



Low-Stack HAETAE for Memory-Constrained Microcontrollers

YoungBeom Kim¹, Christine van Vredendaal², Gustavo Banegas³, and Seog Chung Seo¹

¹Kookmin University ²NXP Semiconductors ³LIX, CNRS, Inria, École Polytechnique

CHES 2026 · Antalya, Türkiye · October 2026

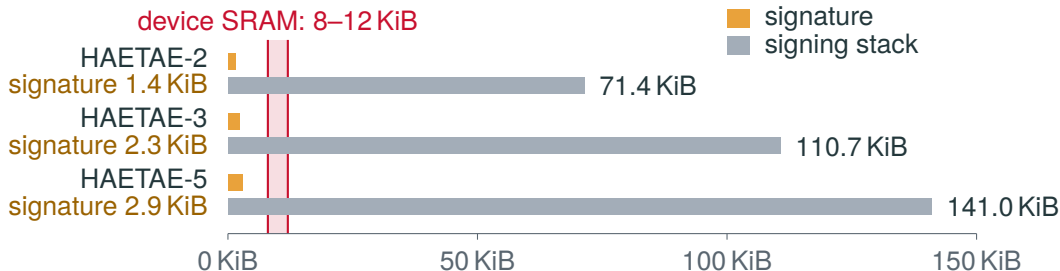
HAETAE

- ▶ Signature over $R_q = \mathbb{Z}_q[X]/(X^N+1)$, module lattices; CHES'24, KpqC winner.
- ▶ Fiat–Shamir with aborts, like ML-DSA: commit y , challenge c , response $z = y + (-1)^b c \star s$, reject.
- ▶ y from a discretized [hyperball](#): shorter keys and signatures; costlier sampler.
- ▶ $\sigma = (\text{HB}(z_1), \text{LB}(z_1), h, c)$, rANS-encoded; the hint h recovers z_2 .

	HAETAE-2	HAETAE-3	HAETAE-5
N, q	256, 64,513		
$ \text{poly} $	$4N = 1,024$ B (32-bit coefficients)		
(k, ℓ)	(2,4)	(3,6)	(4,7)
τ	58	80	128
d	1	1	0
$ pk $	992	1,472	2,080
$ sk $	1,408	2,112	2,752
$ \sigma $	1,474	2,349	2,948
ML-DSA $ \sigma $	2,420	3,309	4,627

Small signatures, large signing memory

- ▶ Signatures: 29–39% shorter than ML-DSA's.
- ▶ Reference signing stack: **about 50× the signature size**.
- ▶ ML-DSA m4fstack: 5–8 KiB signing stack (pqm4; BRS22 strategy).



Same high-level shape, three differences

ML-DSA.Sign(sk, M)

1: $\mu \leftarrow H(tr, M)$; $\mathbf{A} \leftarrow \text{ExpandA}(\rho)$
2: **repeat**
3: $\mathbf{y} \leftarrow \text{uniform in } [-\gamma_1+1, \gamma_1]^\ell$
4: $\mathbf{w} \leftarrow \mathbf{A}\mathbf{y}$
5: $c \leftarrow H(\mu, \text{HB}(\mathbf{w}))$
6: $\mathbf{z} \leftarrow \mathbf{y} + c\mathbf{s}_1$
7: **until** $\|\mathbf{z}\|_\infty < \gamma_1 - \beta$ and
 $\|\text{LB}(\mathbf{w} - c\mathbf{s}_2)\|_\infty < \gamma_2 - \beta$
8: $\mathbf{h} \leftarrow \text{MakeHint}(-c\mathbf{t}_0, \mathbf{w} - c\mathbf{s}_2 + c\mathbf{t}_0)$
9: **return** $(\tilde{c}, \mathbf{z}, \mathbf{h})$, fixed-length bits

FIPS 204

HAETAE.Sign(sk, M)

