A Reversible Multi-Radix Computational Cell for Hamiltonian Symbolic Processing in the RHEA-UCM Framework

Paul M. Roe
TecKnows, Inc.
Lewiston, ME, USA
ORCID: 0009-0005-6159-947X
tecknows.maine@gmail.com

Abstract

We introduce a reversible multi-radix computational gate designed for integration into the RHEA-UCM (Recursive Homeostatic Evolutionary Algorithm – Universal Cell Model) framework. The construction provides a single hardware-level cell that dynamically supports binary, ternary, and pentary symbolic arithmetic under a unified reversible mapping. In binary mode the cell behaves as a standard irreversible CMOS-compatible gate for drop-in usability; in higher-radix modes it implements strictly bijective operators over finite state spaces, enabling zero-erasure computation consistent with Hamiltonian dynamics and Landauer–Bennett thermodynamic bounds.

We present: (1) the conceptual RHEA-UCM gate behavior, (2) formal reversible mappings on $\mathbb{Z}_3 \times \mathbb{Z}_3 \times \mathbb{Z}_5$ and $\mathbb{Z}_5^3$, (3) proofs of bijectivity using triangular embeddings, (4) a behavioral Verilog specification suitable for synthesis and hardware exploration, and (5) a Python-based reversibility checker for exhaustive verification. This cell forms a concrete substrate for future RHEA-IC designs, enabling Hamiltonian, measure-preserving symbolic transformations atop CMOS-compatible reversible primitives such as Z.E.L.-class devices.

1 Introduction

Reversible computation provides the only thermodynamically admissible route toward large-scale, ultra-low-dissipation digital systems. Following Landauer, the destruction of information is the sole operation with a fundamental energy cost. Bennett showed that logically reversible mappings incur no such penalty.

In the RHEA-UCM framework, symbolic recursion rules are embedded into divergence-free, measure-preserving Hamiltonian flows. Irreversible events must therefore be isolated, scheduled, or clustered using Lorenz-type entropy modulators.

A missing component has been a unified computational cell that simultaneously:

- behaves as a CMOS-compatible binary gate,
- supports reversible ternary arithmetic over $\mathbb{Z}_3$,
- supports reversible pentary arithmetic over $\mathbb{Z}_5$,

- maintains a local symbolic/entropy register, and
- guarantees strict bijectivity in all non-binary modes.

The present work introduces such a cell.

2 A Multi-Radix Reversible RHEA-UCM Cell

We define a computational cell with inputs A, B, an internal register G, and a mode selector $M \in \{00, 01, 10\}$ governing binary, ternary, or pentary operation:

Mode M	Domain	Behavior
00	$\{0, 1\}$	Binary irreversible (e.g., NAND)
01	$\mathbb{Z}_3 \times \mathbb{Z}_3 \times \mathbb{Z}_5$	Fully reversible
10	$\mathbb{Z}_5^3$	Fully reversible

Binary mode ensures drop-in CMOS compatibility. The higher-radix modes implement Hamiltonian-like reversible transformations suitable for symbolic recursion, entropy tracking, and glyphic modulation.

3 Reversible Mappings

3.1 Ternary Core on $\mathbb{Z}_3 \times \mathbb{Z}_3 \times \mathbb{Z}_5$

Let

$$A, B \in \{0, 1, 2\}, \qquad G \in \{0, 1, 2, 3, 4\}.$$

Define the forward map:

$$A' = A, \tag{1}$$
$$B' = (B + A) \bmod 3, \tag{2}$$
$$G' = (G + B) \bmod 5. \tag{3}$$

Invertibility. Given (A', B', G'):

$$A = A', \tag{4}$$
$$B = (B' - A') \bmod 3, \tag{5}$$
$$G = (G' - B) \bmod 5. \tag{6}$$

Because the system is triangular, bijectivity is immediate. Thus the map is a permutation of the 45-state domain.

3.2 Pentary Core on $\mathbb{Z}_5^3$

Let

$$A, B, G \in \{0, 1, 2, 3, 4\}.$$

Forward:

$$\begin{aligned} A' &= A, && (7) \\ B' &= (B + A) \bmod 5, && (8) \\ G' &= (G + B) \bmod 5. && (9) \end{aligned}$$

Inverse:

$$\begin{aligned} A &= A', && (10) \\ B &= (B' - A') \bmod 5, && (11) \\ G &= (G' - B) \bmod 5. && (12) \end{aligned}$$

Again the triangular structure guarantees bijectivity.

