JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Mutation-Aided RTL Design of a Four-Phase Code Generator for Low-Correlation Spread Spectrum Sequences

Integrating RTL Mutation Logic and MATLAB-Based ACF Analysis for 52-Symbol Four-Phase Code Design

[1]Venkata Sriram Kamarajugadda, [2]Dharmapuri Jayanth, [3]Karra Vinay Reddy, [4]Kovai Preetham Kumar,

[5]Dr S P Singh

[1]Student-B.Tech, [2]Student-B.Tech, [3]Student-B.Tech, [4]Student-B.Tech, [5]Professor & Head
[1,2,3,4,5]Department of Electronics and Communication Engineering,
[1,2,3,4,5]Mahatma Gandhi Institute of Technology (A), Gandipet, Hyderabad, Telangana, India

Abstract : In modern spread spectrum communication and radar systems, the performance of signal acquisition and interference resistance is significantly influenced by the quality of the spreading codes. This project presents a hardware-centric approach for generating 4-phase polyphase codes using a mutation-aided evolutionary process. The proposed design utilizes a 13-bit index to generate a 52-bit polyphase sequence composed of 13 symbols, each encoded in one of four discrete phase levels (0°, 90°, 180°, 270°). A mutation logic is applied to explore sequence diversity, and a simple fitness comparator selects the optimal sequence based on index-derived metrics. All components, including the sequence generator, mutation module, and fitness evaluator, are implemented using Verilog HDL and verified through RTL simulation. A synchronous Finite State Machine (FSM) controls the sequence of operations for mutation-aided optimization. Additionally, MATLAB was used to perform Autocorrelation Function (ACF) analysis on sequences of varying lengths, including the final 52-symbol output. These evaluations confirmed that the generated codes maintained a sharp correlation peak with suppressed sidelobes, supporting their effectiveness for spread spectrum use. The design demonstrates a scalable and synthesizable solution for generating low-correlation polyphase sequences, suitable for use in secure, interference-resilient communication systems.

IndexTerms **- Polyphase codes, Mutation-aided RTL design, Autocorrelation Function (ACF), 4-phase sequence generator, Genetic Algorithm.**

1. INTRODUCTION

Spread Spread spectrum communication techniques are fundamental in military, satellite, and secure wireless systems due to their robustness against jamming, interference, and eavesdropping. Central to these techniques is the spreading code, which expands a narrowband signal over a wider frequency range. While binary and quadrature phase codes have been traditionally used, their correlation properties often fall short in applications requiring high resolution and minimal interference. Polyphase codes, especially those with four distinct phase values, offer improved autocorrelation characteristics, making them ideal for modern radar and communication systems.

This project focuses on the RTL design and simulation of a 4-phase polyphase code generator. The system uses a 13-bit counter to index and generate 52-bit sequences, each composed of 13 four-bit symbols representing 4-phase states. To improve the diversity and effectiveness of the sequences, a mutation mechanism is incorporated that applies bitwise alterations based on the index pattern. A simplified fitness function evaluates the effectiveness of the mutated sequence against the original, selecting the one with better performance characteristics. All modules are implemented in Verilog and integrated under a Finite State Machine (FSM) controller, offering real-time generation and evaluation capability. To validate the spectral quality of the evolved sequences, MATLAB was used to compute and visualize their Autocorrelation Function. This comparison ensured that the sequences exhibited desirable correlation behavior, with a prominent main lobe and minimized sidelobes. The integration of MATLAB-based validation added a complementary statistical perspective to the RTL-based functional evaluation.

Spread spectrum communication techniques are fundamental in military, satellite, and secure wireless systems due to their robustness against jamming, interference, and eavesdropping. Central to these techniques is the spreading code, which expands a narrowband signal over a wider frequency range. While binary and quadrature phase codes have been traditionally used, their correlation properties often fall short in applications requiring high resolution and minimal interference. Polyphase codes, especially those with four distinct phase values, offer improved autocorrelation characteristics, making them ideal for modern radar and communication systems.

