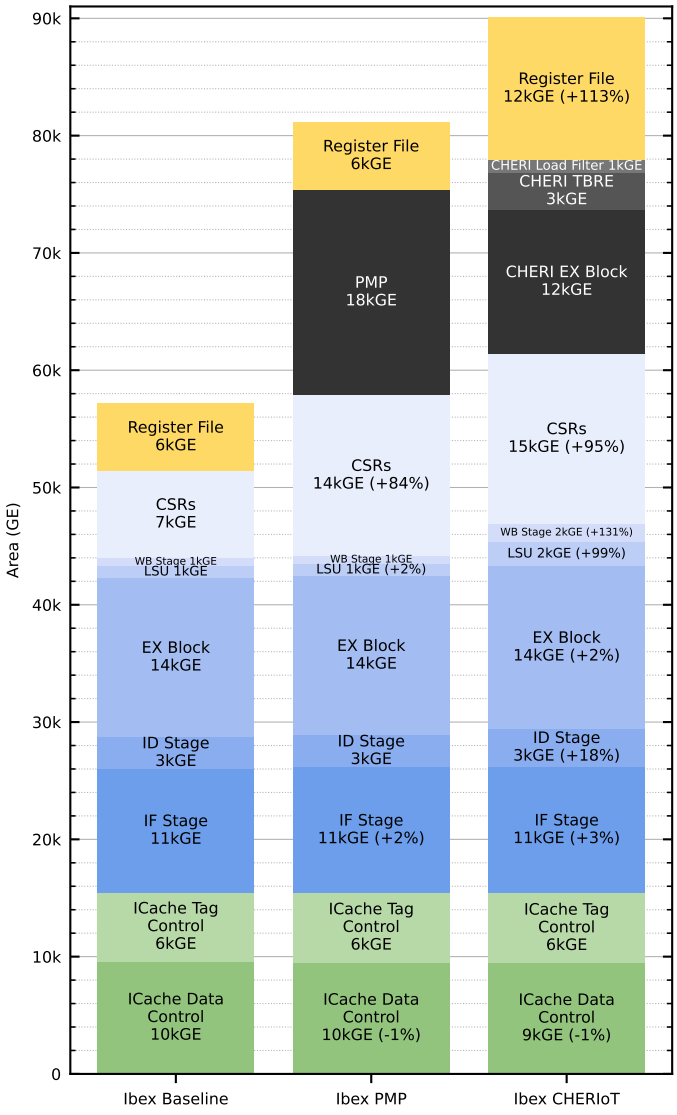


Fig. 1. Area comparison between the Baseline, the PMP-enabled, and the CHERIoT-enabled Ibex cores, with the area increase annotated relative to the Baseline Ibex for modules that change more than 1%. The modules in black are modules unique to the specific configuration.

has a logic area of 187 kGE, which is significantly larger than that of our baseline Ibex core at 57 kGE. To analyse the impact of adding PMP or CHERIoT to the system, we replace the Ibex core in Earl Grey v1.0.0 with our baseline Ibex core described in Section II. We therefore subtract the original Ibex core's logic area from Earl Grey v1.0.0's total logic area of 2190 kGE and add our baseline Ibex in its place, resulting in a new total logic area of 2061 kGE for the Earl Grey-based system using the baseline Ibex core. A detailed breakdown of this system's logic area is provided in Table III and Fig. 3.

The logic area breakdown reveals that a significant portion (63%) of Earl Grey's logic area is dedicated to security peripherals, including large accelerators like the programmable OpenTitan Big Number Accelerator (OTBN) and the Keccak Message Authentication Code (KMAC) blocks. Many of these hardware components are larger than the baseline Ibex core. The remaining logic area is distributed among I/O peripherals,

TABLE III
LOGIC AREA BREAKDOWN OF THE OPENTITAN EARL GREY BASED SYSTEM WITH THE BASELINE IBEX CORE PRESENTED HERE. THE TABLE SHOWS THE LOGIC AREA OF EACH BLOCK IN kGE, ALONG WITH THE PERCENTAGE CONTRIBUTION OF EACH BLOCK.

