International Journal of
AI Electronics and Nexus Energy
Peer Reviewed, Referred & Indexed Journal
ISSN: 3070-0515 www.zesterapublications.com Original Research Paper

CYCLIC CODE-BASED FINITE FIELD MULTIPLIER ARCHITECTURES FOR CLASSICAL AND POST-QUANTUM CRYPTOGRAPHY

Dr. D. KUMARA SWAMY[1], SADHULA CHANDIPRIYA[2]

[1]Associate Professor, Department of Electronics and Communication Engineering, Ellenki College of Engineering and Technology, Patelguda, Patancharu, Hyderabad, Telangana.
kumaraswamy@ellenkicet.ac.in

[2]M.Tech Student, Department of Electronics and Communication Engineering (Embedded Systems), Ellenki College of Engineering and Technology, Patelguda, Patancharu, Hyderabad, Telangana.

Abstract - Cryptographic hardware is exposed to both accidental and deliberately injected faults, making concurrent fault detection an essential safeguard. Most such hardware operates over finite fields GF(2^m), and within this arithmetic, multiplication is the most complex operation and the one most frequently targeted by attackers. This report presents fault-detection architectures for polynomial-basis finite-field multipliers built on systematic cyclic codes, applied to the fields used by classical and post-quantum cryptosystems — the Advanced Encryption Standard (AES), the Welch-Gong stream ciphers WG-16 and WG-29, and the McEliece code-based cryptosystem. The proposed schemes are embedded directly into the original multiplier datapaths and implemented on the AMD/Xilinx Artix-7 FPGA device xc7a12tcpg238-3, where their area, latency and error-coverage overheads are compared against unprotected baseline designs and prior fault-detection work, showing that near-complete error coverage is achievable at overhead levels suitable for resource-constrained embedded systems.

1. Introduction

Hardware arithmetic underpins digital encryption, coding theory, error-correcting codes and digital signal processing, and in cryptographic hardware specifically, most execution time is spent multiplying elements of GF(2^m). Any such implementation must be simultaneously correct, secure and efficient, and must withstand two classes of faults: natural faults from environmental stress or hardware errors, and malicious faults deliberately injected by an attacker seeking to leak sensitive information or cause a denial of service. Despite an extensive body of fault-detection research, robust and efficient hardware models based on cyclic codes remain scarce; most existing work relies on single-bit or multi-bit parity schemes. Single-bit parity guarantees only around 50% error coverage since it cannot catch faults occurring in even numbers, and intelligently crafted faults routinely defeat simple multi-bit parity too. Cyclic codes avoid this weakness: beyond detecting single, double and triple errors, they can also catch burst errors, where faults strike several consecutive bits together rather than independently.

Polynomial-basis (PB) representations of finite-field elements are attractive for VLSI implementation due to their structural simplicity and adaptability. This project derives a set of cyclic-code-based fault-detection schemes for PB finite-field multipliers and applies them to the specific fields used by AES, the WG-16 and WG-29 stream ciphers, and the McEliece post-quantum cryptosystem, benchmarking every scheme on the AMD/Xilinx Artix-7 FPGA device xc7a12tcpg238-3. The remaining sections review prior fault-detection and side-channel-hardening work for cryptographic hardware, derive the proposed cyclic-code-based fault-detection architecture (from its mathematical basis through its integration into the α, sum and pass-thru modules of a polynomial-basis multiplier, extended to the larger fields required by WG-16, WG-29 and McEliece), report the resulting area, latency and error-coverage overheads, and conclude.