1: $\mu \leftarrow H(tr, M)$; $\mathbf{A} \leftarrow \text{UnpackA}_d(\text{seed}_{\mathbf{A}}, \psi)$
2: **repeat**
3: $\mathbf{x} \leftarrow \text{Gaussian}$; $\mathbf{y} \leftarrow \lfloor \alpha \mathbf{x} \rfloor$, $\alpha = B_0 \Lambda / \|\mathbf{x}\|_2$
4: $\mathbf{w} \leftarrow \mathbf{A} \lfloor \mathbf{y}_1 \rfloor + 2 \lfloor \mathbf{y}_2 \rfloor \bmod q$
5: $c \leftarrow H(\text{HB}(\mathbf{w}'), \text{LSB}(\lfloor \mathbf{y}_{1,1} \rfloor), \mu)$
6: $\mathbf{z} \leftarrow \mathbf{y} + (-1)^b c\mathbf{s}$
7: **until** $\|\mathbf{z}\|_2 < B'$ and $(\|2\mathbf{z} - \mathbf{y}\|_2 > B \text{ or } b')$
8: $\mathbf{h} \leftarrow \text{HB}(\mathbf{w}') - \text{HB}(\mathbf{w}' - 2 \lfloor \mathbf{z}_2 \rfloor)$
9: **return** $\text{rANS}(\text{HB}(\mathbf{z}_1), \mathbf{h}) \parallel \text{LB}(\mathbf{z}_1) \parallel c$

Alg. 3 in the paper

Blue: (3, 7) norm of the full vector;

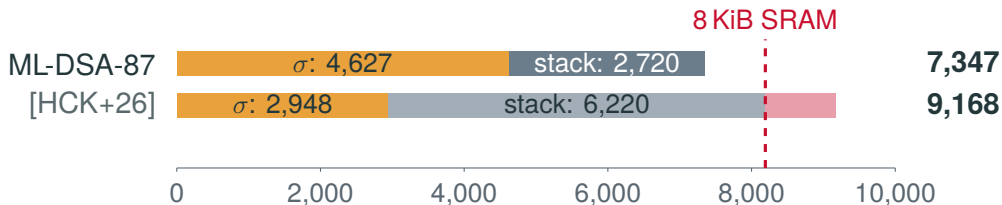
(8) hint: reference keeps $\mathbf{w}$ and $\mathbf{z}$ live; $\mathbf{z}_2$ not sent;

(9) rANS: last symbol first.

Gray: shared operations. BRS22 applies directly to matrix streaming and sparse challenge multiplication.
Both listings omit secondary checks.

Prior HAETAE-5 verification exceeds 8 KiB

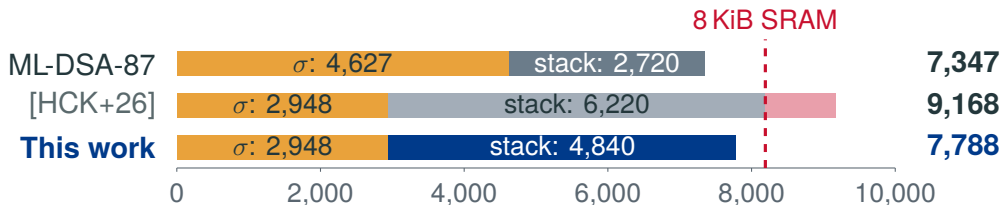
- ▶ Firmware Verify (Cortex-M4): σ + stack; pk in Flash.
- ▶ Low-stack ML-DSA [BRS22]: fits;
prior low-stack HAETAE-5 [HCK+26]: exceeds budget.



Level 5 on Cortex-M4: measured stack (pqm4) + $|\sigma|$, pk in Flash; ML-DSA-87 $|\sigma|$ from FIPS 204. 9,168 B as reported; other totals computed. [HCK+26] gives low-stack results for HAETAE-5 only.

Prior HAETAE-5 verification exceeds 8 KiB

- ▶ Firmware Verify (Cortex-M4): σ + stack; pk in Flash.
- ▶ Low-stack ML-DSA [BRS22]: fits;
prior low-stack HAETAE-5 [HCK+26]: exceeds budget.



- ▶ **This work:** Verify fits in 8 KiB at every level, like ML-DSA.
- ▶ KeyGen/Sign need more; 12 KiB results next.

Level 5 on Cortex-M4: measured stack (pqm4) + $|\sigma|$, pk in Flash; ML-DSA-87 $|\sigma|$ from FIPS 204. 9,168 B as reported; other totals computed. [HCK+26] gives low-stack results for HAETAE-5 only.