Block	Area (kGE)	Contribution (%)
Earl Grey based system	2060.75	100.00%
|-Baseline Ibex Core	57.32	2.78%
|-Security Peripherals	1289.99	62.60%
| |-AES	110.29	5.35%
| |-Alert handler	77.52	3.76%
| |-CSRNG	135.18	6.56%
| |-EDN (x2)	47.68	2.31%
| |-Entropy Source	99.71	4.84%
| |-HMAC	73.49	3.57%
| |-KMAC	197.14	9.57%
| |-Key Manager	93.58	4.54%
| |-Life Cycle Controller	28.03	1.36%
| |-OTBN	319.77	15.52%
| |-OTP Fuse Controller	107.59	5.22%
|-IO Peripherals	316.20	15.34%
| |-Pinmux/Padctrl	46.52	2.26%
| |-GPIO	8.09	0.39%
| |-I2C (x3)	45.75	2.22%
| |-SPI (x3)	152.84	7.42%
| |-UART (x4)	45.84	2.22%
| |-USB Device	17.16	0.83%
|-Memory Controllers	160.53	7.79%
| |-Retention SRAM Controller	16.62	0.81%
| |-Flash Controller	111.25	5.40%
| |-Main SRAM Controller	16.34	0.79%
| |-ROM Controller	16.33	0.79%
|-Others	226.03	10.97%
| |-AON Managers	116.57	5.66%
| |-Debug Module	16.44	0.80%
| |-Misc	11.48	0.56%
| |-Interrupt Controller	19.96	0.97%
| |-XBar (x2)	61.57	2.99%

memory controllers, and various supporting modules such as the interrupt controller, timers, and the debug module. Notably, the baseline Ibex core itself occupies just 2.8% of Earl Grey's total logic area.

2) PMP and CHERIoT in Earl Grey: When considering the full SoC, including proprietary macros, the split between logic area and these macros is approximately even, as previously noted. As a result, the contribution of baseline Ibex to the total SoC area is effectively halved compared to its share of the logic area. Specifically, while Ibex accounts for 2.8% of the logic area, this translates to just 1.4% of the overall SoC area. This breakdown is illustrated in Fig. 4, which highlights the distribution of logic and memory macros in Earl Grey, with Ibex's contribution clearly marked.