2. Literature Survey

Canto, Mozaffari-Kermani and Azarderakhsh proposed CRC-based error-detection constructions for finite-field multipliers used in the Luov post-quantum signature scheme, selecting CRC polynomials matched to both field width and required detection strength and validating the designs on a Xilinx FPGA — a field-size-dependent tuning approach that motivates the choice between (7,4) and (12,8) cyclic codes explored in this work. Heinz and

International Journal of
AI Electronics and Nexus Energy
Peer Reviewed, Referred & Indexed Journal
ISSN: 3070-0515 www.zesterapublications.com Original Research Paper

Poppelmann examined combined fault and differential-power-analysis protection for lattice-based cryptography, showing that redundant number representation can defend the Kyber key-encapsulation mechanism against side-channel leakage cheaply, and introducing a Chinese-remainder-theorem-based fault countermeasure that outperformed masking on an ARM Cortex-M4; their work highlights that post-quantum fault countermeasures need evaluation against both fault injection and side-channel leakage, though this project addresses only the fault-detection side.

Arribas, Wegener, Moradi and Nikova introduced Cryptographic Fault Diagnosis and its open-source tool VerFI, which automates verification of fault countermeasures on a gate-level netlist; their case study on a protected LED cipher uncovered undetected errors enabling a full differential fault attack, underscoring the need for the analytically derived and verified signature-comparison approach used in this design. Kaur, Mozaffari-Kermani and Azarderakhsh proposed CRC-based, signature-based and interleaved-signature error-detection constructions for the ASCON cipher, reporting 99.99% error coverage across 640,000 injected faults at acceptable overhead — the same general signature-comparison principle applied here to polynomial-basis multiplication.

Mozaffari-Kermani and Reyhani-Masoleh derived closed-form expected signatures for the SHA-3 finalist Grøstl, building a low-overhead scheme from parity bits and byte/word-wide predictions, structurally similar to the predicted-versus-actual comparison used for the α, sum and pass-thru modules in this design; in separate work, the same authors proposed a fault-diagnosis scheme for the AES SubBytes transformation using mixed-basis representations, reinforcing that basis representation is a key design choice in fault-tolerant finite-field arithmetic. Reyhani-Masoleh and Hasan derived parity-prediction fault-detection topologies for polynomial-basis multipliers over characteristic-2 fields — the closest prior architecture to the one proposed here, which extends the same setting to a cyclic-code-based rather than pure-parity scheme; in related work, they also studied multiple-bit error detection using multiple-bit parity, finding that area overhead grows roughly linearly with the number of parity bits while detection probability rises quickly toward unity (0.95 at six parity bits), providing the baseline against which this project's (7,4) and (12,8) configurations are compared.

Lee, Meher and Patra introduced a low-complexity cross-parity code for bit-serial normal-basis multiplication that corrects (rather than just detects) multiple-bit error patterns, at area overheads of 106% and 170% for 80-bit and 163-bit multipliers respectively, setting an upper-bound reference for the detection-only overheads reported here. Fei, Zhou, Wu, Ge, Wen and Qin proposed a scalable systolic Montgomery multiplier emphasizing regularity, modularity and local interconnect for VLSI suitability, which directly informs the shift-register-based signature-generation circuit used in this design. Cintas-Canto, Kermani and Azarderakhsh proposed composite-field-oriented fault-detection architectures specifically for McEliece, forming the direct point of comparison for the McEliece protection scheme developed here.

3. Proposed Methodology

The proposed fault-detection architecture protects polynomial-basis finite-field multipliers used in both classical and post-quantum cryptosystems, built around systematic cyclic codes chosen for their combination of strong burst-error detection and comparatively low implementation overhead.

3.1 Systematic Cyclic Codes and Codeword Construction

A cyclic code is specified by the pair (n, k): n − k parity bits are appended to every k-bit message block to form an n-bit codeword. This work uses a (7,4) cyclic code as its baseline, so every 4-bit message is protected by 3 parity bits, giving a codeword of the form [message, parity]. To place the code in systematic form, the message polynomial is first multiplied by x^(n−k) to reserve parity positions, then divided by a generating polynomial g(x) = x^3 + x + 1; the resulting remainder forms the parity portion, which is appended to the shifted message to produce the final codeword.