It runs on real 8 KiB and 12 KiB boards

ATmega2560

8-bit AVR, 8 KiB SRAM

- ▶ Runs Verify, all levels
- ▶ HAETAE-5: σ + stack (pk in Flash)
7,961 B
- ▶ Headroom: 25 B of 7,986 B
(8,192 B – 206 B of harness data)
- ▶ ML-DSA on AVR [HKS25]: simulator,
memory budget over 16 KiB

STM32C031

Cortex-M0+, 12 KiB SRAM

- ▶ Runs KeyGen, Sign, Verify, all levels
- ▶ HAETAE-5 Sign: $sk + \sigma$ + stack
12,056 B
- ▶ Within 12 KiB = 12,288 B; all levels fit
- ▶ For scale: ML-DSA-65 m4fstack
needs 13,949 B (Cortex-M4, pqm4)
- ▶ HAETAE-5 [HCK+26], Cortex-M4 (C):
Sign total 13,792 B > 12 KiB

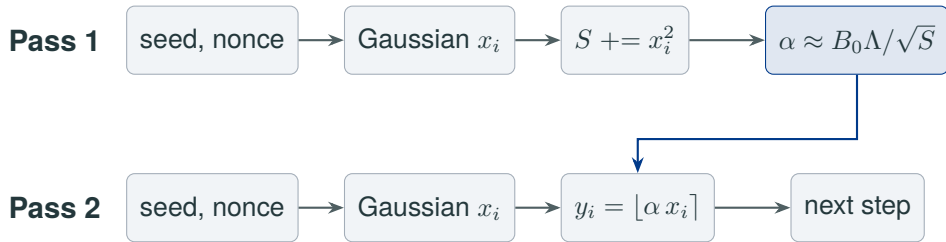
Memory fits; cycle counts follow.

Boards: measured, ours (C). Comparisons: AVR simulator / Cortex-M4.

AVR: 7,986 B for σ + stack after 206 B of harness static data.

Techniques

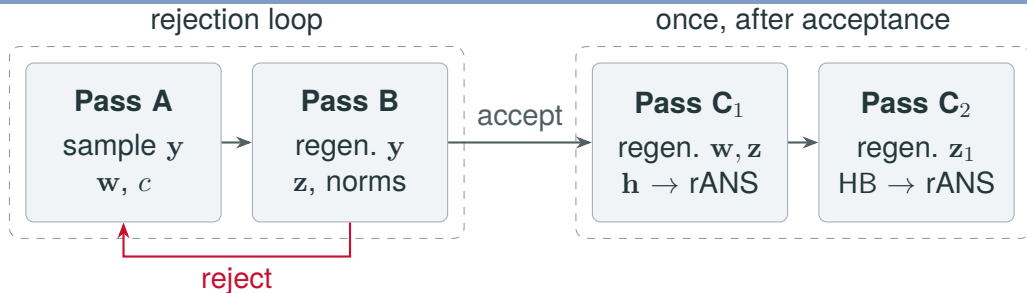
Two-pass hyperball sampler



- ▶ $O(1)$ extra sampler memory instead of $(\ell+k)N$ samples
- ▶ Same (seed, nonce), same Gaussian samples and output
- ▶ Two-pass approach [LWKP25, HCK+26];
streaming Gaussian backend: `.bss = 0`

Anything seed-derived can be regenerated: what matters is the order.

Rejection-aware pass decomposition

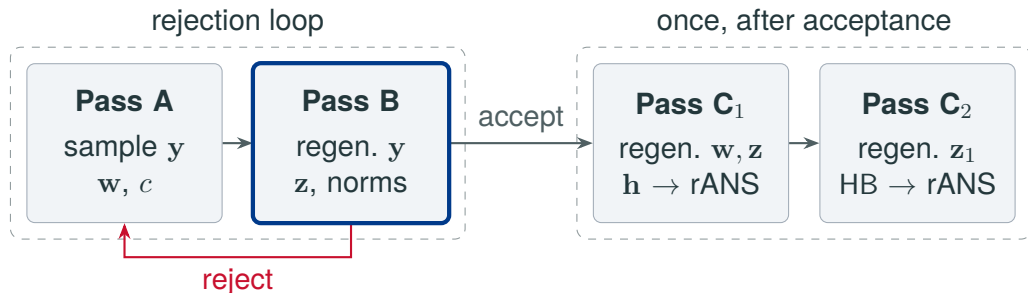


$$S_{\text{sign}} = S_{\text{driver}} + \max(S_A, S_B, S_{C_1}, S_{C_2})$$

- ▶ Between passes: retain the driver state and c ; regenerate intermediate polynomials.
- ▶ Hint computation and encoding: post-acceptance passes.

S_{driver} : the persistent driver state (seeds, nonce counter, hash prefix), live throughout.

