

Article

Not peer-reviewed version

Continuous Universal Analog Logic (LCUA): Smooth Functional Equivalents of Digital Gates Without Comparators

[Ricardo Adonis Caraccioli Abrego](#) *

Posted Date: 30 October 2025

doi: 10.20944/preprints202510.2372.v1

Keywords: continuous logic; analog computing; comparator-free logic; booleanizer; contraction mapping; neuromorphic hardware; mixed-signal circuits; phase/oscillator logic



Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a Creative Commons CC BY 4.0 license, which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

Article

Continuous Universal Analog Logic (LCUA): Smooth Functional Equivalents of Digital Gates Without Comparators

Ricardo Adonis Caraccioli Abrego 

Department of Electrical Engineering, Universidad Nacional Autónoma de Honduras (UNAH-VS), Campus Cortés, Honduras; ricardo.caraccioli@unah.edu.hn

Abstract

We present a framework to implement Boolean logic using only smooth functions—sums, products, and low-cost nonlinearities—without hard thresholds or explicit comparators, while exactly preserving truth tables on the Boolean vertices $\{0, 1\}$. The core is a family of polynomial gate primitives (AND, OR, XOR, NAND, etc.) that are exact at the vertices and are interleaved with a *smooth booleanizer* $B_n(x) = \frac{x^n}{x^n + (1-x)^n}$ that stabilizes trajectories toward the attractors 0 and 1 without discretization. We prove (i) exactness on the Boolean vertices; (ii) existence of attractors with local contraction for B_n ; (iii) a block-contraction condition that yields stability in deep cascades; and (iv) a phase/harmonic variant via the encoding $s = 2b - 1 \in \{-1, +1\}$ (equivalently $s = \cos(\pi(1 - b))$). We synthesize a continuous half-adder and a continuous SR latch (smooth ODEs), and we outline an analog-hardware route using standard op-amp adders/scalers, analog multipliers, and exponential (log–antilog) cells. We discuss sensitivity, cost, speed, and robustness under variation, and we identify research directions linking continuous logic to neuromorphic and AI-centric analog-digital co-design.

Keywords: continuous logic; analog computing; comparator-free logic; booleanizer; contraction mapping; neuromorphic hardware; mixed-signal circuits; phase/oscillator logic

1. Introduction and Motivation

Decades of CMOS scaling made digital logic the dominant computing substrate. Yet the energy costs of ubiquitous inference and sensing have revived interest in analog and mixed-signal approaches for specific tasks (e.g., multiply–accumulate, sensing near the source, and low-latency control). Contemporary AI workloads, along with edge constraints, motivate exploring logic families that: (i) minimize conversion overhead between analog and digital domains, (ii) offer graceful degradation under noise, and (iii) can interface naturally with neuromorphic and analog ML primitives.

This paper develops *Continuous Universal Analog Logic (LCUA)*: smooth gate realizations exact on $\{0, 1\}$, stabilized by a simple booleanizer B_n . The contributions are:

- **Smooth polynomial gates** (NOT/AND/OR/XOR/NAND/XNOR) that coincide with Boolean truth tables on $\{0, 1\}$.
- **Booleanizer B_n** : a two-attractor map on $[0, 1]$ with local contraction around 0 and 1; a single scalar parameter $n > 1$ trades off noise tolerance and flatness around $1/2$.
- **Cascade stability**: a sufficient block-contraction condition ensuring that deep interleavings of gates and booleanizers remain within a tube around the Boolean vertices and converge.
- **Harmonic/phase variant**: an equivalent gate family in the encoding $s = 2b - 1 \in \{-1, +1\}$ (equivalently $s = \cos(\pi(1 - b))$), relevant to oscillator/phase domains.
- **Analog route-to-hardware**: practical realizations using op-amp adders/scalers, analog multipliers (e.g., Gilbert-cell/OTA-based), and a log–antilog realization of B_n .

Why now?

Inference energy, analog sensing, and on-device AI increasingly rely on compact, low-latency primitives. LCUA eliminates hard comparators as the fundamental discretization step while still asymptotically restoring Boolean states. This can reduce analog–digital crossings and enable continuous-time pipelines (e.g., in-sensor processing) that are compatible with AI front-ends and neuromorphic back-ends.

