Programmable Fine-Grained Power Management and System Analysis of RISC-V Vector Processors in 28nm FD-SOI

Colin Schmidt, Alon Amid, John Wright, Ben Keller, Howard Mao, Keertana Settaluri,
Jarno Salomaa, Jerry Zhao, Albert Ou, Krste Asanović *Fellow, IEEE*, Borivoje Nikolić *Fellow, IEEE*

Abstract

This work presents a RISC-V system-on-chip (SoC) with fully-integrated switched-capacitor DC-DC converters, adaptive clock generators, mixed-precision floating-point vector accelerators, a 5 Gb/s serial memory interface, and an integrated power-management unit (PMU) manufactured in 28nm FD-SOI. The vector accelerator improves performance and energy per task on a matrix multiplication kernel by 15× and 13× respectively, and end-to-end performance on machine learning and graph analytical workloads by 8×-12×. Inclusion of microarchitectural counters and fine spatial power-domain granularity facilitate predictive power-management algorithms that reduce energy per task by 13-22% compared to the baseline scalar processor. System-level simulations of a range of SoC architectural variations with multiple cores and vector accelerators complement the silicon measurements.

I. System Description

The drive for increased performance under constant power envelope requires the development of accelerator-rich architectures to enable specialization for computational domains. Open-source system-on-a-chip generators based on the RISC-V instruction set architecture enable exploration of domain-specific hardware acceleration [1]. For example, popular workloads, such as deep neural networks (DNNs) may require varying precision between different layers [2] and training and inference tasks, which can be efficiently accelerated by using vector engines. In specialized architectures, accelerators dominate the area and energy budget, but given their relatively low utilization need to be efficiently managed. Simultaneously, performance of all compute systems highly depends on the memory system that supports the compute units.

The test system, named Hurricane-2, is based on 64-bit RISC-V Rocket in-order cores with a dual-lane vector accelerator and dedicated switched capacitor DC-DC units. The Rocket core is an in-order 5-stage single-issue pipeline [1] supporting the RISC-V RV64G ISA version 2.1 and the 1.9 privileged ISA. It includes separate 16KiB L1 instruction and data caches, branch predictors, page-table walker, and is capable of computing a single- or double-precision fused multiply-add (FMA) in every cycle. The Hwacha [3] vector accelerator is a multi-lane (design-time configurable) decoupled vector architecture optimized for ASIC processes. Each vector lane includes four banks of SRAM-based vector register file, a 128-bit memory port to an L2 cache, and four double-precision, eight single-precision, and 16 half-precision FMA units, enabling 8 double-precision, 16 single-precision, or 32 half-precision floating-point operations per cycle per lane.

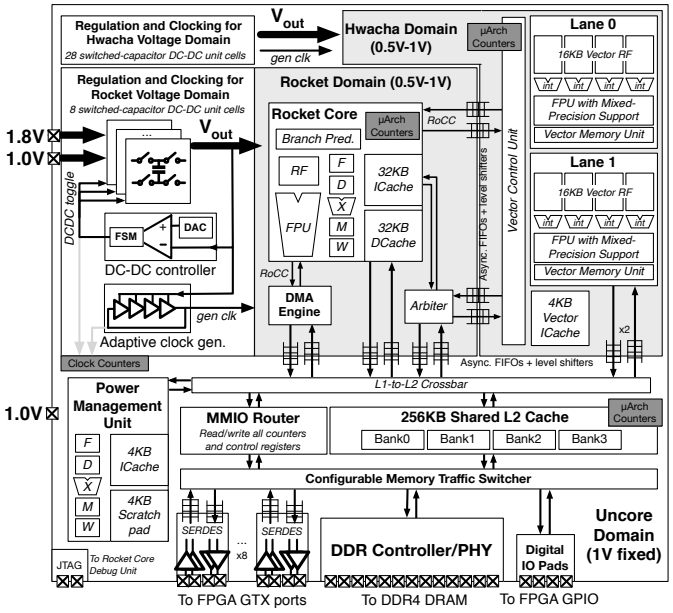

Fig. 1: Hurricane-2 SoC block diagram, with a single Rocket core, dual-lane vector accelerator, caches, power management and peripherals.