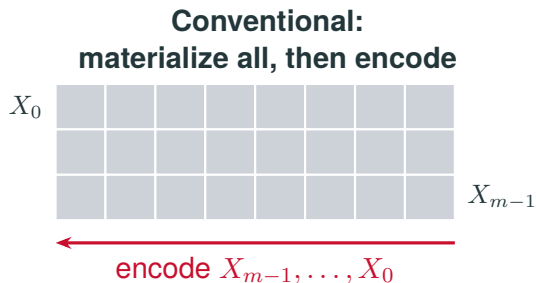
Rejection-aware pass decomposition



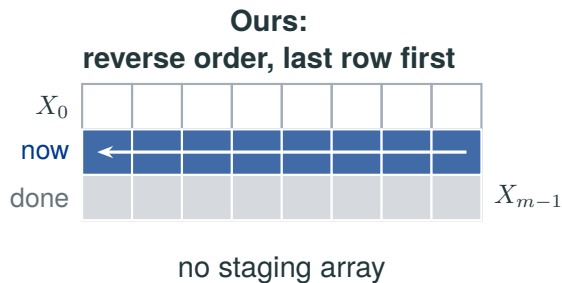
$$S_{\text{sign}} = S_{\text{driver}} + \max(S_A, S_B, S_{C_1}, S_{C_2})$$

- **Component-level early rejection** (adapted from [BRS22]):
 $z_{1,0}$ first (no secret-key material); partial $\|z\|^2 > B'^2 \Rightarrow \text{reject}$.

Reverse-order streaming entropy coding



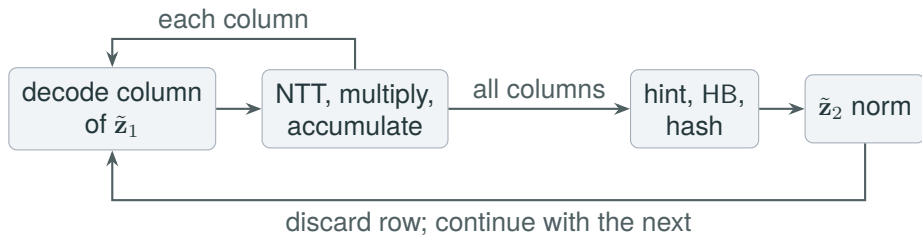
the whole array is materialized



recompute each row; encode immediately

- ▶ Same symbols in the same order $\Rightarrow$ **byte-identical encoder output**
- ▶ No full staging arrays for $\mathbf{h}$ and $\text{HB}(\mathbf{z}_1)$ (HAETA5: 4,096 and 1,792 B)

Verify one row at a time



[HCK+26]: column-streamed ours: row-streamed

kept live	all of $\tilde{w}$ (k polynomials)	one row of $\tilde{w}$
re-computation	–	re-decode $\tilde{z}_1$ and redo ℓ NTTs per row

Cortex-M4 (pqm4): Verify stack 4.7 KiB at every level.
HAETA5-5: 6,220 $\rightarrow$ 4,840 B.

Results

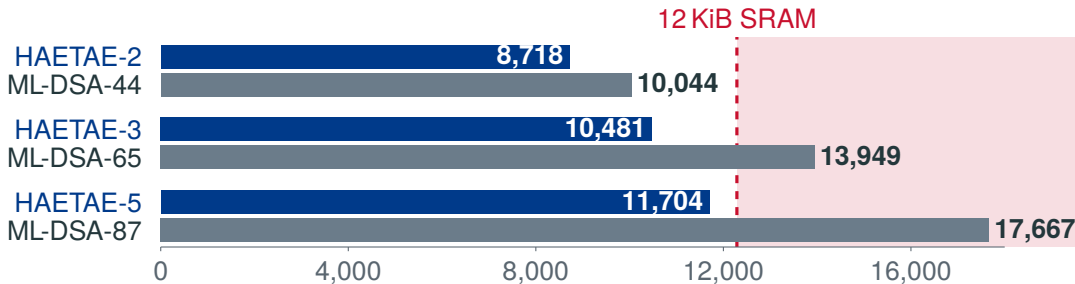
Peak stack and cycles on Cortex-M4

ours (C)	KeyGen			Sign			Verify		
	stack (B)		cycles (M)	stack (B)		cycles (M)	stack (B)		cycles (M)
vs. ref (C)									
HAETAE-2	5,816	−76%	11.3 1.22×	5,836	−92%	92.0 2.11 ×	4,824	−86%	1.1 0.63 ×
HAETAE-3	5,816	−86%	22.4 1.87×	6,020	−95%	115.6 2.16 ×	4,848	−91%	2.3 0.72 ×
vs. [HCK+26] (HAETAE-5 only)									
HAETAE-5	4,776	−8%	10.8 0.70 ×	6,004	−26%	172.8 0.57 ×	4,840	−22%	2.8 0.66 ×

