International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 10 | Oct - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

FPGA-Oriented Implementation and Simulation Verification of an I-CDPK Codec

Dr. G. P. Halde[1], Jay Gandhewar[2], Mrunal Apkure[3], Harshal Barai[4], Atharva Surse[5]

[1] Assistant Professor; [2]–[5] Students, Department of Electronics and Communication Engineering, Priyadarshini College of Engineering, Nagpur, India Corresponding authors: jaygandhewar310@gmail.com | gouri.halde@pcenagpur.edu.in

Abstract— Interleaved Chaotic Differential Peaks Keying (I-CDPK) is a chaos-based digital communication technique that represents information using peak-related features of a chaotic sequence. This paper reports the current FPGA-oriented implementation and simulation-verification status of an I-CDPK codec. A Lorenz system provides the state sequence, followed by peak-event detection, differential processing, bit-dependent keying, receiver-side processing, and event-level comparison. The workflow combines a Python behavioural reference, signed Q7.14 fixed-point arithmetic, SystemVerilog RTL, Vivado/XSim simulation, and Value Change Dump (VCD) analysis. The recorded dataset contains 5,000 Lorenz samples, 191 canonical information events, 955 pipeline-stage events, and 991 timeline entries. An 18-record transmit/receive comparison reports zero serial mismatches. These results support consistency for the recorded simulation cases only; they do not establish physical FPGA operation, general bit-error performance, or cryptographic security.

Keywords— I-CDPK, chaotic communication, Lorenz system, FPGA, SystemVerilog, fixed-point arithmetic, Q7.14, RTL verification.

1. INTRODUCTION

Chaos-based communication studies deterministic nonlinear dynamical systems whose trajectories appear irregular and exhibit sensitivity to initial conditions and parameter variations. The continuous-time Lorenz system is a benchmark nonlinear attractor [1]. State synchronization between coupled transmitter and receiver systems has motivated research into chaos synchronization for transmitting and recovering information across communication channels [2].

Chien, Hung, and Liao introduced Interleaved Chaotic Differential Peaks Keying (I-CDPK) as a non-correlator-based digital communication scheme in which payload bits are interleaved across chaotic state variables and recovered from differences between consecutive peak events [3]. Surveys have examined trade-offs among carrier synchronization, power efficiency, data throughput, and receiver-processing overhead [4]. Related studies have considered chaotic signalling with impulsive synchronization and noisy transmission environments [5], [6].

This paper documents an FPGA-oriented hardware-modelling and simulation-verification workflow rather than proposing a new modulation concept.

2. OBJECTIVES AND CONTRIBUTIONS

The objectives are to construct a deterministic software reference model for the Lorenz generator and modulation stages; establish signed fixed-point datapaths; describe the end-to-end codec architecture in SystemVerilog RTL; verify timing behaviour and state transitions using AMD Vivado/XSim simulation and Value Change Dump (VCD) trace analysis; and execute automated consistency checks between transmitted and received payloads. The contribution is a reproducible, simulation-verified digital-hardware workflow with defined implementation boundaries.

3. SYSTEM MODEL AND METHODOLOGY

3.1 Lorenz Chaotic Generator

The baseline continuous-time chaotic generator is modelled by three coupled autonomous first-order differential equations:

$$dx/dt = \sigma(y - x) \quad (1)$$

$$dy/dt = x(\rho - z) - y \quad (2)$$

$$dz/dt = xy - \beta z \quad (3)$$

The parameters are $\sigma = 10$, $\rho = 28$, and $\beta = 8/3$, with integration time step $\Delta t = 0.01$ and initial values $x_0 = y_0 = z_0 = 1.0$. A baseline sequence of 5,000 samples was generated to establish reference datasets.

3.2 Discrete Update and Fixed-Point Arithmetic

The behavioural software reference uses explicit first-order Euler integration. The hardware-oriented datapath uses signed 21-bit Q7.14 fixed-point formatting with scale factor $2^{14} = 16{,}384$. The quantized integer constants are SIGMA_F = 163,840, RHO_F = 458,752, BETA_F = 43,691,
and DT_F = 164. Because BETA_F and DT_F involve fractional quantization, reproducing RTL-equivalent outputs requires these exact integer representations.

During Q7.14 multiplication, intermediate products contain 28 fractional bits and are realigned to Q7.14 using an arithmetic right shift by 14 bits. Rounding, saturation boundaries, and shift sequencing must be consistent across the hardware modules.

3.3 Differential Processing and Keying

Let P[n] denote the peak value identified by the peak-detection engine. The differential metric is:

$$D[n] = P[n] - P[n-1] \quad (4)$$

The keying rule is:

$$K[n] = D[n] + 4(2b[n] - 1), \quad b[n] \in \{0, 1\} \quad (5)$$

This introduces an offset of +4 for bit 1 and −4 for bit 0.

4. IMPLEMENTATION AND VERIFICATION

The design cycle uses a multi-tier verification methodology. A Python golden reference model produces expected numerical states and quantized Q7.14 traces. Hardware modules are developed in IEEE Std 1800-2017 SystemVerilog [9] and simulated using AMD Vivado/XSim [8].

