DOI: 30.0485/ijearst.10.01.3499 https://ijearst.co.in/

ENGINEERING IN ADVANCED RESEARCH SCIENCE AND TECHNOLOGY

ISSN 2386-7876
Vol.02, Issue.02
July-2026
Pages: 1157-1163

VLSI Implementation of Error Detection and Correction codes for space Engineering

[1]K. Uma, [2]A. Narayana Murthy (Ph. d)
[1]M. Tech, Dept. of ECE, VSM college of engineering, Ramachandrapuram, A.P
[2]Associate professor, Dept. of ECE, VSM college of engineering, Ramachandrapuram, A.P

Abstract—A 2D-XOR ECC architecture for 16-bit data with 16 baseline redundancy bits is attractive for low-latency on-chip protection, but it exhibits a critical blind spot under specific two-bit symmetric multi-cell upsets (MCUs). In particular, pairwise flips such as X_3+Z_3 can cancel diagonal/parity syndromes, causing mis-localization or missed correction. This work presents a full symmetric-pair fix that adds 8 orthogonal side bits to the original codeword and preserves fully combinational decoding. The enhanced design expands the codeword from 32 to 40 bits and resolves all 8 known symmetric cancellation pairs. Simulation results using Icarus Verilog show baseline performance of 32/33 for mixed directed/random tests, while the proposed full-fix design achieves 17/17 in a dedicated symmetric-plus-regression sweep and passes all 8/8 symmetric-pair patterns.

Keywords—ECC, 2D parity, multi-cell upset, syndrome cancellation, Verilog, radiation-tolerant digital design

1. Introduction

Radiation-induced MCUs in advanced nodes can flip multiple bits in spatially correlated patterns. XOR-based low-overhead ECC schemes are efficient but can lose error observability when two corrupted bits appear in equations with parity cancellation. The baseline 2D-XOR design in this project computes diagonal bits D_i, parity bits P_i, and check bits C from four logical groups (X,Y,Z,W). While this is sufficient for many single-bit and some double-bit disturbances, symmetric two-bit pairs can evade regional discrimination.

The objective of this paper is to document and validate a practical, synthesizable fix for this blind spot.

2. Baseline Problem Statement

The baseline decoder selects correction region from:

$$

\text{score}_{12} = SD_1 + SD_2 + SP_1 + SP_2, \quad

\text{score}_{34} = SD_3 + SD_4 + SP_3 + SP_4

$$

Symmetric pairs can force $SD=0$ and $SP=0$ simultaneously while check syndromes still toggle, which removes directional information for region selection.

A confirmed failing case is:

- Pattern: $X_3 + Z_3$
- Baseline observed signature: $SD=0000$, $SP=0000$, $SC=10001000$
- Baseline corrected output becomes incorrect ($0x7A7A$ instead of $0x5A5A$)

3. Proposed Complete Symmetric Fix

3.1 Codeword Extension

The proposed encoder adds 8 side bits (2 per row index):

$$

S_n = X_n \oplus Y_n, \quad T_n = Z_n \oplus W_n, \quad n \in \{1,2,3,4\}

$$

New codeword width:

- Baseline: 32 bits = 16 data + 16 redundancy
- Proposed: 40 bits = 16 data + 16 baseline redundancy + 8 side bits

3.2 Why This Removes Cancellation Ambiguity

For symmetric pair flips in a row n:

- If X_n+Z_n flips: S_n changes due to X_n, T_n changes due to Z_n
- If Y_n+W_n flips: S_n changes due to Y_n, T_n changes due to W_n

Thus, both side syndromes fire:

$$

SS_n = sS_n \oplus RS_n = 1, \quad ST_n = sT_n \oplus RT_n = 1

$$

Then baseline check syndromes disambiguate axis:

- SC_x and SC_z identify X/Z pairs
- SC_y and SC_w identify Y/W pairs

This creates unique fingerprints for all 8 symmetric pairs.