The Hurricane-2 SoC in Figure 1 contains a single Rocket application core, with a dual-lane vector accelerator capable of 16 double-precision, 32 single-precision, or 64 half-precision floating-point operations per cycle. In addition, the SoC includes a power management unit (PMU) comprised of a small integer-only RISC-V core, capable of controlling all on-chip peripherals. The application core and power management unit share a 256KiB 4-bank, 8-way L2 cache. The backside of the L2 cache is connected by a runtime-configurable switch to three off-chip memory interfaces: a low-speed 4-bit parallel interface; a test-site with a set of eight 5-Gb/s serial links; and a DDR4 PHY test site.

II. Test Chip Implementation and Measurement

The Hurricane-2 test chip was manufactured in 28nm fully-depleted silicon-on-insulator (FD-SOI) process with ultra-thin body and buried oxide (UTBB), measuring 3.9mm by 4.3mm. The die photograph is shown in Figure 2, illustrating that vector accelerators occupy a significantly larger area than the scalar Rocket core. The processor cores, the accelerator, and the digital uncore are based on the open-source Rocket Chip generator, written in Chisel. For testing, a BGA-packaged chip is attached to a daughterboard alongside an FMC connector. The daughterboard connects to a motherboard, which contains clock generators, voltage sources, and voltage and current reference generators. Finally, the motherboard attaches to an FPGA which can configure the motherboard and provide access to its DRAM over the parallel I/O and serial links.

Figure 3 shows a sweep of the chip's operating points over a range of voltages and frequencies. The most energy-efficient point is at 780mV and 115MHz, where the vector accelerator achieves 22.3 double-precision GFLOPS/W, and 36.5 half-precision GFLOPS/W. In comparison to the scalar processor's 1.76 double-precision GFLOPS/W, the vector accelerator achieves a 15× improvement in performance and a 13× increase in energy per operation over the scalar core alone. Hurricane-2's micro-benchmark performance is comparable to similar state-of-the-art systems, summarized in Table I.

III. Power Management

The Hurricane-2 SoC features a dedicated RISC-V PMU that performs power and system management functions, supporting a larger application core with an accelerator. The PMU can execute independent power management code which monitors the performance and power utilization of the system with a set of memory-mapped counters and control registers. This set of counters enables the PMU to measure the memory bandwidth being used by the application core's L1 and L2 caches, the rate of instruction execution, and the rate of energy consumption by the DC-DC converters. Hurricane-2 further includes an additional set of counters within the vector unit, tracking the type and number of instructions pending, instructions in flight, and memory operations in flight, allowing for monitoring the utilization of the vector engine.

Ref.	Hurricane-1 [4]	Hurricane-2	[5]	[6]	[7]	[2]
Technology	28nm FD-SOI	28nm FD-SOI	28nm FD-SOI	28nm FD-SOI	22nm FinFET	28nm FD-SOI
Die size (mm^2)	7.84	16.7	3.03	2.37	160	1.87
Off-chip components	No	No	No	No	Package	N/A
Peak energy efficiency	19.6 GD-FLOPS/W	36.5 GH-FLOPS/W	41.8 GD-FLOPS/W	26.2 GD-FLOPS/W	Un-specified	10 TOPS / W†
DVFS transition time (µs)	0.5	0.5	0.5	N/A	0.5	Un-specified
Volt. domain granularity	2.5 mm^2	0.5 mm^2	3.03 mm^2	2.37 mm^2	< 0.5 mm^2	0.9 mm^2

†1 four-bit MAC = 2 operations

TABLE I: Comparison with prior art

The chip includes multiples sets of switched-capacitor DC-DC converters [6] paired with adaptive clock generators [5], powering the core and accelerator voltage domains. The on-chip DC-DC converters in conjunction with the microarchitectural counters and a fully programmable PMU enable for the implementation of fine-grained dynamic voltage and frequency scaling (DVFS) algorithms.