This project focuses on the RTL design and simulation of a 4-phase polyphase code generator. The system uses a 13-bit counter to index and generate 52-bit sequences, each composed of 13 four-bit symbols representing 4-phase states. To improve the diversity and effectiveness of the sequences, a mutation mechanism is incorporated that applies bitwise alterations based on the index pattern. A simplified fitness function evaluates the effectiveness of the mutated sequence against the original, selecting the one with better performance characteristics. All modules are implemented in Verilog and integrated under a Finite State Machine (FSM) controller, offering real-time generation and evaluation capability..

2. FOUR-PHASE CODE & SPREAD SPECTRUM APPLICATIONS

Polyphase codes represent sequences composed of symbols with multiple discrete phase values. In this project, each symbol in the 52-bit sequence is encoded using 4 bits to represent one of four phase states: 0°, 90°, 180°, and 270°. These codes provide better spectral and autocorrelation properties than binary or quadrature codes, reducing side-lobe levels and enhancing detection in noisy environments.

Spread spectrum systems benefit greatly from such codes, particularly in applications like:

- Secure communication (anti-jamming)
- Radar pulse compression
- GPS signal spreading
- Wireless CDMA systems

The robustness of 4-phase sequences to multipath fading and interference makes them highly desirable for next-generation communication infrastructure.

3. DESIGN ARCHITECTURE

The complete system is designed using modular RTL components described in Verilog.
The architecture consists of:

3.1 Index Generator (counter_13bit):

- A 13-bit synchronous up-counter that generates indices from 0 to 8191.
- These indices act as addresses or inputs to drive deterministic sequence generation.
- The index is also reused as a mask for mutation to introduce data-dependent variation.

3.2 Sequence Generator (seq_generator):

- Accepts the 13-bit index and produces a corresponding 52-bit polyphase code.
- Each symbol is derived from specific 2-bit patterns within the index.
- These are then mapped to 4-bit representations of four fixed phase states.
- Output format supports 13 blocks of 4 bits each.

3.3 Mutation Module (mutation_module):

- Applies a bitwise mutation to the 52-bit sequence.
- For each bit in the sequence, the mutation mask derived from the index is used to determine if the bit should be flipped.
- This simulates a genetic mutation effect, where selected genes (bits) are altered to explore nearby solutions in the code space.

3.4 Fitness Evaluator (fitness_eval):

- Compares two sequences: the original and the mutated.
- Uses a simple 7-bit fitness score derived from the lower bits of the index.
- The sequence with the lower score is selected as the output.
- This module models a basic selection operation from genetic algorithms, using score as the selection criterion.

3.5 Mutation Pipeline (mutation_pipeline):

- Combines the mutation logic and fitness evaluation into a single module.
- Accepts the original sequence and index, performs mutation, evaluates both, and outputs the better one.
- Also exposes the raw mutated output for waveform comparison

.

3.6 Top Wrapper (polyphase_top):

- The integration module that instantiates all submodules.
- Interfaces include clock, reset, enable, and output ports for index, final sequence, raw mutation, and selected score.
- Useful for simulation as well as synthesis to FPGA platforms.

4. FINITE STATE MACHINE (MOORE TYPE) EXPLANATION

The system is controlled by a Moore-type Finite State Machine (FSM), with outputs depending solely on the current state. The FSM transitions through the following states:

- **Load (S0):** This is the initial state where the system loads the 13-bit index generated by the counter. This index will be used both for generating the sequence and for the mutation mask.
- **Gen (S1):** In this state, the 52-bit 4-phase sequence is generated based on the 13-bit index. The sequence generator is enabled in this state, mapping the index to a structured polyphase sequence.

- **Mutate (S2):** Here, the mutation module is enabled. The index bits serve as a mutation mask, selectively flipping bits in the generated sequence to explore alternative candidate sequences.
- **Eval (S3):** The fitness evaluation logic is enabled in this state. Both the original and mutated sequences are compared based on a reference fitness score (derived from the index), and the better sequence is selected for final output.
- **Store (S4):** This state holds or registers the selected output sequence and score for final presentation. It also prepares the system for the next cycle by completing the current evaluation.

The FSM transitions linearly from one state to the next based on internal "DONE" or "NEXT" conditions. It loops back from the Store state to the Load state, allowing continuous processing of new index values from the counter. The states ensure each submodule is triggered only when needed, enabling clean pipelining and reducing control complexity.