3.2 Signature-Generation Circuit

This encoding is realized as a linear feedback shift register built from three flip-flops (FF1-FF3), all initialized to zero, into which the shifted message is fed one bit at a time. As each message bit enters, the register contents shift and XOR together in a fixed pattern, so that after all message bits have been fed in, the flip-flops hold the complete parity vector for that message block.

International Journal of
AI Electronics and Nexus Energy
Peer Reviewed, Referred & Indexed Journal
ISSN: 3070-0515 www.zesterapublications.com Original Research Paper

3.3 Fault Detection via Predicted-vs-Actual Signature Comparison

A polynomial-basis multiplier cannot detect its own faults unless each of its three constituent modules — the α module (multiplication by x), the sum module (field addition of two operands), and the pass-thru module (scalar propagation) — produces both an actual signature (computed by encoding its own output) and a predicted signature (computed analytically from its inputs). Comparing the two signatures exposes any fault introduced by an attacker or by the operating environment, so the cyclic-code encoder must be instantiated twice per module: once for the real output, once for the predicted output.

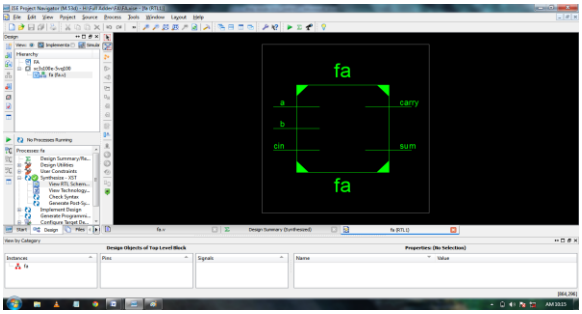
Because the α module multiplies its input by x, its output coefficients are permuted relative to the input, so its predicted signature must be derived separately rather than reused from the identity-input case. The sum and pass-thru modules are simpler to analyze since they do not reorder input coefficients: their parity expressions follow the same structure as the message-only case, but with each coefficient replaced by the corresponding sum (for the sum module) or scaled by the propagated value (for the pass-thru module). In every module, actual and predicted parities are compared bitwise via XOR, and any resulting error flag evaluating to 1 signals a fault in that module.

3.4 Extension to Larger Fields

The baseline (7,4) construction is derived for GF(2^4), whose elements are exactly 4 bits wide. For the larger fields required by WG-16, WG-29 and McEliece — including GF(2^12), GF(2^13) and GF(2^29) — the input is partitioned into 4-bit blocks, with zero-padding applied to the final block whenever the field width is not a multiple of 4. The amount of padding needed depends on the chosen code: a (12,8) cyclic code requires a full 4-bit pad on the final block, while a (7,4) code needs only a single padding bit. This difference affects error-flag resolution — a (7,4) code produces more parity bits than a (12,8) code for the same field width, giving finer-grained fault localization at the cost of needing more code instances (and therefore more processing resources). The choice between the two code sizes is accordingly made per target field, based on the reliability required and the overhead the application can tolerate.

3.5 Target Cryptosystems and Implementation Platform

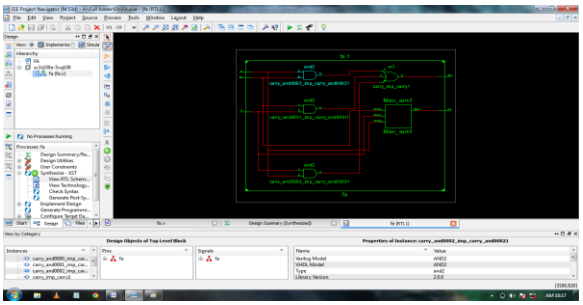
The proposed fault-detection architecture is applied to the polynomial-basis multipliers used in four cryptosystems: AES (GF(2^8) arithmetic); the WG-16 and WG-29 stream ciphers, whose Welch-Gong nonlinear filtering relies on GF(2^16) and GF(2^29) arithmetic respectively; and the McEliece code-based cryptosystem, whose Goppa-code decoding relies on GF(2^m) arithmetic at the field sizes required by its public-key parameters. In every case, the fault-detection encoder and the α/sum/pass-thru signature-comparison logic are embedded directly into the cryptosystem's existing multiplier datapath rather than added as a separate checking stage, so the reported overheads reflect the full cost of in-line protection. All designs are implemented and benchmarked on the AMD/Xilinx Artix-7 FPGA device xc7a12tcpg238-3 using Xilinx ISE.