2. Related Work

Classical Boolean algebra (Boole) and its realization via relays/switching (Shannon) underlie digital electronics. Fuzzy logic and t/s-norms (Zadeh; Hájek; Klement–Mesiar–Pap) explore continuous operations on $[0, 1]$ (e.g., product t-norm $a \cdot b$ and probabilistic s-norm $a + b - ab$). Neuromorphic computing (Mead; Indiveri & Liu) and ultra-low-power analog design (Sarpeshkar) show the efficacy of analog primitives for efficient computation. Prior analog-logic proposals (e.g., continuous-time analog logic for probabilistic inference) typically end with discretization stages or assume domain-specific comparators. Here we prove a sufficient cascade-contraction condition for a family of smooth gates that need no hard thresholds to recover Boolean behavior.

3. Formal Framework

3.1. Signal Space and Polynomial Gates

We work on the signal space $S = [0, 1]$. For $a, b \in [0, 1]$, define smooth gate primitives

$$\text{NOT}(a) = 1 - a, \quad (1)$$

$$\text{AND}(a, b) = ab, \quad (2)$$

$$\text{OR}(a, b) = a + b - ab, \quad (3)$$

$$\text{XOR}(a, b) = a + b - 2ab, \quad (4)$$

$$\text{NAND}(a, b) = 1 - ab, \quad (5)$$

$$\text{XNOR}(a, b) = 1 - a - b + 2ab. \quad (6)$$

Lemma 1 (Exactness on vertices). *For $a, b \in \{0, 1\}$, the above formulas coincide with the corresponding Boolean truth tables.*

Proof. Use $a^2 = a$ for $a \in \{0, 1\}$ and evaluate each polynomial on the vertices. \square

3.2. Smooth Booleanizer

Definition 1 (Booleanizer B_n). *For $n > 1$, define $B_n : [0, 1] \rightarrow [0, 1]$ by*

$$B_n(x) = \frac{x^n}{x^n + (1-x)^n}. \quad (7)$$

Proposition 1 (Fixed points and symmetry). *$B_n(0) = 0$, $B_n(1) = 1$, and $B_n(1-x) = 1 - B_n(x)$.*

Lemma 2 (Local contraction at the attractors). *There exist $\delta \in (0, \frac{1}{2})$ and $\kappa \in (0, 1)$ such that for $x \in [0, \delta] \cup [1 - \delta, 1]$, one has $|B'_n(x)| \leq \kappa$. In particular, 0 and 1 are locally attractive fixed points.*

Proof. Differentiate to obtain

$$B'_n(x) = \frac{n x^{n-1} (1-x)^{n-1}}{(x^n + (1-x)^n)^2}. \quad (8)$$

On $[0, \delta]$ with $\delta < 1/2$, use $(1-x) \geq (1-\delta)$ to bound $|B'_n(x)| \leq n x^{n-1} / (1-\delta)^{n+1} \rightarrow 0$ as $x \rightarrow 0$, for $n > 1$. By symmetry, the same holds on $[1 - \delta, 1]$. Continuity implies the existence of $\kappa < 1$. \square

3.3. Deep Cascades with Interleaved Projection

Let each block $G_\ell : [0, 1]^m \rightarrow [0, 1]$ be L_{G_ℓ} -Lipschitz on its operating range $I_\ell \subset [0, 1]$. Consider the depth- L interleaving

$$\text{output}_\ell = B_n(G_\ell(\text{inputs}_\ell)), \quad \ell = 1, \dots, L. \quad (9)$$

Theorem 1 (Sufficient block-contraction for cascade stability). *If for every subchain $[i, k]$ we have*

$$\prod_{j=i}^k L_{G_j} \cdot \prod_{j=i}^k \sup_{x \in I_j} |B'_n(x)| < 1, \quad (10)$$

then the interleaved network (9) is contractive by blocks: trajectories remain in a neighborhood (“tube”) of $\{0, 1\}$ and converge to Boolean vertices after sufficiently many stages.

