Marian: An Open-Source Implementation of the RISC-V Vector Cryptography Extension

Thomas Szymkowiak, Endrit Isufi, **Markku-Juhani O. Saarinen**
Tampere University, Finland
<markku-juhani.saarinen@tuni.fi>

RISC-V Summit North America
Santa Clara -- October 22, 2024

Tampere University

Outline: Marian – Integrating Zvk with ARA2

- **Marian** is the first open-source hardware implementation of a vector processor with the Zvk extensions. Implementation is in SystemVerilog.
- Design is based on the PULP “Ara” vector unit [1].
- Tested using a Virtex Ultrascale+ FPGA prototype, with a planned tapeout (ASIC flows) targeting a 22nm process node.

[1] M. Perotti, M. Cavalcante, R. Andri, L. Cavigelli, and L. Benini. “*Ara2: Exploring Single- and Multi-Core Vector Processing with an Efficient RVV 1.0 Compliant Open-Source Processor*”. IEEE Transactions on Computers 73.7 (July 2024), pp. 1822–1836.

RISC-V Cryptography Extensions ("K")

Zkt Scalar Crypto (Ratified 2021)**:** AES, SHA2, SM3, SM4, CMUL (GCM) with 32- and 64-bit **scalar registers**. + "Constant time" & Entropy Source.

Zvkt Vector Crypto (Ratified 2023)**:** AES, SHA2, SM3, SM4, GCM with **vector registers**: Make bulk crypto even faster with *parallel* AES-GCM etc.

-> support in QEMU, Linux Kernel, OpenSSL, going into Android Platform

-> however, no open-source hardware PoC available – topic of this talk

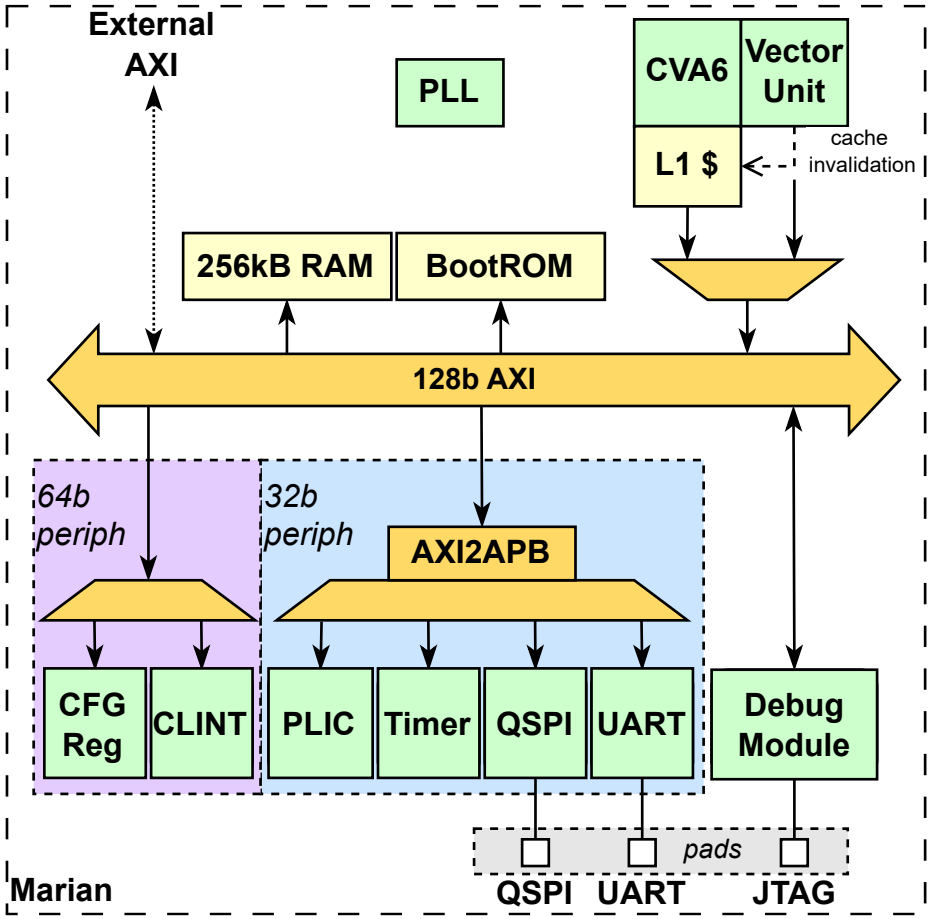
Active future ratification efforts:

High Assurance Crypto TG, Post-Quantum Crypto TG (From late 2023)