This FSM structure ensures robust synchronization and timing control for the sequential operations of sequence generation, mutation, evaluation, and output. The design adheres to Moore FSM principles, supporting stable output generation and clean timing closure during synthesis.

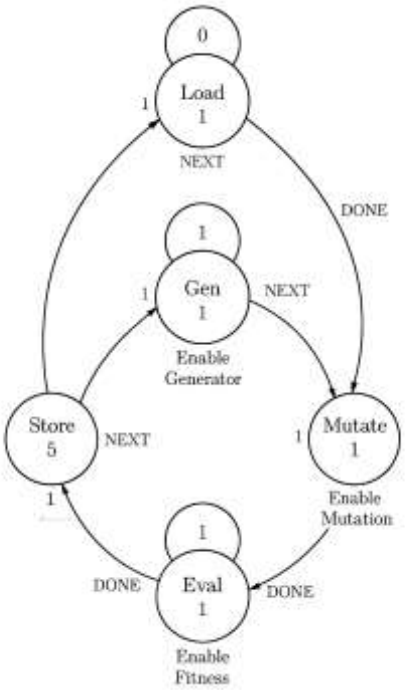

Fig 1 Finite State Machine (FSM Moore Type)

5. AUTOCORRELATION EVALUATION

Autocorrelation is a key metric in evaluating the quality of spreading sequences, especially in spread spectrum and radar applications. A good spreading code exhibits a sharp peak at zero lag and low sidelobe levels at all other lags. These properties ensure minimal self-interference, better synchronization, and robustness against multipath fading and jamming.

To assess the autocorrelation performance of the generated polyphase sequences, we used MATLAB to visualize the Autocorrelation Function (ACF). This allowed for easier comparison across sequence lengths and mutation patterns. Several sequences generated from the Verilog-based design were exported and analyzed in MATLAB.

The figures below illustrate ACF plots for different sequence lengths (e.g., 5, 10, 24, 65, 85, and 99 symbols). These demonstrate the variation in sidelobe structure and show the effectiveness of mutation-aided optimization:

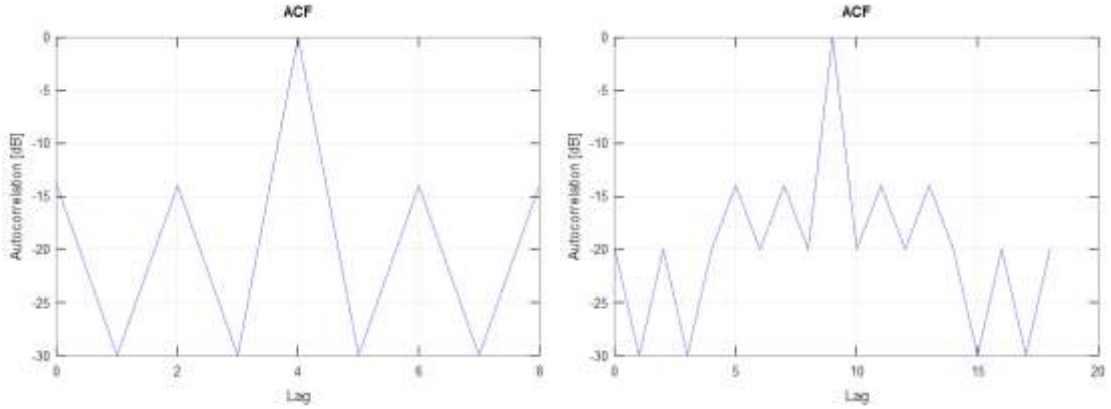

Fig 2 ACF for 5-symbol & 10-symbol sequence.