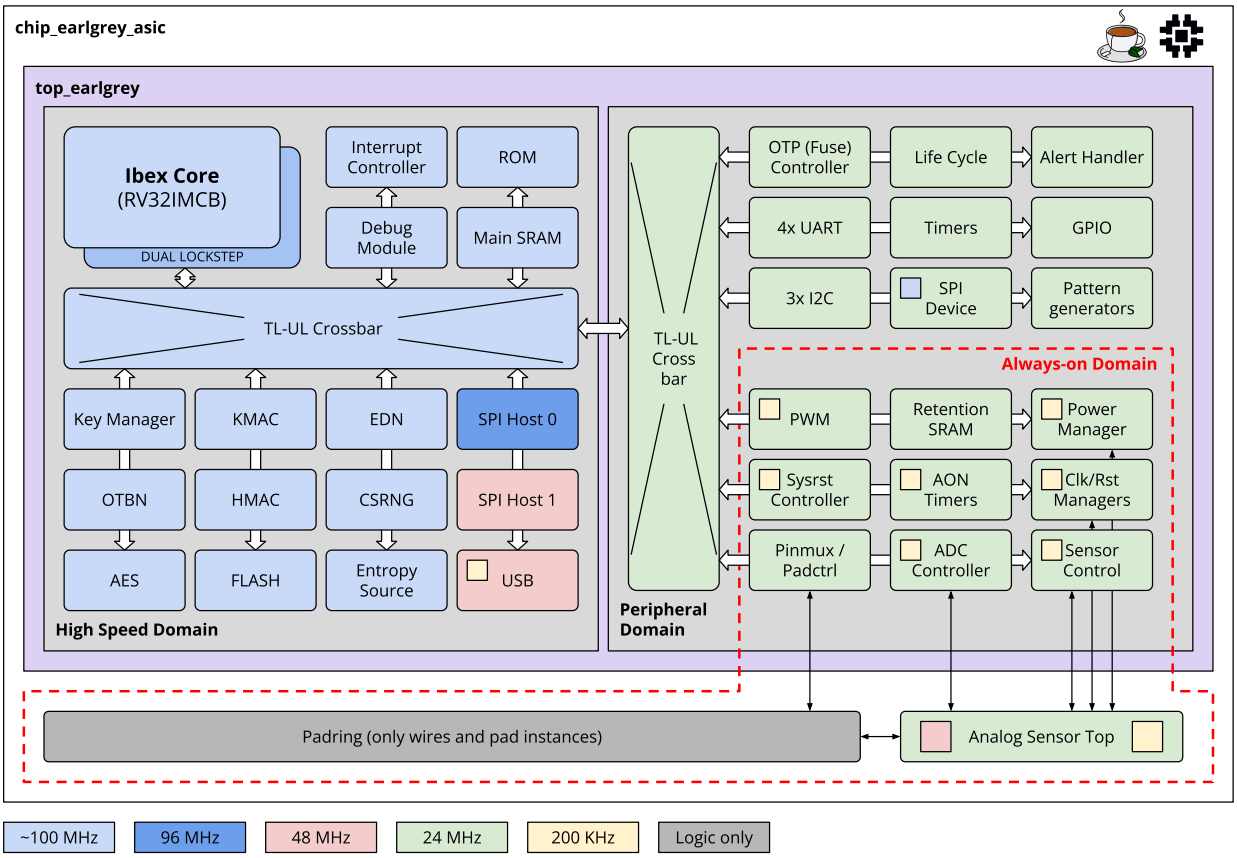
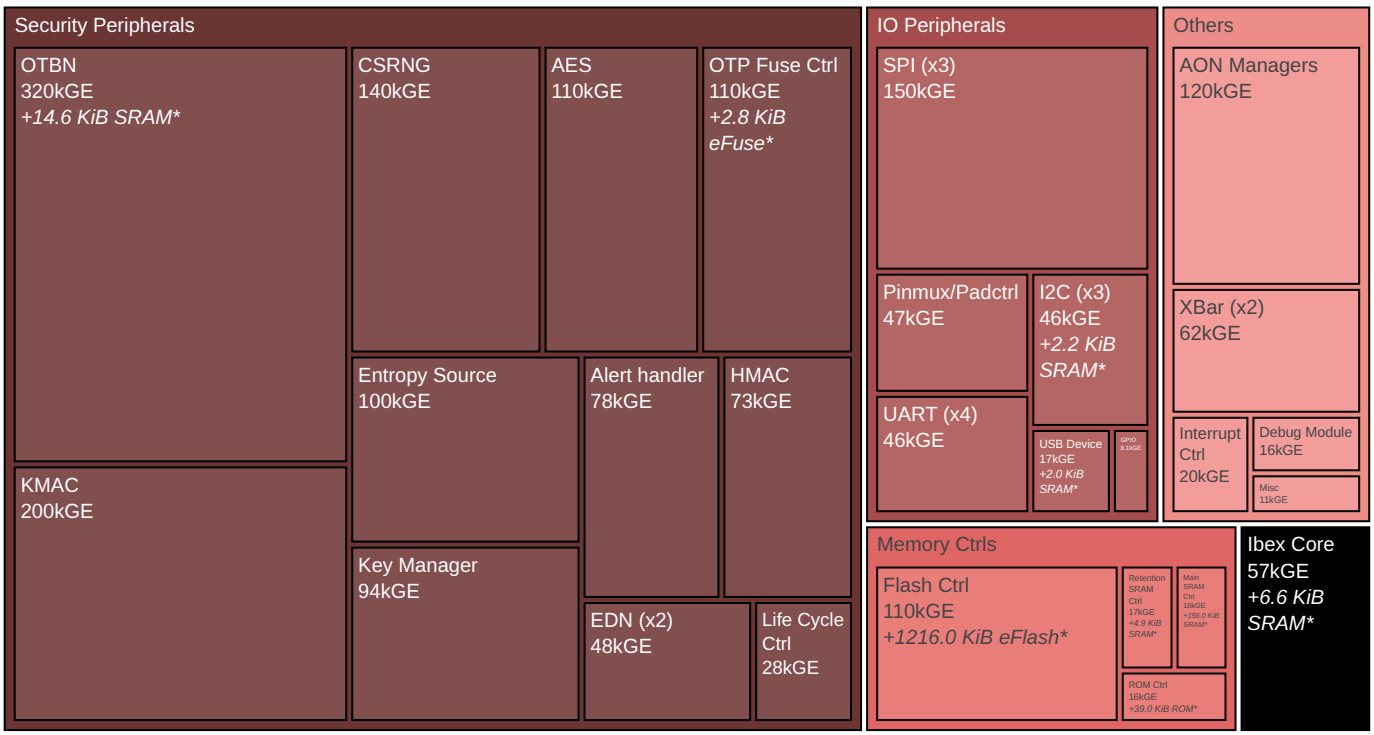