System Overview

- Marian extends the ARA2 Vector Unit with several new components.
- ARA2 itself extends the CVA6 core and has its own load-store unit.
- This is instantiated on as a "sub-system" (for a SoC Hub test chip.)
- For our end-to-end benchmarks we ran this on UltraScale+ FPGA target.

https://github.com/soc-hub-fi/Marian

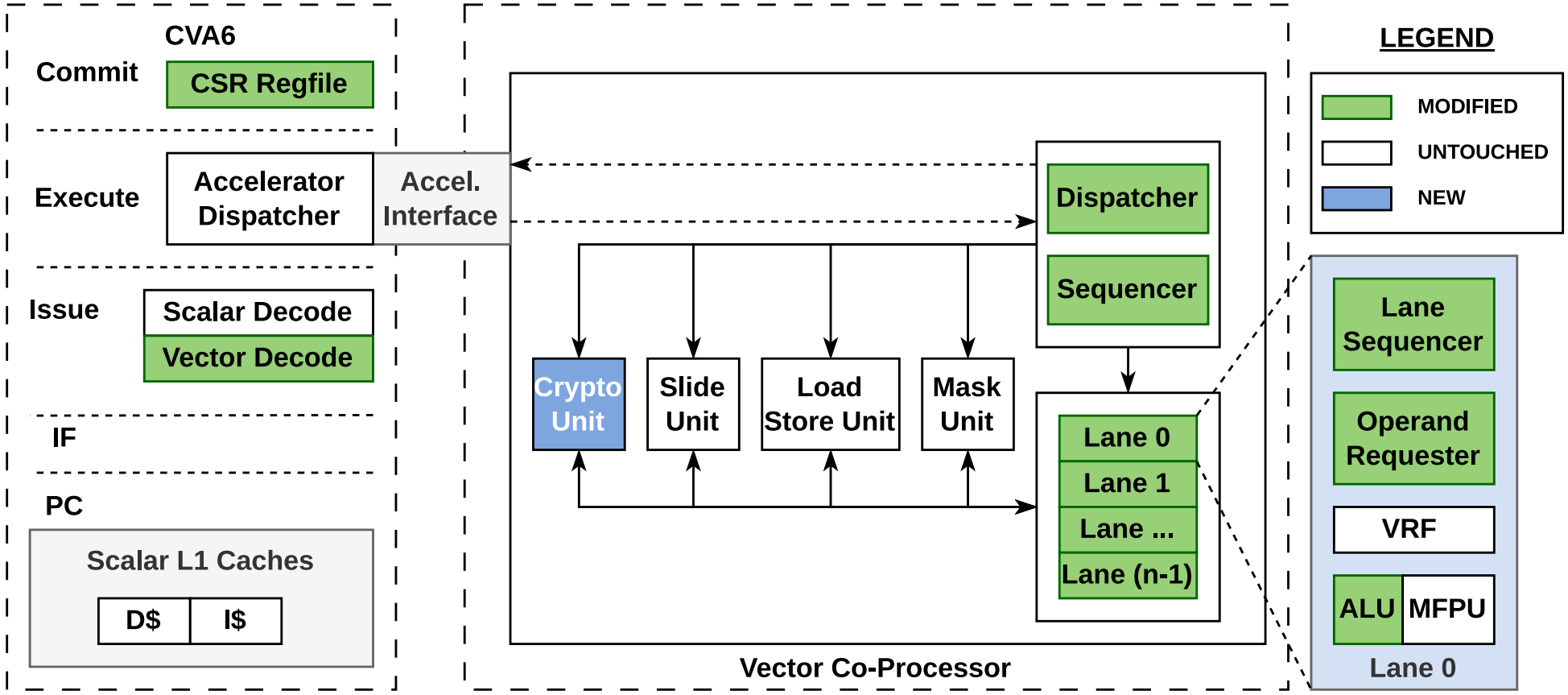


Zvk Instructions & Element Groups

- Zvk is in the ISA manual now.
- Implements all instruction groups (China & U.S. NIST.)
- All but Zvkb requires *element groups*; EGW is 128 or 256 bits. This is pehaps the most novel part of implementation.
- Some instruction critical paths are relatively long (esp. GCM.)
- All instructions DIEL / Zvkt.

Group	Ext.	EGW	EEW	EGS	Mnemonics	
AES	Zvkned	128	32	4	vaesef	vaesem
					vaesdf	vaesdm
					vaeskf1	vaeskf2
					vaesz	
SHA256	Zvknha	128	32	4	vsha2ms	vsha2c[hl]
SHA512	Zvknhb	256	64	4	vsha2ms	vsha2c[hl]
GCM	Zvkg	128	32	4	vghsh	vgmul
SM4	Zvksed	128	32	4	vsm4k	vsm4r
SM3	Zvksh	256	32	8	vsm3me	vsm3c
BitManip	Zvkb			1	vrol	vror
					vbrev8	vrev8
					vandn	
Random	Zkr	Entropy Source CSRs: seed.				
DIEL	Zvkt	Asserts a “constant-time list” of instructions.				