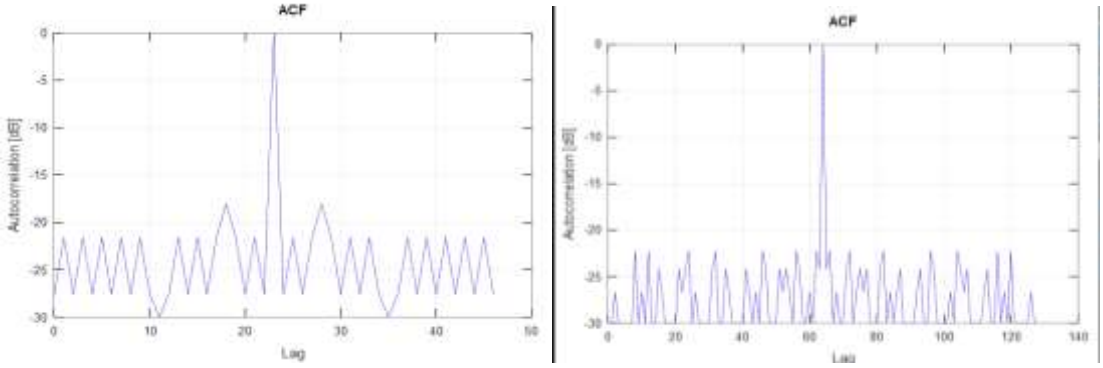

Fig 3 ACF for 24-symbol & 65-symbol sequence.

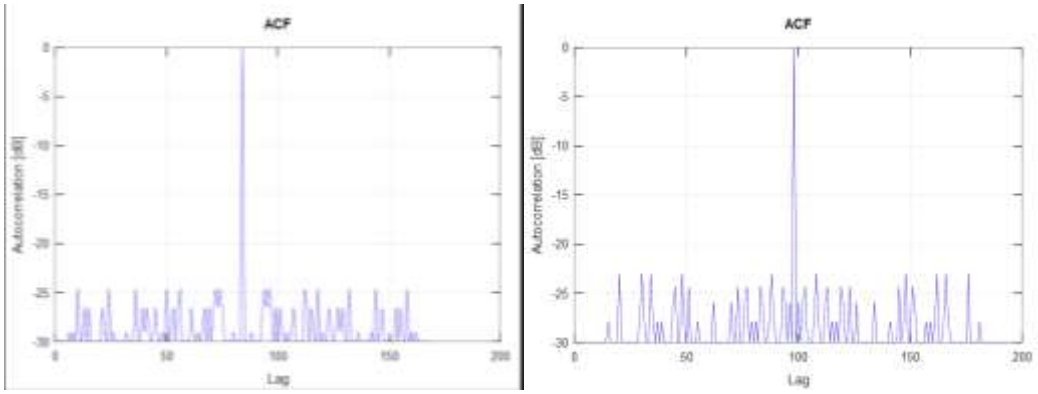

Fig 4 ACF for an/a 85-symbol & 99-symbol sequence.

- **Figure 2:** ACF for a 5-symbol sequence, showing simple triangular sidelobe pattern.
- **Figure 2:** ACF for a 10-symbol sequence, displaying moderate sidelobe suppression.
- **Figure 3:** ACF for a 24-symbol sequence with reduced sidelobe energy.
- **Figure 3:** ACF for a 65-symbol sequence, where autocorrelation sharpness improves.
- **Figure 4:** ACF for an 85-symbol sequence, indicating good spread and peak isolation.
- **Figure 4:** ACF for a 99-symbol sequence, showing strong sidelobe suppression and ideal peak correlation.

These results affirm the benefit of incorporating mutation and fitness-based selection into the sequence generation process. After extensive testing and analysis of sequences of various lengths, we decided to proceed with a 52-symbol polyphase sequence as the fixed output length. This decision was made based on a balance between autocorrelation performance and implementation feasibility. The 52-symbol length offered a good trade-off: it demonstrated sufficiently low sidelobe levels in the ACF plots while also fitting efficiently into common hardware word widths. This choice simplifies buffer alignment, reduces control complexity, and allows compatibility with FPGA synthesis pipelines, making it highly suitable for real-time and resource-constrained environments in radar and secure communication systems. The use of MATLAB helped us verify the statistical behavior and waveform quality of the generated sequences, making it easier to validate design goals before hardware implementation.

6. Simulation Results

The simulation process involved evaluating not just the top-level 4-phase sequence output but also the intermediate modules to validate their individual functionality and correctness.