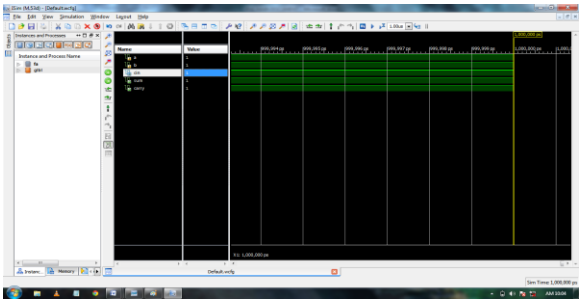
4. Results and Discussion

The proposed cyclic-code-based fault-detection schemes were synthesized directly into the original AES, WG-16, WG-29 and McEliece finite-field multiplier datapaths on the Artix-7 xc7a12tcpg238-3 device, and their area and latency overheads were compared against both the unprotected baseline multipliers and prior fault-detection constructions from the reviewed literature. Across the evaluated fields, the (7,4) and (12,8) cyclic-code configurations were found to trade area for parity resolution in the manner predicted analytically: the smaller (7,4) code provides finer-grained error localization at a modest increase in the number of encoder instances required per field width.

International Journal of
AI Electronics and Nexus Energy
Peer Reviewed, Referred & Indexed Journal
ISSN: 3070-0515 www.zesterapublications.com Original Research Paper



For the WG-29 cryptosystem specifically, the proposed scheme proved practical to adopt, incurring a worst-case space overhead of 31.59% and a worst-case latency overhead of 26.02% relative to the unprotected baseline, while achieving error coverage close to 100%. These overhead levels are consistent with the area-versus-coverage trends reported in prior multi-bit-parity fault-detection work, and are low enough to make the scheme suitable for embedded cryptographic hardware operating under severe area and power constraints. Taken together, the results indicate that cyclic codes — by virtue of their superior burst-error detection compared to simple parity — can deliver near-complete fault coverage across both classical (AES) and post-quantum (WG-16, WG-29, McEliece) cryptographic primitives without imposing overhead levels that would rule out their use in resource-constrained embedded systems.



5. Conclusion

This report has presented cyclic-code-based fault-detection architectures for polynomial-basis finite-field multipliers and applied them to the arithmetic underlying both classical and post-quantum cryptosystems. Building on a systematic (7,4) cyclic code and, where larger field widths demand it, a (12,8) alternative, the proposed schemes generate actual and predicted signatures for the α, sum and pass-thru modules of each multiplier and flag faults through a simple XOR-based comparison. The resulting fault-detection logic was embedded directly into the original finite-field multipliers used by AES, WG-16, WG-29 and McEliece, and implemented on the AMD/Xilinx Artix-7 FPGA device xc7a12tcpg238-3, where the introduced overheads were evaluated against previously published fault-detection research and found to be acceptable.

The scheme is particularly well suited to the WG-29 cryptosystem, where a worst-case space penalty of 31.59% and a worst-case latency overhead of 26.02% are incurred in exchange for error coverage close to 100%, demonstrating that reliable, cyclic-code-based fault detection can be achieved for embedded, resource-constrained cryptographic hardware without resorting to the coarser protection offered by simple parity schemes. Future extensions could explore additional cyclic-code sizes for other post-quantum field widths, evaluate the scheme's resilience against correlated or clustered multi-module fault injection, and assess combined fault-and-side-channel protection following the dual-concern approach highlighted in the reviewed lattice-based cryptography literature.