New Crypto Unit: Modifications & Additions



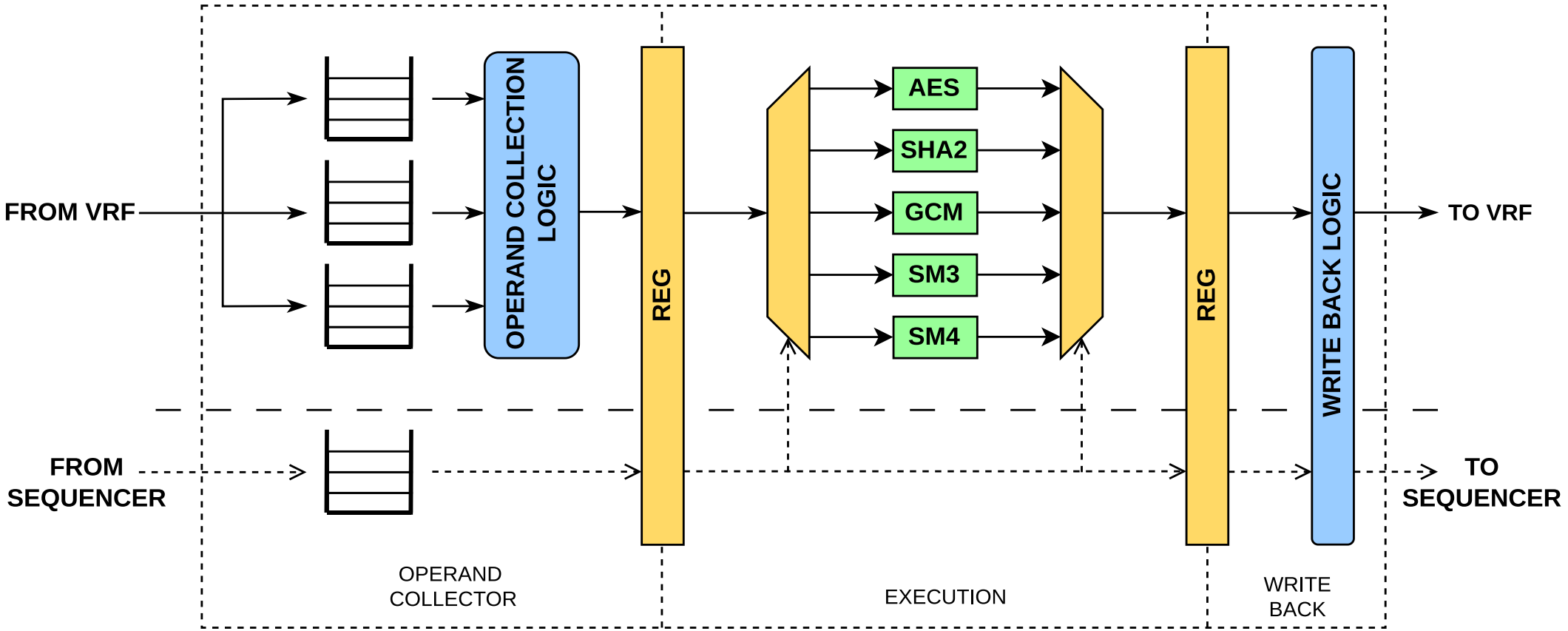
Crypto Unit Architecture

- Arithmetic logic of the cryptographic operations was separated from the vector lane logic and placed into a dedicate Crypto Unit module to reduce design complexity. (Ara lanes are ELEN = 64.)

- The Crypto Unit is a three-stage, latency-insensitive pipeline consisting of an operand collection stage, an execution unit stage, and a write-back stage.

- Instructions are executed and results are written back in-order.

Operand Collector & Write-Back



6x..100x Bulk Encryption Performance

Observed a 6x-100x speedup in cycles and between 12x-300x reduction in instruction count compared to OpenSSL C code.

Timing countermeasures make the reference AES very slow.

The Zvk implementations are DIEL / "constant-time" (Zvkt).

Operation		Reference		Zvk	
		Cycles	Instret	Cycles	Instret
AES128	Enc. block	18,794	12,482	343	53
	Dec. block	23,731	15,077	226	53
AES256	Enc. block	24,493	17,478	441	65
	Dec. block	32,677	21,213	278	65
SM4	Enc. block	4,187	1,423	272	39
	Dec. block	2,564	1,425	178	39
SHA256	Hash / 1kB	156,205	82,179	12,106	3,802
SHA512	Hash / 1kB	109,905	45,903	9,140	2,712
SM3	Hash / 1kB	304,031	70,075	8,134	1,410

Note: AES OpenSSL C Reference is “ Constant-Time” , SM4 is not.