Proof Sketch. A composition of Lipschitz maps is Lipschitz with constant the product of the constituent constants. The “tube” around $\{0, 1\}$ is precisely $[0, \delta] \cup [1 - \delta, 1]$, where B_n is contractive (Lemma 2). Even if a gate G_ℓ pushes a signal slightly away from the attractors (e.g., $\text{OR}(0.9, 0.9) = 0.99$), the subsequent B_n pulls it back with a net contraction factor. The product bound (10) ensures contraction on each subchain and thus convergence by the Banach fixed-point principle. \square

4. A Harmonic (Phase) Variant

Encode $b \in \{0, 1\}$ as $s = 2b - 1 \in \{-1, +1\}$ (equivalently $s = \cos(\pi(1 - b))$). This encoding is natural in oscillator/phase domains and differential signaling: it avoids DC offsets, improves common-mode noise rejection, and interfaces cleanly with coupled-oscillator and Ising-like substrates. For $s_a = 2a - 1$ and $s_b = 2b - 1$, define

$$\text{NOT}_\varphi(a) = \frac{1 - s_a}{2}, \quad (11)$$

$$\text{AND}_\varphi(a, b) = \frac{1 + s_a + s_b + s_a s_b}{4}, \quad (12)$$

$$\text{OR}_\varphi(a, b) = \frac{3 + s_a + s_b - s_a s_b}{4}, \quad (13)$$

$$\text{XOR}_\varphi(a, b) = \frac{1 - s_a s_b}{2}. \quad (14)$$

These formulas are smooth and match the Boolean truth tables exactly on $\{0, 1\}$ under the $s = 2b - 1$ encoding. Interleaving with B_n remains recommended in noisy phase domains.

5. Circuit Synthesis Examples

5.1. Continuous Half-Adder

For $a, b \in [0, 1]$ let

$$\text{SUM} = B_n(a + b - 2ab), \quad (15)$$

$$\text{CARRY} = B_n(ab). \quad (16)$$

5.2. Continuous SR Latch (Smooth ODEs)

For states $x, y \in [0, 1]$ and inputs $S, R \in [0, 1]$, with time constant $\tau > 0$,

$$\tau \dot{x} = -x + B_n((1 - R)(1 - y)), \quad (17)$$

$$\tau \dot{y} = -y + B_n((1 - S)(1 - x)). \quad (18)$$

With $S = R = 0$, the system has two stable equilibria near $(x, y) \approx (1, 0)$ and $(0, 1)$; smooth pulses on S or R toggle the state, as in a standard SR latch built from cross-coupled NOR gates.

6. Toward Analog Hardware

Blocks.

(i) Op-amp adders/subtractors and scalers; (ii) analog multipliers (e.g., AD633-class, OTA/Gilbert cells); (iii) a log–antilog or exponential cell to realize $z \mapsto z^n$ and $(1 - z)^n$; (iv) low-pass RC for glitch suppression.

Scaling and range.

The polynomial gates map $[0, 1]^2 \rightarrow [0, 1]$ for inputs in range. In the presence of noise or offsets, apply B_n periodically to restore margins.

Realizing B_n .

Options include: (a) explicit log–antilog stages to compute x^n and $(1 - x)^n$, followed by a soft division; (b) rational/Padé approximations near $x \in [0, 1]$; (c) piecewise-smooth approximations tuned for hardware efficiency.

7. Discussion: Sensitivity, Cost, Speed, Robustness

Sensitivity to n . Small n may not contract enough; large n flattens near $x = 1/2$ and can amplify offsets. Choose n to balance RMS error in cascades vs. settling time and bandwidth.

Cost of the booleanizer. B_n requires exponentiation or an accurate surrogate. Compare area/energy to a CMOS inverter baseline; explore Padé approximants and log–antilog implementations. The implementation cost of B_n is likely higher than a simple digital comparator/inverter in standard CMOS, but the trade-off is the elimination of ADC/DAC boundaries in fully analog pipelines.

Speed and bandwidth. Settling is set by the booleanizer pole and multiplier bandwidth. Small-signal/Bode analysis can bound maximum rate and accumulated jitter in deep chains.

Robustness to variation. Monte Carlo (SPICE) with $\pm 5\%$ – $\pm 10\%$ tolerances on gains/resistors and input noise: measure truth-table error, 0/1 margins, and switching-time distributions.

Comparison with CMOS. Define a minimal NAND/half-adder benchmark to estimate static/dynamic energy, area, and speed relative to a CMOS baseline.

8. Protocol for Validation

1. **Numerical:** compare deep cascades with and without B_n ; log-relative error and settling time vs. n .
2. **SPICE:** implement AND/OR/XOR and B_n ; AC and transient analysis; full Monte Carlo.
3. **Prototype:** PCB with op-amps and OTA/Gilbert multipliers, plus a log–antilog stage for B_n ; validate truth tables, robustness, and throughput.