The effectiveness of the fully-programmable PMU for fined-grained adaptive DVFS is demonstrated by comparing several DVFS algorithms across three synthetic benchmarks executed and measured on the test-chip (Figure 4). The baseline algorithm (*none*) runs the application core at maximum voltage and frequency. The *simple* algorithm replicates [5] by increasing voltage and frequency during periods of high activity noted by the DC-DC toggle rate. The last two algorithms are driven by the architectural performance counters. They monitor the miss rates of the L1 data caches (*AVS1*) and L2 cache (*AVS2*) respectively, and decrease voltage and frequency when the core is in a memory-bound program phase.

The benchmarks run on the application core and repeatedly alternate between computing a median filter and performing a generic matrix multiply (GEMM) of 24-, 64-, or 128-element square matrices. The 24-element dataset fits in L1 cache,

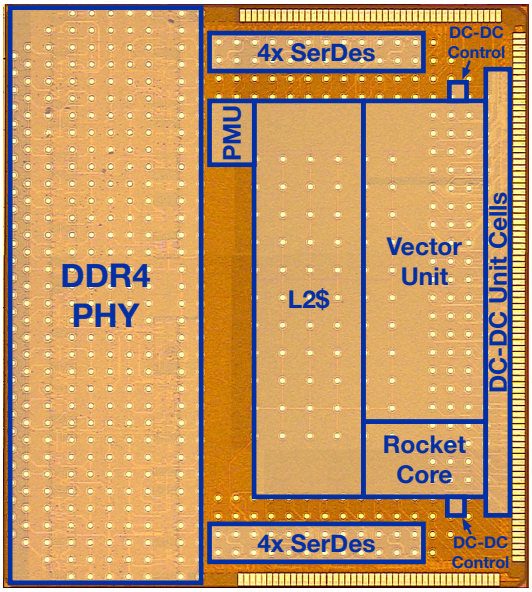

Fig. 2: Hurricane 2 chip micrograph.

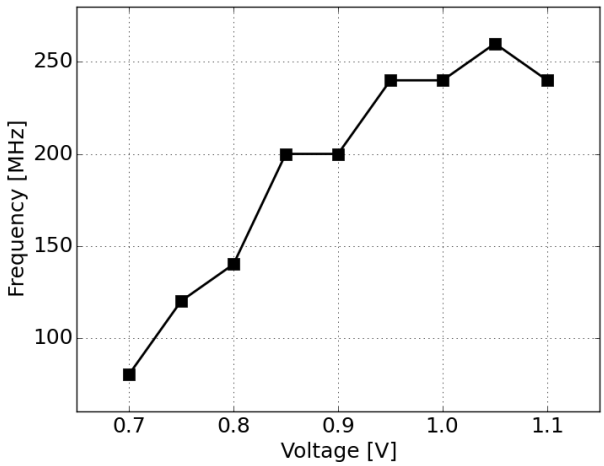

Fig. 3: Hurricane-2 matrix multiplication (DGEMM) shmoo plot.

so no cache misses occur, obviating the adaptive algorithms. The 64-element dataset only fits in L2 cache. The adaptive algorithms are able to identify the memory-bound regions, but the L1 monitor has false positives for phases that miss in the L1 but hit in the L2. Finally, in the 128-element benchmark, all L1 misses become L2 misses, so both adaptive algorithms successfully slow down the core during memory-bound phases, saving energy in the core. These measurements demonstrate that fine-grained power management based on monitoring architectural counters, when they are available, outperforms management solutions based on monitoring power consumption due to its faster and deterministic response. Specifically, the example in Figure 4 demonstrates that the addition and fine-grained monitoring of just two key architectural performance counters can provide up to 14% additional energy savings compared to traditional DVFS algorithms.