The codec flow includes input bits, Lorenz state generation, peak detection, differential processing, bit-dependent keying, receiver-side demodulation, VCD event extraction, unified timeline construction, and transmit/receive comparison. VCD dump files capture timestamped transitions; parsed timeline datasets are used to inspect pipeline timing and bit recovery. Behavioural simulation confirms RTL functional consistency for the tested cases but does not establish FPGA placement, routing, or timing closure [10], [11].



Fig. 1. I-CDPK codec simulation and verification architecture.

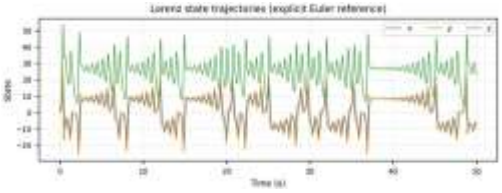


Fig. 2. Lorenz state trajectories generated using the stated explicit-Euler reference parameters.

5. RESULTS AND DISCUSSION

The quantitative verification results extracted from the recorded VCD dataset are summarized in Table 1.

Verification metric	Value	Context / scope
Lorenz state samples	5,000	Euler integration run length
Canonical information events	191	Detected peak-keying events
Pipeline-stage events	955	Logged across RTL stages
Unified timeline entries	991	Parsed VCD transitions
TX/RX records compared	18/18	Extracted test words
Serial bit mismatches	0	Nominal simulation run
Keying-rule checks	191/191	Recorded checks
Demodulation checks	191/191	Recorded checks

Table 1. Simulation verification and VCD-analysis summary.

The 18-record comparison reported zero bit mismatches during nominal simulation. This result is bounded to the recorded test conditions and is not a general bit-error rate (BER) measurement across arbitrary communication channels.

6. LIMITATIONS AND FUTURE WORK

The results are limited to simulation. Synthesis and physical implementation on an FPGA (for example, an AMD Artix-7 or Zynq device) remain necessary to establish post-implementation maximum operating frequency, DSP48 utilization, and setup/hold timing slack. Noisy channels such as additive white Gaussian noise (AWGN) and Rayleigh fading should be evaluated to obtain empirical BER-versus-E_b/N_0 curves. Chaotic keying does not replace certified cryptography; security claims require separate validation.

7. CONCLUSION

This work demonstrates a structured FPGA-oriented simulation workflow for an I-CDPK codec using signed Q7.14 arithmetic and SystemVerilog. Across 5,000 baseline chaotic samples, the 18 transmit/receive records achieved full keying and demodulation agreement in Vivado/XSim without serial mismatches. Future work focuses on FPGA hardware deployment and BER analysis.

ACKNOWLEDGMENT

The authors thank the project guide and faculty members for their guidance and support, and Priyadarshini College of Engineering for providing facilities and resources for the work.

REFERENCES

[1] E. N. Lorenz, "Deterministic nonperiodic flow," Journal of the Atmospheric Sciences, vol. 20, no. 2, pp. 130–141, 1963.

[2] L. M. Pecora and T. L. Carroll, "Synchronization in chaotic systems," Physical Review Letters, vol. 64, no. 8, pp. 821–824, 1990.

[3] T.-I. Chien, Y.-C. Hung, and T.-L. Liao, "A non-correlator-based digital communication system using I-CDPK modulation," Chaos, Solitons & Fractals, vol. 29, no. 4, pp. 965–977, 2006.

[4] G. Halde and A. S. Gandhi, "Impulsive observer based chaotic synchronization with application to interleaved chaotic differential peaks keying (I-CDPK)," in Proc. TENCON 2015, pp. 1–4, 2015.

[5] G. Kaddoum, "Wireless chaos-based communication systems: A comprehensive survey," IEEE Access, vol. 4, pp. 2621–2648, 2016.

[6] G. P. Halde, A. S. Gandhi, and A. S. Telang, "Secure data communication through impulsive synchronization," research manuscript, 2016.

[7] P. S. Mawale, G. Halde, V. Shirpurkar, and S. Chaturvedi, "A secure chaotic communication system," International Journal of Scientific Research and Development, vol. 2, no. 7, pp. 507–510, 2014.

[8] AMD, Vivado Design Suite User Guide: Logic Simulation, UG900, 2025.

[9] IEEE, IEEE Standard for SystemVerilog, IEEE Std 1800-2017, 2018.

[10] AMD, Vivado Design Suite User Guide: Synthesis, UG901, 2025.

[11] AMD, Vivado Design Suite User Guide: Implementation, UG904, 2025.

[12] G. Halde, S. Wasnik, and S. Parihar, "Performance analysis of chaos based transmission," IJSDR, vol. 2, no. 7, 2017.

[13] G. P. Halde and A. S. Gandhi, "Secure data communication through impulsive synchronization of multiple attractor chaotic systems," Multidisciplinary Journal of Pensee, vol. 51, no. 11, pp. 1047–1059, 2021.