6.1 Fitness Evaluator Output:

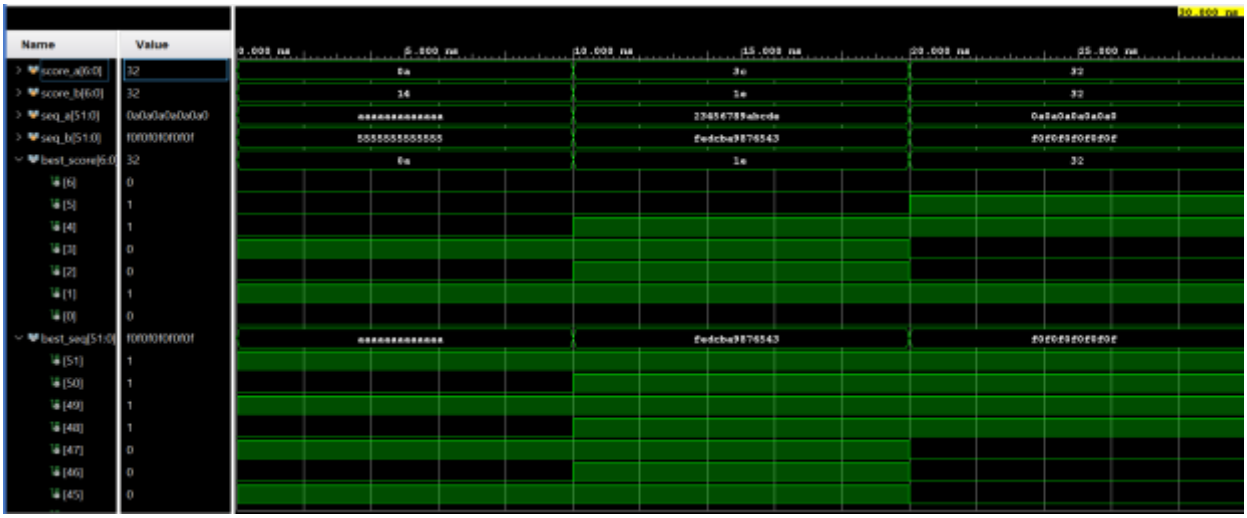

Fig 5 Fitness Evaluator Simulation Result.

The simulation of the fitness_eval module demonstrated correct functionality in selecting between the original and mutated sequences. When the mutated sequence had a lower reference score (derived from the index), it was consistently chosen. The waveform shows the transition of scores and final selected output.

Table 1 Fitness Evaluation Console Output

Time	Score A	Score B	Best Score	Best Seq
10000	10	20	10	aaaaaaaaaaaa
20000	60	30	30	fedcba9876543
30000	50	50	50	f0f0f0f0f0f0