- ▶ Stack: **4.7–5.9 KiB** for every operation and level (reference: 23–141 KiB).
- ▶ Sign: **2.1–2.6×** ref (C) cycles (seed-based recomputation); Verify: faster than ref (C).
- ▶ Cortex-M0+, 48 MHz, ours (C), −0s: Sign 6.6–17.7 s (measured).

Compared with low-stack ML-DSA (Dilithium)

Full signing on Cortex-M4: $sk + \sigma$ + signing stack, in bytes



- ▶ Every HAETAE level signs within 12 KiB; ML-DSA-65 and -87 do not.
- ▶ Verify (ours C): 2.99–4.25× faster than ML-DSA m4fstack.
- ▶ ML-DSA m4fstack: faster Sign, smaller code.

References

- BRS22** J. W. Bos, J. Renes, A. Sprenkels.
Dilithium for memory constrained devices. AFRICACRYPT 2022.
- HKS25** V. Hwang, Y. Kim, S. C. Seo.
Multiplying polynomials without powerful multiplication instructions.
TCHES 2025(1).
- HCK+26** Y. Hyoungh, S. Cho, U. Kim, M. Lee, H. Seo, M. Sim.
Memory-efficient implementation of SMAUG-T and HAETAE. ePrint 2026/442.
- LWKP25** S. Lee, J. Woo, J. Kim, J. H. Park.
Generalized centered binomial distribution for bimodal lattice signatures.
IEEE Access 13, 2025.

Conclusion

- ▶ Stack 4.7–5.9 KiB for every operation and level (Cortex-M4); Verify within 8 KiB (AVR), Sign within 12 KiB (Cortex-M0+).
- ▶ Cost: Sign $2.1\text{--}2.6\times$ ref (C) cycles; Verify faster than ref (C).

Peak stack: **computation order and buffer lifetimes.**

regenerate

two-pass hyperball sampler,
streamed matrix generation

non-overlapping lifetimes

rejection-aware pass decomposition

align computation order

reverse-order streaming rANS,
row-streamed verification

Pass decomposition and reverse-order streaming
can carry over to ML-DSA hint and bit-packing.



code on GitHub

Thank you!

Appendix

Appendix: correctness

Encoder (Lemma). For a fixed rANS configuration:

1. encoding is deterministic in the symbol sequence;
2. both encoders take the indices $m-1, \dots, 0$;
3. each streamed symbol: same accepted seed/nonce and signing state; same rounding/decomposition routines.

So the encoded output is byte-identical.

Two-pass sampler.

- ▶ The same (seed, nonce) regenerates the same Gaussian sequence.
- ▶ With the same fixed-point $1/\sqrt{\cdot}$ and rounding, $S, \alpha, (\mathbf{y}_1, \mathbf{y}_2)$, and the acceptance decision equal the one-pass reference.

Differential tests: 10,000 signatures per level; encoded streams matched byte for byte.

Appendix: why 32-bit coefficients?

- ▶ $q = 64,513$: 16 bits; $2q = 129,026$: more than 16 bits.
Signing: $w = \mathbf{A}[y_1] \bmod 2q$.
- ▶ Narrow types where they suffice: verification keeps $\text{HB}(z_1)$ as `int8_t` and the hint row as `uint16_t`

Appendix: measurement setup

platform	toolchain and flags	cycles
NUCLEO-L4R5ZI Cortex-M4 (pqm4) [HCK+26] row	arm-none-eabi-gcc 14.2, -O3, no LTO cycles at 20 MHz, stack at 120 MHz published: -O3 cycles, -Os stack and size	median of 100
ATmega2560 Arduino Mega	avr-gcc 15.2.0, -O3	median of 10
NUCLEO-C031C6 Cortex-M0+	-Os; Flash size with -Os -flto	median of 100

- ▶ Stack values are measured peaks, not compiler-independent bounds.
- ▶ Separate translation unit + no LTO $\Rightarrow$ no inlining into the driver.