4 Hamiltonian Interpretation within RHEA-UCM

The ternary/pentary reversible mappings serve as discrete analogues of:

- divergence-free vector fields,
- symplectic transformations,
- measure-preserving Hamiltonian flows.

In the RHEA-UCM architecture:

- A, B encode symbolic operands (the Ψ and Φ channels),
- G encodes local glyph/entropy/trust phase,
- the reversible core enforces zero-erasure dynamics,
- the Lorenz scheduler determines when binary mode may be entered to cluster entropy.

Thus the reversible cell forms a discrete Hamiltonian unit in the UCM sense.

5 Behavioral Verilog Specification

```
// ==========================================================
//  RHEA-UCM Reversible Gate (Binary / Ternary / Pentary)
// ==========================================================
module rhea_reversible_gate #(
    parameter DATA_WIDTH_BIN = 1
)(
```

```verilog
    input  wire [1:0] mode,   // 00=binary, 01=ternary, 10=pentary
    input  wire [2:0] A_in,
    input  wire [2:0] B_in,
    input  wire [2:0] G_in,
    output reg  [2:0] A_out,
    output reg  [2:0] B_out,
    output reg  [2:0] G_out
);

// mod-3 arithmetic
function [1:0] add_mod3;
    input [1:0] x, y;
    reg [2:0] s;
begin
    s = x + y;
    add_mod3 = (s >= 3) ? s - 3 : s[1:0];
end
endfunction

// mod-5 arithmetic
function [2:0] add_mod5;
    input [2:0] x, y;
    reg [3:0] s;
begin
    s = x + y;
    add_mod5 = (s >= 5) ? s - 5 : s[2:0];
end
endfunction

always @* begin
    case (mode)
        2'b00: begin
            A_out = {2'b00, ~(A_in[0] & B_in[0])};
            B_out = 3'b000;
            G_out = G_in;
        end
        2'b01: begin
            A_out = {1'b0, A_in[1:0]};
            B_out = {1'b0, add_mod3(B_in[1:0], A_in[1:0])};
            G_out = add_mod5(G_in, {1'b0, B_in[1:0]});
        end
        2'b10: begin
            A_out = A_in;
            B_out = add_mod5(B_in, A_in);
            G_out = add_mod5(G_in, B_in);
        end
        default: begin
            A_out = A_in;
            B_out = B_in;
            G_out = G_in;
        end
```

```
    endcase
end

endmodule
```

6 Python Reversibility Checker

```python
def ternary_step(A, B, G):
    return A, (B + A) % 3, (G + B) % 5

def pentary_step(A, B, G):
    return A, (B + A) % 5, (G + B) % 5

def check_bijective(step_fn, sizes):
    nA, nB, nG = sizes
    seen = {}
    for A in range(nA):
        for B in range(nB):
            for G in range(nG):
                out = step_fn(A,B,G)
                if out in seen:
                    print("Collision:", (A,B,G), "vs", seen[out])
                    return
                seen[out] = (A,B,G)
    print("Mapping␣is␣bijective␣over", nA*nB*nG, "states.")

if __name__ == "__main__":
    print("Ternary:")
    check_bijective(ternary_step, (3,3,5))
    print("Pentary:")
    check_bijective(pentary_step, (5,5,5))
```

7 Conclusion

We introduced a reversible, multi-radix symbolic gate suitable for the RHEA-UCM computational architecture. The gate supports binary compatibility, reversible ternary and pentary arithmetic, glyph/entropy tracking, and strictly bijective transition rules. Its Hamiltonian interpretation aligns with the RHEA-UCM framework, providing a concrete path toward RHEA-IC hardware capable of reversible symbolic cycles with controlled entropy production.

References

- C. H. Bennett, "Logical Reversibility of Computation," *IBM J. Res. Dev.*, 1973.
- R. Landauer, "Irreversibility and Heat Generation in the Computing Process," *IBM J. Res. Dev.*, 1961.

- E. Fredkin and T. Toffoli, "Conservative Logic," *Int. J. Theor. Phys.*, 1982.
- P. M. Roe, Submitted manuscripts to SIADS, SICON, SINUM (2025).
- P. M. Roe, "RHEA-UCM Framework," Zenodo (embargoed), 2025.