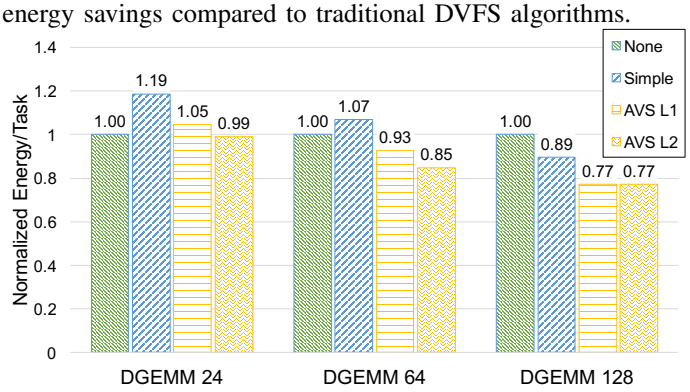

Fig. 4: Comparison of DVS algorithms in Hurricane-2

IV. Vector Processor System Analysis

The Hwacha vector accelerator was designed to execute data-parallel workloads in Linux-based data-intensive applications. Common workloads require a GNU/Linux environment to utilize a variety of libraries and system services. The Caffe machine learning framework running the SqueezeNext [8] deep neural network (DNN) is a representative application for running full-system machine-learning inference and training, while a PageRank workload integrated into the GraphMat [9] framework represents a sparse full-system workload for graph processing. We have used FireSim [10] to evaluate these applications across the architectural design space of SoCs using the Hwacha vector accelerator (Fig. 5).

FireSim is an open-source FPGA-accelerated RTL-derived cycle-exact simulation environment on the Amazon EC2 F1 public cloud. The FireSim environment constructs an FPGA-based simulator from the source RTL of the simulated design, making it a single-source-of-truth for both simulation and chip designs. FireSim enables integration of the simulated SoC with various peripheral and system-level models such as DDR3 memory [11], Ethernet NIC models, and UART interfaces. Firesim has been recently used to evaluate the performance of a commercial SoC [12]. Unlike conventional FPGA prototyping approaches, FireSim and its underlying compiler transform the target RTL description into a simulator rather than synthesize the target RTL onto the FPGA. This transformation enables decoupling between the simulated target design and the host FPGA platform, which allows for deterministic timing-accurate modeling of memory and I/O interfaces. In particular, full-system evaluation requires an outer memory system, which is not available on most test chips. The Hurricane-2 SoC includes an experimental third-party DDR3/4 interface, which was not functional on our test chip. Instead, each simulated SoC instance includes a DDR3 memory model with a 14-14-14 speed-grade. We evaluate twelve different SoC configurations by using FireSim and varying the number of processor tiles, the number of vector lanes, and the size of L2 cache.

In particular, we are interested in comparing the Hurricane-2 architecture with the architecture of our previous test chip, Hurricane-1 [4], which was also fabricated in 28nm UTBB FD-SOI process and implements an architecture that contains dual Rocket application cores, each with a single-lane Hwacha vector accelerator. The Hurricane-1 configuration enables additional task-level parallelism, in contrast to Hurricane-2's single-core, dual-lane configuration which exploits more data-level parallelism.

Evaluation results confirm that application parameters such as batch size or graph size affect the speedup obtained from an SoC configuration (Fig. 6). For DNN inference, additional vector lanes do not provide additional speedup for batch size increase from 1 to 16. Hence, for infrequent inference applications, a dual-lane/single-tile SoC configuration such as Hurricane-2 is more beneficial, while for batch-oriented applications, a dual-tile/single-lane configuration such as Hurricane-1 is better. This is not an obvious result, as batched applications are known to expose more data-level parallelism than unbatched applications. We postulate that this result is due to non-vectorized processing that is required of the additional data in the batched workload. This non-vectorized processing is parallelized across the two scalar cores in the Hurricane-1, while it cannot be parallelized using the single scalar core in the Hurricane-2. In PageRank, the most significant speedup (25x) is obtained for a small graph that fits in L2 cache and utilizes the added computational resources. Between the Hurricane-1 and Hurricane-2 configurations, the dual-tile/single-lane design achieves the greatest PageRank speedup, because fine-grained load balancing across the decoupled cores targets the irregular structure of graph representations better. Notably, variations in L2 cache size have minimal effects on all workload types, indicating little temporal locality within the applications. The only significant effect of cache size on speedup relates to the PageRank execution on the wikiVote graph which fits entirely within even the smallest evaluated cache size.