Appendix: Cortex-M4, HAETA-E-2

	KeyGen		Sign		Verify	
	cycles	stack	cycles	stack	cycles	stack
ref (C)	9,253K	23,844	43,710K	73,112	1,732K	33,448
m4fspeed	7,109K	19,764	16,147K	55,552	916K	23,288
m4fstack	6,465K	17,204	30,472K	41,608	916K	23,288
ours (C)	11,268K	5,816	92,024K	5,836	1,098K	4,824
ours (asm)	9,881K	5,816	63,603K	5,836	852K	4,824

Appendix: Cortex-M4, HAETA-E-3

	KeyGen		Sign		Verify	
	cycles	stack	cycles	stack	cycles	stack
ref (C)	12,007K	41,236	53,546K	113,328	3,170K	54,232
m4fspeed	13,524K	29,500	24,622K	83,296	1,756K	31,768
m4fstack	15,396K	22,340	35,388K	58,008	1,756K	31,768
ours (C)	22,437K	5,816	115,568K	6,020	2,267K	4,848
ours (asm)	15,768K	5,816	120,087K	6,020	1,715K	4,848

Appendix: Cortex-M4, HAETA5-5

	KeyGen		Sign		Verify	
	cycles	stack	cycles	stack	cycles	stack
ref (C)	16,491K	52,500	65,896K	144,408	3,968K	68,856
m4fspeed	17,308K	34,108	37,381K	103,856	2,317K	37,184
m4fstack	13,359K	22,332	50,813K	69,280	2,317K	37,184
[HCK+26] (C)	15,328K	5,212	300,881K	8,092	4,187K	6,220
ours (C)	10,804K	4,776	172,765K	6,004	2,758K	4,840
ours (asm)	12,190K	4,776	166,236K	6,004	2,205K	4,840

Appendix: ML-DSA m4fstack for scale

	KeyGen		Sign		Verify	
	cycles	stack	cycles	stack	cycles	stack
ML-DSA-44	1,797K	4,408	10,330K	5,064	3,816K	2,720
ML-DSA-65	3,422K	4,408	21,668K	6,608	6,771K	2,720
ML-DSA-87	5,770K	4,408	27,150K	8,144	11,713K	2,720

- ▶ ML-DSA m4fstack signs faster and verifies with less stack
- ▶ Our verify (C): $2.99\text{--}4.25\times$ faster (HAETAE-2/3/5 vs. ML-DSA-44/65/87)
- ▶ $sk + \sigma + \text{stack}$ within 12 KiB: all HAETAE levels; not ML-DSA-65/87

Appendix: ATmega2560, Verify (8 KiB SRAM)

	impl.	σ + stack (B)	headroom (B)	time (ms)
HAETAE-2	C	6,458	1,528	1,327
	asm	6,458	1,528	1,089
HAETAE-3	C	7,355	631	2,684
	asm	7,355	631	2,154
HAETAE-5	C	7,961	25	3,351
	asm	7,956	30	2,775

16 MHz, avr-gcc 15.2.0, -O3, median of 10; *pk* in Flash.
Headroom out of 7,986 B (after the 206 B harness).

Appendix: Cortex-M0+ (12 KiB SRAM)

ours (C)	KeyGen		Sign		Verify	
	bytes	ms	bytes	ms	bytes	ms
HAETAE-2	8,512	1,379	9,064	6,643	7,560	105
HAETAE-3	9,696	2,138	10,832	17,747	8,928	216
HAETAE-5	9,912	1,548	12,056	15,179	10,128	260

Bytes: KeyGen $sk+pk+stack$, Sign $sk+\sigma+stack$, Verify $pk+\sigma+stack$.

Appendix: side-channel scope

- ▶ Constant time except the rejection loop of Fiat–Shamir with aborts
- ▶ First early exit: $z_{1,0} = y_{1,0} \pm c$; no secret-key material
- ▶ Rejection-loop early exits: outside the constant-time claim.
Side-channel analysis: future work.
- ▶ Masking and fault countermeasures: future work

Appendix: minimum-state estimates (HAETA-5)

	estimate (bytes)	[HCK+26]	ours
KeyGen	$2 \cdot 1,024 + 2,048 = 4,096$	5,212	4,776
Sign	$(k+3) \cdot 1,024 + \ell N = 8,960$	8,092	6,004
Verify	$(k+2) \cdot 1,024 = 6,144$	6,220	4,840

- ▶ Estimates count only the dominant ring objects; they are not lower bounds.