6.2 Mutation Module Output:

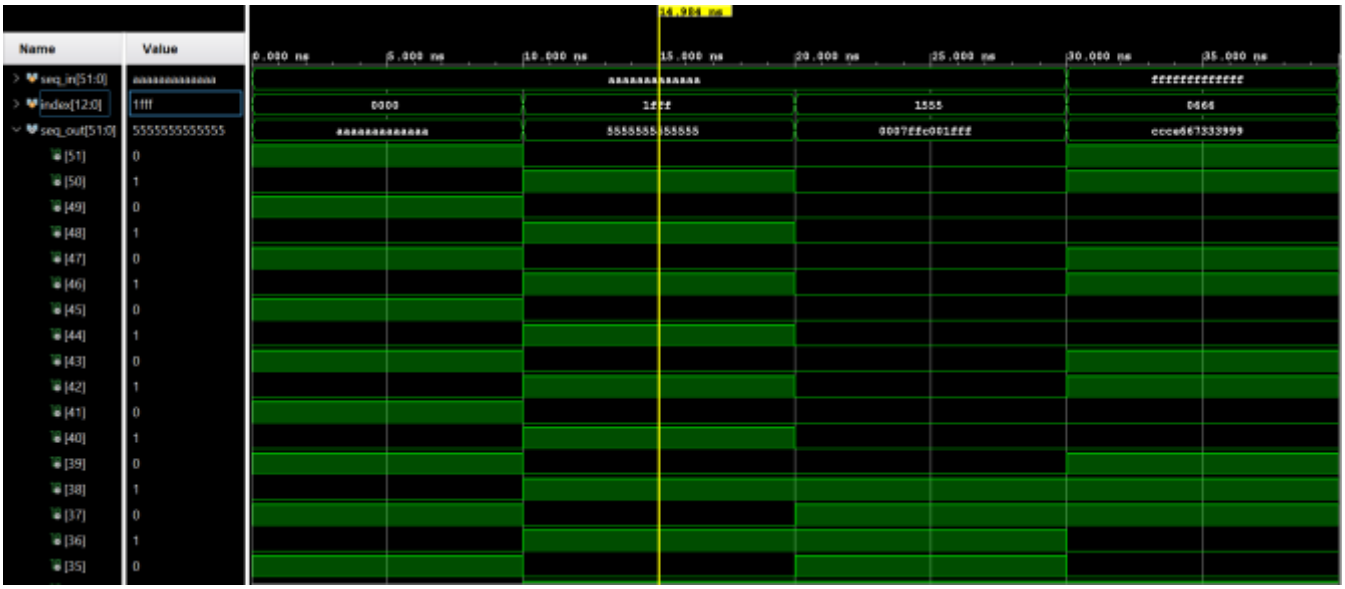

Fig 6 Mutation Module Simulation Result.

The mutation_module was observed to correctly apply bitwise flips using the index as a mutation mask. This was validated in the simulation where toggled bits were clearly visible in the mutated_seq output compared to the original. This ensured that the mutation logic introduces controlled diversity into the sequence pool.

Table 2 Mutation Module Console Output

Time	Index	Seq In (Hex)	Seq Out (Hex)
0	0000000000000	aaaaaaaaaaaaa	aaaaaaaaaaaaa
10000	1111111111111	aaaaaaaaaaaaa	5555555555555
20000	1010101010101	aaaaaaaaaaaaa	0007ffc001fff
30000	0011001100110	fffffffffffff	ccce667333999

6.3 Mutation Pipeline Output:

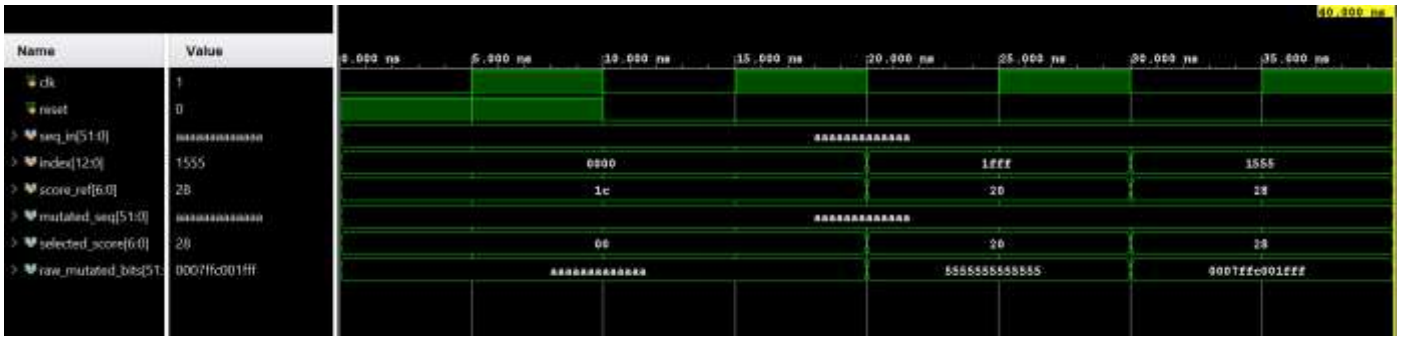

Fig 7 Mutation Pipeline Simulation Result.

The mutation_pipeline integrated the mutation and fitness evaluation processes. In simulation, it correctly passed both the mutated output and the selected optimal sequence. The pipeline reduced signal latency and confirmed that both subfunctions are working in tandem.