V. Conclusion

This work demonstrates programmable fine-grained power-management and system analysis of RISC-V vector processors. Through hybrid use of test chips and FPGA-accelerated simulation, this work presents both silicon measurements and FPGA-simulated full-system analysis of power and performance resulting in a 13-22% energy-per-task improvement and 8×-12× performance improvement on relevant workloads.

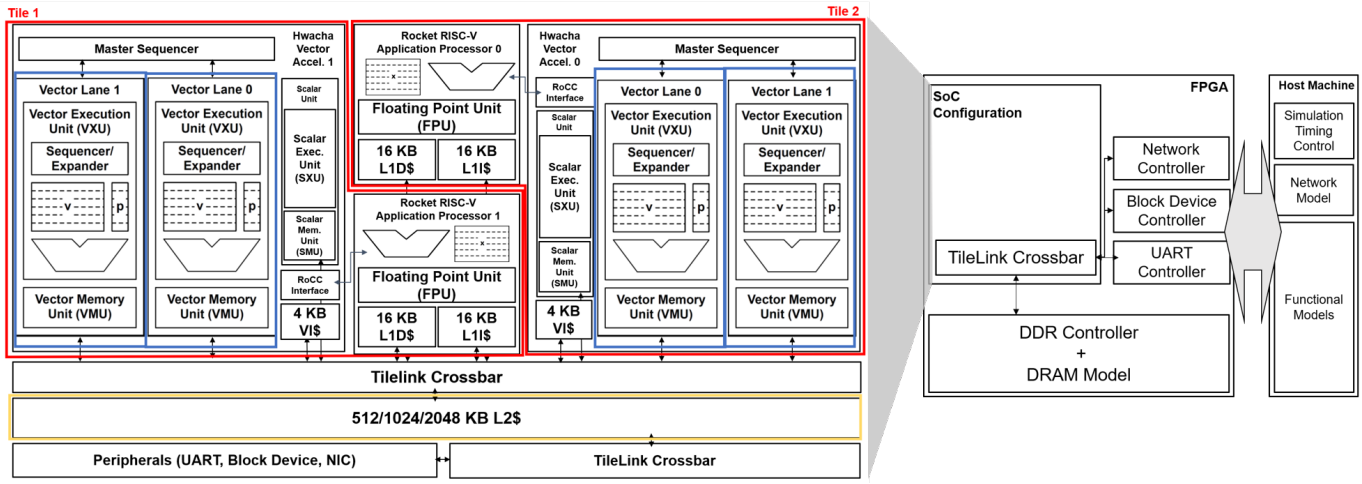

Fig. 5: FPGA-accelerated simulation with DDR3, block device, network, and other peripheral models, using various configurations of generated target SoC RTL. The generated configurations are varied across the numbers of tiles (marked in red), numbers of vector lanes per tile (marked in blue) and the L2 cache size (marked in yellow)