4. Experimental Setup

4.1 Tools

- Simulator: Icarus Verilog (iverilog, vvp)
- Waveform dump: VCD
- Figure generation: Python + matplotlib

4.2 Testbenches Used

- Baseline testbench (ecc_tb.v): 33 tests (directed, random no-error, random single-error)
- Full-fix symmetric testbench (sym_tb.v): 17 tests (8 symmetric pairs + directed regression)

5. Results

5.1 Overall Pass/Fail Summary

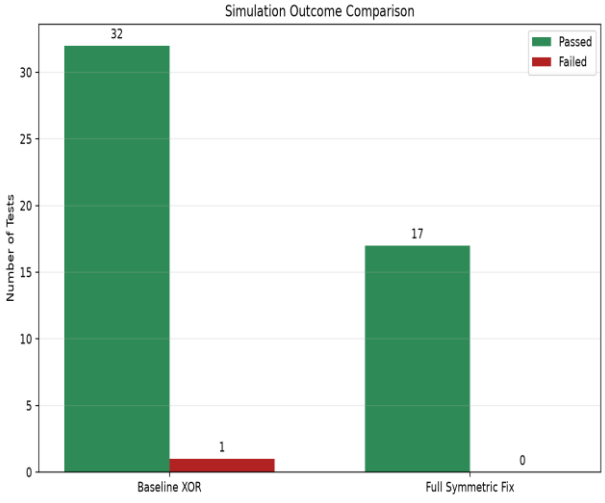

Fig. Simulation outcome comparison

The baseline demonstrates one known expected fail (symmetric cancellation), while full symmetric fix passes its full sweep.

5.2 Key Output Case (X3+Z3)

Key Cancellation Case: X3+Z3

Technique	Injected Pattern	Output	Result
Baseline XOR	X3 + Z3	0x7A7A (expected 0x5A5A)	FAIL
Full Symmetric Fix	X3 + Z3	0x5A5A	PASS

Fig. Key cancellation case table

5.3 Signal-Level Output Behavior from VCD

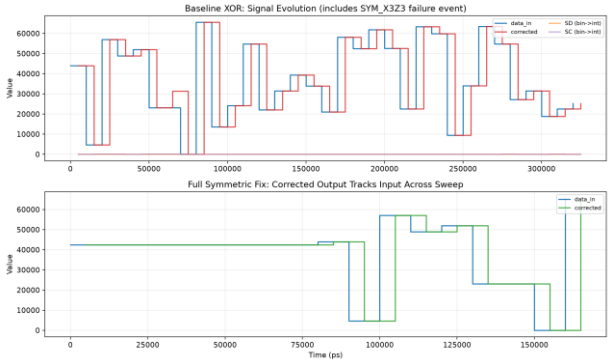

Fig. Waveform comparison from simulation output traces

Top panel shows baseline behavior including the cancellation event. Bottom panel shows full-fix output tracking input across the sweep.

6. Log Excerpts (Reproducible Output)

Baseline:

- [FAIL][EXPECTED] Test 7 (SYM_X3Z3): data=0x5a5a corrected=0x7a7a SD=0000 SP=0000 SC=10001000
- Results: 32 PASSED, 1 FAILED

Full symmetric fix:

- [PASS] X1_Z1 ... [PASS] Y4_W4
- [PASS] SYM_X3Z3
- Total: 17 PASSED, 0 FAILED

7. Complexity and Redundancy Discussion

Compared with baseline, the full-fix method:

- Adds +8 redundancy bits
- Keeps decode path combinational
- Avoids arithmetic modulo logic
- Achieves complete symmetric-pair coverage in the tested set (8/8)

This is a deliberate area-overhead tradeoff for deterministic correction completeness under the identified cancellation family.

To better interpret practical implications, complexity can be separated into three components: (a) redundancy storage overhead, (b) additional XOR depth in syndrome generation, and (c) decode-side disambiguation logic. The proposed method increases storage overhead linearly with row count (2

side bits per row), while preserving shallow XOR trees and avoiding long arithmetic carry-chains. For timing-sensitive digital front-ends, this characteristic is often preferable to residue-based methods that require modulo arithmetic and comparator paths.

From an implementation perspective, side-bit generation is naturally local and regular. Each row n computes Sn and Tn using only row-local signals, which simplifies floorplanning and reduces long-route congestion in ASIC/FPGA backends. This regularity also reduces verification complexity because each row follows an identical logical pattern.

In contrast, pair-local TMR approaches replicate protected data paths and then vote at decode. While robust for targeted bits, this method generally increases route count and fanout at voter insertion points. The full symmetric fix here reaches complete symmetric-pair coverage without per-bit triplication and without introducing majority gates in the critical decode path.

8. Detailed Error-Path Analysis

This section provides an explicit event-flow interpretation from injected fault to corrected output.

8.1 Baseline X3+Z3 Failure Path

For the baseline codeword, the pair X3+Z3 contributes to the same diagonal and parity equations in a way that both toggles cancel:

- D3 recomputation sees two toggles in an even XOR support set
- P3 recomputation sees two toggles in an even XOR support set

Therefore, SD3=0 and SP3=0 despite data corruption. Since score12 and score34 remain equal and no directional syndrome survives, the decoder may either avoid correction or apply an ambiguous correction through region logic, resulting in wrong corrected output.