FPGA & ASIC Resource Reports

VCU118 (UltraScale+) @ 75 MHz

Top Module	Registers	LUT (logic)	LUT (RAM)	BRAM (kB)	DSP
Marian	115,767	420,056	1908	360	225
CVA6	24,924	40,900	884	117	28
Vector Unit	67,680	322,474	1,024	0	197
Lane (single)	14,513	57,175	256	0	49
Crypto Unit	2,800	33,465	0	0	0

ASIC 22nm LP with F_{max} 1GHz

Top Module	Logic Cell Area (mm^2)	Area (kGE)	% of Total Area
Marian	2.08	1834.263	100%
CVA6	0.28	242.973	13.25%
Vector Unit	1.04	915.536	49.91%
Lane (single)	0.21	181.040	9.87%
Crypto Unit	0.14	118.131	6.44%

- End-to-end functional & cycle count testing was performed on the FPGA.
- The ASIC numbers are from mock synthesis – no tapeout yet.

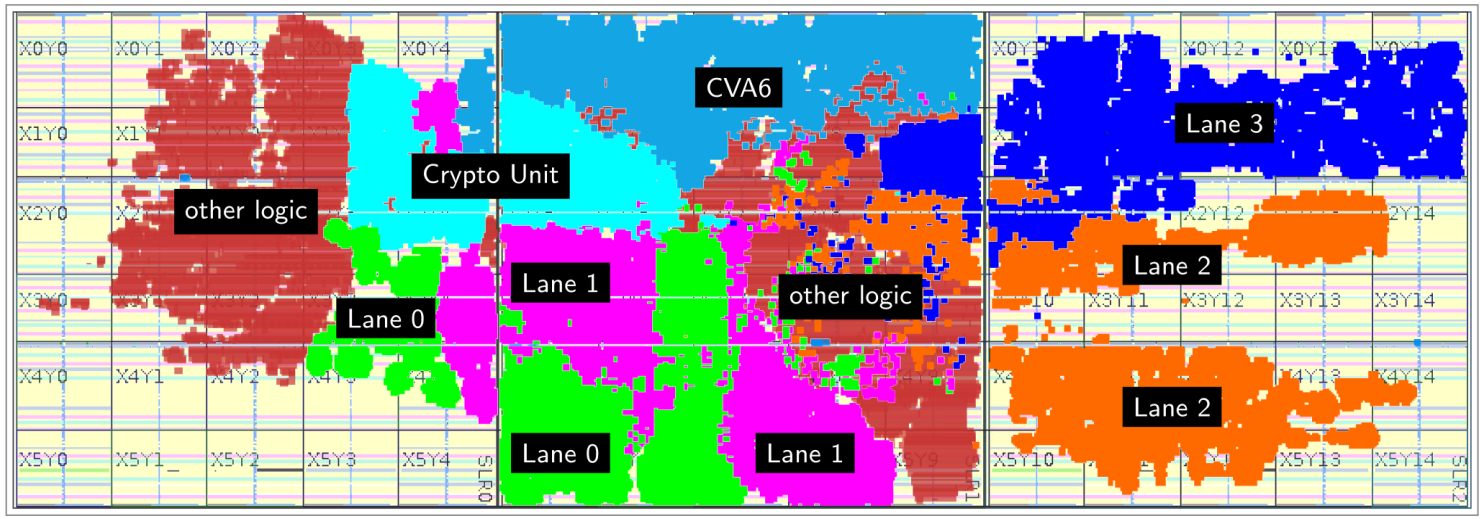
Findings & Conclusions

- Zkt is likely to require its own unit (outside "lanes") due to ELEN.
- Do it! Even the most primitive, bolt-on vector cryptography implementation (like ours) will give very substantial performance gains.
- Additional area for single "full" Zvk (AES/SHA2/SM4/SM3) was ~100 kGE.

For high performance implementation..

- Consider 128- and 256- bit EGW when organizing your vector register file. AES/SM4 and GCM can utilize full data parallelism.
- Implement AES pipelining (AES is "always" the same instruction sequence).
- Allow GCM to be in flight during AES execution, etc.

Thank You! Questions?



Source code: https://github.com/soc-hub-fi/Marian

Short writeup: T. Szymkowiak, E. Isufi, M.-J. Saarinen. *"Marian: An Open Source RISC-V Processor with Zvk Vector Cryptography Extensions."* https://ia.cr/2024/1449