9. Limitations and Future Work

Global contraction without input-range assumptions remains open; the cost of B_n versus CMOS in scaled technologies must be characterized; the phase variant needs a complete phase-noise analysis. Future work: alternative booleanizers with the same fixed points but lower cost; more sequential elements (JK/T) and continuous memory; co-design with neuromorphic arrays and oscillator networks; and system-level studies for in-sensor and edge-AI pipelines.

10. Conclusions

Boolean logic can be realized with smooth functions without hard comparators, exactly preserving truth tables on $\{0, 1\}$ while achieving stability through interleaved projection. This opens a path for analog-universal architectures that trade some generality for significant efficiency and noise tolerance in AI-adjacent workloads.

Acknowledgments: The author thanks colleagues and students at UNAH-VS for insightful discussion and technical feedback.

Appendix A. Derivatives and Bounds for B_n

Explicit derivatives and uniform bounds of $|B'_n|$ on $[0, \delta] \cup [1 - \delta, 1]$, and products of constants for block contraction.

Appendix B. Truth Tables and Verification

Truth tables for the polynomial and harmonic variants, showing exact agreement on $\{0, 1\}$.

References

1. G. Boole, *An Investigation of the Laws of Thought*, Walton and Maberly, 1854.
2. C. E. Shannon, "A Symbolic Analysis of Relay and Switching Circuits," *Trans. AIEE*, 57 (1938).
3. L. A. Zadeh, "Fuzzy Sets," *Information and Control*, 8(3):338–353, 1965.
4. P. Hájek, *Metamathematics of Fuzzy Logic*, Kluwer, 1998.
5. E. P. Klement, R. Mesiar, E. Pap, *Triangular Norms*, Kluwer, 2000.
6. C. Mead, *Analog VLSI and Neural Systems*, Addison-Wesley, 1989.
7. R. Sarpeshkar, *Ultra Low Power Bioelectronics: Fundamentals, Biomedical Applications, and Bio-Inspired Systems*, Cambridge Univ. Press, 2010.
8. G. Indiveri and S.-C. Liu, "Memory and Information Processing in Neuromorphic Systems," *Proc. IEEE*, 103(8):1379–1397, 2015.
9. E. Chicca and G. Indiveri, "A recipe for creating ideal hybrid memristive–CMOS neuromorphic processing systems," *APL Materials*, 7(8):081109, 2019.
10. B. Vigoda, *Analog Logic: Continuous-Time Analog Circuits for Statistical Signal Processing*, PhD Thesis, MIT, 2003.
11. K. Chen and L. Leu, "Analog Logic Automata," 2008, preprint/tech report.
12. N. H. E. Weste and D. M. Harris, *CMOS VLSI Design: A Circuits and Systems Perspective*, 4th ed., Addison-Wesley, 2010.
13. K. Ogata, *Modern Control Engineering*, 5th ed., Prentice Hall, 2010.
14. K. J. Åström and R. M. Murray, *Feedback Systems*, Princeton Univ. Press, 2008.
15. G. A. Baker Jr. and P. Graves-Morris, *Pade Approximants*, 2nd ed., Cambridge Univ. Press, 1996.
16. AD633 *Analog Multiplier Data Sheet*, Analog Devices, Rev. 2016.
17. B. Gilbert, "A precise four-quadrant multiplier with subnanosecond response," *IEEE J. Solid-State Circuits*, 3(4):365–373, 1968.
18. D. B. Strukov et al., "The missing memristor found," *Nature*, 453:80–83, 2008.
19. J. Lazzaro, S. Ryckebusch, M. A. Mahowald, C. A. Mead, "Winners-Take-All Networks of O(N) Complexity," *Advances in Neural Information Processing Systems*, 1991.
20. L. O. Chua, "Memristor—The missing circuit element," *IEEE Trans. Circuit Theory*, 18(5):507–519, 1971.
21. R. H. Katz, *Contemporary Logic Design*, 2nd ed., Pearson, 2004.
22. A. S. Sedra and K. C. Smith, *Microelectronic Circuits*, 7th ed., Oxford Univ. Press, 2015.
23. S. Franco, *Design with Operational Amplifiers and Analog Integrated Circuits*, 4th ed., McGraw-Hill, 2014.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.