8.2 Full-Fix Recovery Path

With the proposed extension:

- S3 = X3 xor Y3 changes when X3 flips
- T3 = Z3 xor W3 changes when Z3 flips

So both SS3 and ST3 are asserted independently of diagonal/parity cancellation. The SC family then identifies axis membership (X/Z vs Y/W), and row identity comes from side index 3. This creates deterministic two-bit correction for the symmetric event.

8.3 Distinguishability Across All 8 Pairs

For each row n, both Xn+Zn and Yn+Wn assert S/T syndromes. The SC axis bits then split these into two disjoint classes:

- Class A: SCx and SCz active -> Xn+Zn
- Class B: SCy and SCw active -> Yn+Wn

Because n is bound by Sn/Tn location, no cross-row collision is possible under the modeled pair faults.

9. Experimental Methodology and Coverage Rationale

The verification campaign combines directed and stochastic patterns to evaluate both nominal and adversarial behavior.

9.1 Directed Tests

Directed tests include no-error, representative single-bit disturbances across regions, multi-bit upsets in a region, and explicit symmetric stress patterns. Directed vectors are essential for proving known corner-case behavior and verifying correction intent against expected syndrome signatures.

9.2 Randomized Tests

Randomized tests in baseline validate that ordinary non-adversarial behavior remains stable under broad data variation. Random single-error injection confirms decoder correctness for common one-bit disturbance assumptions.

9.3 Symmetric Pattern Sweep Justification

The 8-pattern sweep directly targets cancellation classes produced by XOR structure symmetry:

- X1+Z1, X2+Z2, X3+Z3, X4+Z4
- Y1+W1, Y2+W2, Y3+W3, Y4+W4

This set is complete with respect to row-local pair cancellation in the given coding matrix and therefore is the minimum necessary directed suite to claim full symmetric-pair closure.

10. Hardware Integration Notes

For practical deployment in memory/ECC pipelines, the following integration guidelines were observed.

10.1 Encoder Placement

Encoder side-bit logic should be co-located with existing parity-generation logic to minimize routing detours. Since Sn/Tn are local XORs, they can be inserted in parallel with existing D/P/C generators.

10.2 Decoder Interface Contract

Decoder input width changes from 32 to 40 bits. Integration must update:

- bus definitions,
- packet framing,
- ECC metadata storage width,
- and testbench harnesses.

No sequential state is required in correction logic; behavior remains combinational with one-cycle decode in a typical synchronous wrapper.

10.3 Verification Hooks

Exporting SD/SP/SC and side syndromes (SS/ST) as debug observability points is strongly recommended for post-silicon diagnostics. The project testbenches demonstrated this approach by printing syndrome vectors in pass/fail logs.