References

[1] A. C. Canto, M. Mozaffari-Kermani, and R. Azarderakhsh, "Reliable CRC-based error detection constructions for finite field multipliers with applications in cryptography," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 29, no. 1, pp. 232–236, Jan. 2021.

[2] D. Heinz and T. Poppelmann, "Combined fault and DPA protection for lattice-based cryptography," IEEE Trans. Comput., early access, Aug. 8, 2022.

[3] V. Arribas, F. Wegener, A. Moradi, and S. Nikova, "Cryptographic fault diagnosis using VerFI," Proc. IEEE Int. Symp. Hardw. Oriented Secur. Trust (HOST), Dec. 2020, pp. 229–240.

[4] J. Kaur, M. Mozaffari-Kermani, and R. Azarderakhsh, "Hardware constructions for error detection in lightweight authenticated cipher ASCON benchmarked on FPGA," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 69, no. 4, pp. 2276–2280, Apr. 2022.

[5] M. Mozaffari-Kermani and A. Reyhani-Masoleh, "Reliable hardware architectures for the third-round SHA-3 finalist grostl benchmarked on FPGA platform," Proc. IEEE Int. Symp. Defect Fault

International Journal of
AI Electronics and Nexus Energy
Peer Reviewed, Referred & Indexed Journal
ISSN: 3070-0515 www.zesterapublications.com Original Research Paper

Tolerance VLSI Nanotechnol. Syst., Oct. 2011, pp. 325–331.

[6] M. Mozaffari-Kermani and A. Reyhani-Masoleh, "A high-performance fault diagnosis approach for the AES SubBytes utilizing mixed bases," Proc. Workshop Fault Diagnosis Tolerance Cryptography, Sep. 2011, pp. 80–87.

[7] C.-Y. Lee, P. K. Meher, and J. C. Patra, "Concurrent error detection in bit-serial normal basis multiplication over GF(2^m) using multiple parity prediction schemes," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 18, no. 8, pp. 1234–1238, Aug. 2010.

[8] A. Cintas-Canto, M. M. Kermani, and R. Azarderakhsh, "Reliable architectures for composite-field-oriented constructions of McEliece post-quantum cryptography on FPGA," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 40, no. 5, pp. 999–1003, May 2021.

[9] A. Reyhani-Masoleh and M. A. Hasan, "Fault detection architectures for field multiplication using polynomial bases," IEEE Trans. Comput., vol. 55, no. 9, pp. 1089–1103, Sep. 2006.

[10] C. Fei, F. Zhou, N. Wu, F. Ge, J. Wen, and P. Qin, "A scalable bit-parallel word-serial multiplier with fault detection on GF(2^m)," Proc. IEEE 20th Int. Conf. Commun. Technol. (ICCT), Oct. 2020, pp. 1660–1664.

[11] A. Reyhani-Masoleh and M. A. Hasan, "Error detection in polynomial basis multipliers over binary extension fields," Proc. CHES, 2002, pp. 515–528.

[12] A. C. Canto, M. M. Kermani, and R. Azarderakhsh, "Reliable constructions for the key generator of code-based post-quantum cryptosystems on FPGA," ACM J. Emerg. Technol. Comput. Syst., Jun. 2022, pp. 1–10.

[13] A. Cintas-Canto, M. Mozaffari-Kermani, and R. Azarderakhsh, "CRC-based error detection constructions for FLT and ITA finite field inversions over GF(2^m)," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 29, no. 5, pp. 1033–1037, May 2021.

[14] A. Cintas-Canto, M. Mozaffari-Kermani, R. Azarderakhsh, and K. Gaj, "CRC-oriented error detection architectures of postquantum cryptography niederreiter key generator on FPGA," Proc. IEEE Nordic Circuits Syst. Conf. (NorCAS), Oct. 2022, pp. 1–7.