Fig. 2. Block diagram of the OpenTitan Earl Grey SoC [14]. The synthesised top level discussed in this paper is coloured in purple.



*Indicate the bits of memory that were **not** included in this breakdown of logic area but would be added to that module's total silicon area.

Fig. 3. Area breakdown of Earl Grey Top with baseline Ibex configuration. The blocks are grouped by their functionality. The *AON Managers* block comprises the PWM, Power Manager, Sysrst Controller, AON Timers, Clk/Rst Managers, ADC Controller, Sensor Control. The *Misc* block summarizes the Timers and the Pattern generators.

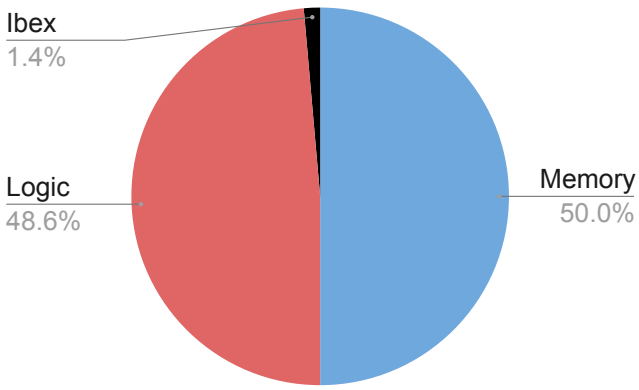


Fig. 4. Earl Grey based SoC breakdown. The pie chart shows the distribution of logic and memory macros in our Earl Grey based SoC, highlighting Ibex's contribution.

Now that we know the area ratios of Earl Grey, we can calculate the impact of our security extensions. Enabling the PMP extension increases Ibex's area by 42%. Ibex then makes up 3.9% of the total logic area, and the total logic area increases by 1.2% compared to the baseline SoC. Enabling the CHERIoT extension instead increases Ibex's area by 58%. Ibex then makes up 4.3% of the total logic area. In this version of the SoC, the logic area is 1.6% larger than the baseline SoC area. With respect to the total chip area, the area increase of these core extensions amounts to roughly 0.6% (PMP) and 0.8% (CHERIoT).

On top of these costs, security hardening of an SoC involves additional overhead beyond just the core. As discussed, CHERIoT transforms all pointers in the system into capabilities. This requires an extra validity tag and revocation tag bit per capability, which necessitates an expansion of the data memory to accommodate these extra two bits per 64 bits of data. Furthermore, the bus connecting the core to the memory needs to be one bit wider. Specifically for Earl Grey, this means to increase the width of the main and retention SRAMs from 32 to 33 bits, which results in an overall increase in chip area of 0.2%. Together with the area increase of the core extension, the estimated cost of adding support for CHERIoT in a system like OpenTitan Earl Grey is around 1%.