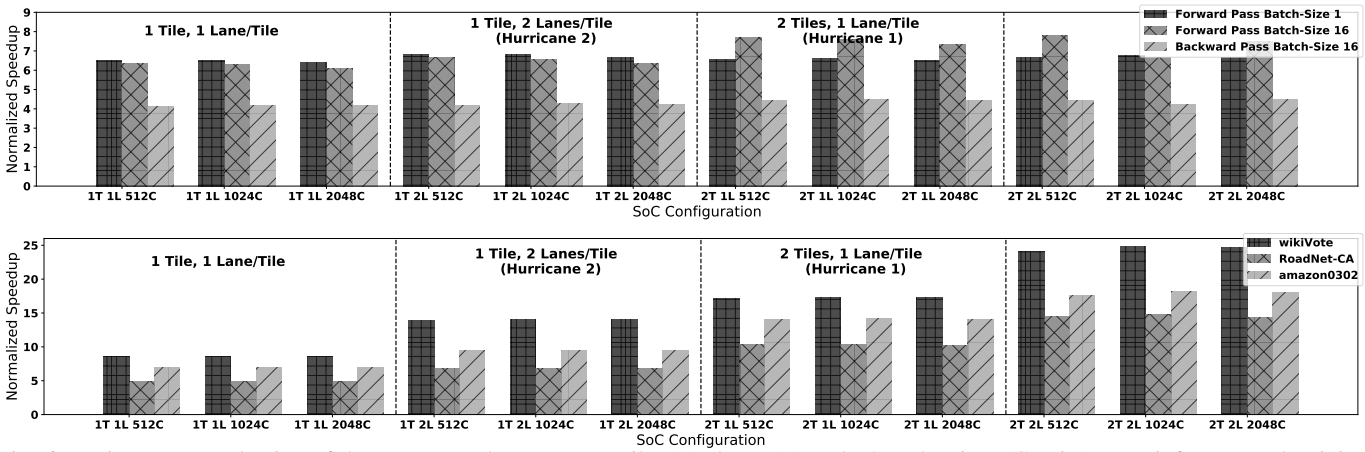

Fig. 6: Design space evaluation of the vector accelerator across tiles (T), lanes (L) and L2 cache sizes (C, KiB). Top: inference and training of the SqueezeNext DNN model on the Caffe framework using the vector accelerator compared to a minimal reference scalar implementation (1T512C). Larger batch sizes expose more parallelism, which enables improved performance using multiple tiles and multiple vector lanes. Note that L2 cache size does not have a consistent effect. Bottom: vectorized PageRank using GraphMat infrastructure on different graphs, compared to a minimal reference scalar implementation. The dual-tile/single-lane configuration provides greater speedup than the single-tile/dual-lane configuration (with similar area overheads). L2 cache size is not a factor for speedups on large graphs.

VI. Acknowledgments

The authors would like to thank Andreia Cathelin, Didier Campos, and Vince Mangion at STMicroelectronics for their support and the design of the package. The information, data, or work presented herein was funded in part by the Defense Advanced Research Projects Agency (DARPA) through the PERFECT Program. Research was partially funded by ASPIRE and ADEPT Lab industrial sponsors and affiliates. We thank STMicroelectronics for donating prototype fabrication. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

References

[1] K. Asanović *et al.*, "The rocket chip generator," Tech. Rep. UCB/EECS-2016-17.
[2] B. Moons *et al.*, "Envision: A 0.26-to-10TOPS/W subword-parallel dynamic-voltage-accuracy-frequency-scalable convolutional neural network processor in 28nm FDSOI," in *ISSCC*, 2017, pp. 246–247.
[3] Y. Lee *et al.*, "Hwacha preliminary evaluation results, version 3.8." Tech. Rep. UCB/EECS-2015-264.
[4] J. Wright *et al.*, "A dual-core risc-v vector processor with on-chip fine-grain power management in 28nm fd-soi," *IEEE TVLSI*, p. under review, 2020.
[5] B. Keller *et al.*, "Sub-microsecond adaptive voltage scaling in a 28nm FD-SOI processor SoC," in *ESSCIRC*, 2016, pp. 269–272.
[6] B. Zimmer *et al.*, "A RISC-V vector processor with tightly-integrated switched-capacitor dc-dc converters in 28nm fdsoi," in *VLSI-C*, 2015, pp. C316–C317.
[7] E. A. Burton *et al.*, "FIVR — fully integrated voltage regulators on 4th generation intel core SoCs," in *APEC*, 2014, pp. 432–439.
[8] A. Gholami *et al.*, "Squeezenext: Hardware-aware neural network design," *arXiv preprint 1803.10615*, 2018.
[9] N. Sundaram *et al.*, "Graphmat: High performance graph analytics made productive," *Proc. VLDB Endow.*, vol. 8, no. 11, pp. 1214–1225, 2015.
[10] S. Karandikar *et al.*, "Firesim: Fpga-accelerated cycle-exact scale-out system simulation in the public cloud," in *ISCA*, 2018, pp. 29–42.
[11] D. Biancolin *et al.*, "FASED: FPGA-accelerated simulation and evaluation of dram," in *FPGA*, 2019.
[12] Y. Lee *et al.*, "Managing chip design complexity in the domain-specific SoC era," in *VLSI-C*, 2020.