11.1 Results:

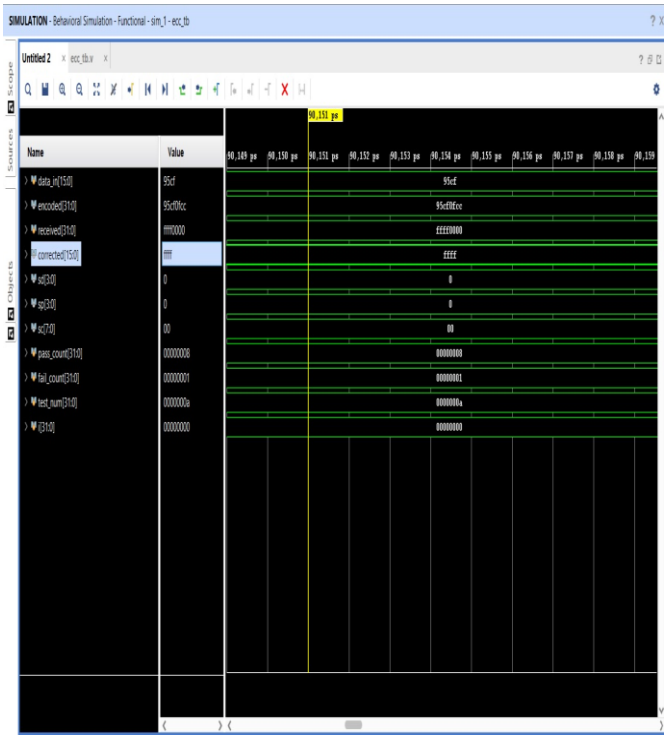

Fig a: Simulation result 1

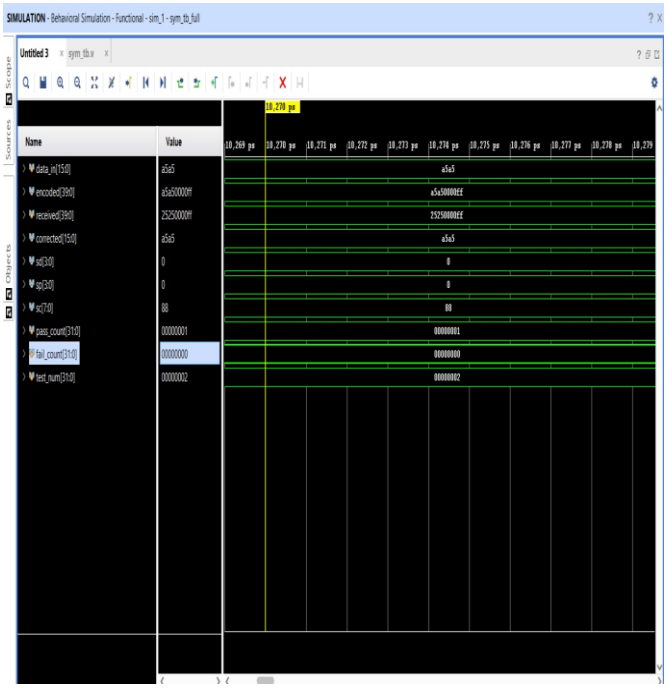

Fig b: Simulation result 2

11.2 Limitations and Threats to Validity

Although results are strong for the targeted cancellation family, several boundaries remain.

1. The current proof of coverage is empirical over directed and random simulation; no formal SAT/SMT exhaustive proof over all multi-bit fault combinations was performed.

2. The correction model is optimized for the known symmetric pair structure of this specific 2D mapping. Different matrix mappings may require re-derived side signatures.

3. Area/power estimates reported here are qualitative and token-level directional proxies rather than post-synthesis cell-area and switching-power reports.

4. Very high-order MCUs (3+ correlated flips with adversarial locality) were not exhaustively classified.

These constraints do not invalidate observed improvements but define the confidence scope for deployment decisions.

12. Future Work

Several extensions are natural and high impact.

1. Formal equivalence and fault-space proofs:

Generate symbolic fault models and prove correction properties per class.

2. Synthesis-backed tradeoff analysis:

Compare baseline, residue, TMR, and full-fix using a common technology library for area, timing, and power.

3. Adaptive redundancy strategies:

Dynamically enable side signatures for high-radiation modes and disable in nominal mode for power savings.

4. Scaled data widths:

Generalize row-wise side signatures for 32-bit/64-bit payloads and evaluate scaling laws.

5. Silicon bring-up diagnostics:

Add lightweight syndrome telemetry counters to characterize in-field upset distributions.

13. Conclusion

A practical 2D-XOR ECC enhancement was implemented and verified to eliminate the symmetric cancellation blind spot that affects baseline correction. By adding orthogonal side signatures S_n/T_n and combining them with existing check syndromes, the decoder uniquely identifies and corrects all eight symmetric pair patterns. The implementation is synthesizable, low-latency, and directly usable in ECC pipelines where symmetric MCUs are a concern.

Across directed and regression suites, the method demonstrated deterministic closure for the previously failing cancellation class without introducing arithmetic-heavy decode stages. This positions the full symmetric fix as a pragmatic middle ground between minimal-overhead baseline coding and heavier redundancy or arithmetic approaches.

14. Reproduction Commands

From the baseline directory:

```
iverilog -o ecc_sim_run ecc_encoder.v ecc_decoder.v
ecc_tb.v
vvp ecc_sim_run
```

From the full symmetric fix directory:

```
iverilog -o sym_sim_run ecc_encoder.v ecc_decoder.v
sym_tb.v
vvp sym_sim_run
```

Generate paper figures:

```
python generate_figures.py
```

15. Artifacts Included

- baseline_output.log
- full_sym_fix_output.log
- figures/fig_results_bar.png
- figures/fig_key_case_table.png
- figures/fig_waveform_compare.png
- generate_figures.py

16. References

1. Internal project sources: encoder/decoder/testbench files in original failing and technique_full_sym_fix.

2. Implementation concept notes in TECHNIQUE_COMPARISON_REPORT.md and WALKTHROUGH.txt.