III. Summary

Memory safety features such as PMP and CHERIoT offer different security benefits for embedded and low-power applications. In this paper, we analyse the area overhead of incorporating them into Ibex-based processors. The results show that the CHERIoT extension causes a slightly larger increase in core area compared to PMP (57.5% vs 42.1%), which is primarily due to the expansion of the register file. However, in a complete system such as a secure microcontroller like OpenTitan Earl Grey, the estimated impact on the overall chip area would be only 0.6% (for PMP) and 1% (for CHERIoT), as the core typically occupies a small fraction of the total system size. The adoption of these memory safety features provides enhanced protection against vulnerabilities without significantly increasing the system's area, with CHERIoT additionally enforcing fine-grained spatial and temporal memory safety, as well as scalable software compartmentalisation, at only a modest area increase, making them valuable choices for systems where security is paramount.

IV. Acknowledgment

We would like to especially thank Prof. Robert N. M. Watson from the University of Cambridge and David Chisnall from SCI Semiconductor for their valuable input and feedback on this work. We also acknowledge the broader CHERI effort by the University of Cambridge and SRI International, as well as Microsoft's implementation of CHERIoT in Ibex, which made this analysis possible.

About lowRISC®

Founded in 2014 at the University of Cambridge Department of Computer Science and Technology, lowRISC is a not-for-profit company (CIC) that provides a neutral home for collaborative engineering to develop and maintain open source silicon designs and tools for the long term. The lowRISC not-for-profit structure combined with full-stack engineering capabilities in-house enables the hosting and management of high-quality projects like OpenTitan and Sunburst via the Silicon Commons® approach.

For more information, visit https://lowrisc.org/

References

[1] lowRISC, "Ibex - A Small RISC-V Core," https://github.com/lowRISC/ibex.
[2] N. Gallmann, P. Vogel, P. D. Schiavone, and L. Benini, "From swift to mighty: a cost-benefit analysis of Ibex and CV32E40P regarding application performance, power and area," in *Fifth Workshop on Computer Architecture Research with RISC-V*, 2021.
[3] A. Waterman and K. Asanović, "The RISC-V Instruction Set Manual, Volume I: User-Level ISA, Document Version 20191214-draft," RISC-V Foundation, Tech. Rep., 2019.
[4] G. Thomas, "A proactive approach to more secure code," https://msrc.microsoft.com/blog/2019/07/a-proactive-approach-to-more-secure-code, Jul. 2019.
[5] Google, "Memory safety," https://www.chromium.org/Home/chromium-security/memory-safety.
[6] Microsoft, "CHERIoT Ibex," https://github.com/microsoft/cheriot-ibex.
[7] A. Waterman, K. Asanović, and J. Hauser, "The RISC-V Instruction Set Manual, Volume II: Privileged Architecture, Document Version 20211203," RISC-V Foundation, Tech. Rep., 2021.
[8] R. N. M. Watson, S. W. Moore, P. Sewell, B. Davis, and P. Neumann, "Capability Hardware Enhanced RISC Instructions (CHERI)," https://www.cl.cam.ac.uk/research/security/ctsrd/cheri/.
[9] S. Amar, D. Chisnall, T. Chen, N. F. Wesley, B. Laurie, K. Liu, R. Norton, S. W. Moore, Y. Tao, R. N. M. Watson, and H. Xia, "CHERIoT: Complete Memory Safety for Embedded Devices," in *proceedings of the 56th IEEE/ACM International Symposium on Microarchitecture*. Association for Computing Machinery, Oct. 2023.
[10] CHERIoT Platform, https://cheriot.org, 2025, accessed: 2025-04-01.
[11] mflowgen, "FreePDK45 Open-Source 45nm Process Design Kit," https://github.com/mflowgen/freepdk-45nm/tree/d3be42a72ca6eabff54a4f3be50c16613f69f1b6.
[12] VLSIDA, "OpenRAM," https://openram.org.
[13] mflowgen, "FreePDK45 NAND2_X1 Cell," https://github.com/mflowgen/freepdk-45nm/blob/master/stdcells.lef#L4845.
[14] lowRISC contributors, "OpenTitan Earl Grey (Discrete Chip) Datasheet," https://opentitan.org/book/hw/top_earlgrey/doc/datasheet.html.
[15] lowRISC, "OpenTitan GitHub," https://github.com/lowRISC/opentitan/tree/earlgrey_1.0.0.