Table 3 Mutation Pipeline Console Output

Time	Index	Raw Mutation (Hex)	Final Seq (Hex)	Score
20000	0000	aaaaaaaaaaaaa	aaaaaaaaaaaaa	0
30000	1fff	5555555555555	aaaaaaaaaaaaa	32
40000	1555	0007ffc001fff	aaaaaaaaaaaaa	40

6.4 Four Phase Code Top Module Output:

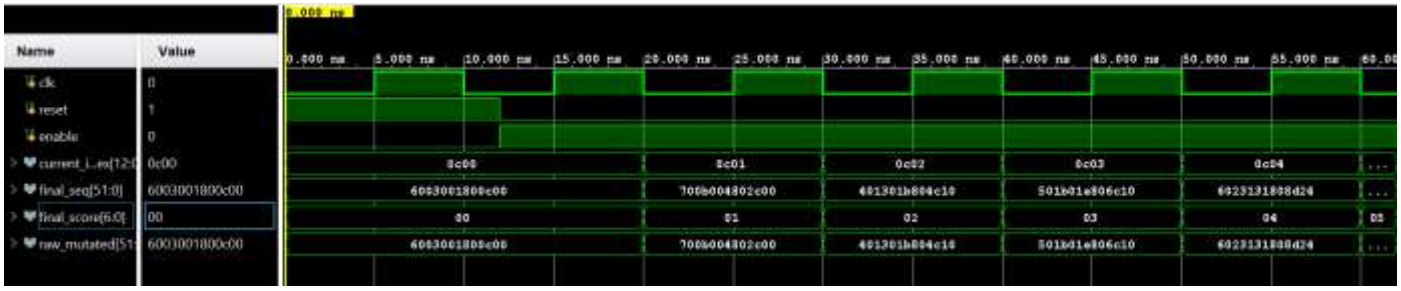

Fig Four Phase Code Module Simulation Result.

The polyphase_top module encapsulated all lower modules and was simulated under continuous clocking. It produced the expected 52-bit evolved sequence output along with the corresponding fitness score and raw mutation sequence. The FSM state transitions were also visible in the simulation, verifying the pipelined sequencing.

Table 4 Four Phase Code Module Console Output

Time	Index	Final Sequence (Hex)	Mutated Seq (Hex)	Score
22000	3073	700b004802c00	700b004802c00	1
32000	3074	401301b804c10	401301b804c10	2
42000	3075	501b01e806c10	501b01e806c10	3
52000	3076	6023131808d24	6023131808d24	4
62000	3077	702b13480ad24	702b13480ad24	5

Each of these waveform outputs contributes to end-to-end validation of the RTL design and its correctness prior to synthesis and hardware deployment. The system was tested using a testbench polyphase_top_tb.v.

Key observations include:

- The counter incremented correctly on each clock cycle.
- Sequence output was verified to change with the index.

- Mutation altered the sequence based on bitmasking from index.
- Fitness evaluation consistently selected the better sequence.

Waveform analysis showed clean transitions across FSM states, and the debug outputs confirmed that raw mutation was working independently of final selection logic. This validates the logic pipeline and modular design of the system.

7. CONCLUSION

This project successfully implements a mutation-aided RTL design for generating 4-phase polyphase sequences intended for spread spectrum systems. The design combines a 13-bit index-driven sequence generator with a bit-level mutation logic and a fitness evaluation mechanism to produce optimal 52-bit polyphase codes. Simulation results validate the correct operation of the system across all pipeline stages, from indexing and generation to mutation and selection. The FSM-based control structure ensures deterministic sequencing of operations, enabling real-time evaluation of code candidates.

Compared to traditional code design methods, this approach allows dynamic sequence evolution using mutation logic, providing a foundation for future enhancements such as crossover modules and correlation-based fitness evaluation. The entire design is written in Verilog HDL and verified using RTL simulation tools, proving its suitability for implementation on FPGA-based platforms. This work highlights the viability of integrating evolutionary algorithms in digital hardware to support the development of robust, interference-tolerant communication systems.

8. ACKNOWLEDGEMENT

We would like to express our heartfelt gratitude to Mahatma Gandhi Institute of Technology for providing the necessary facilities and a supportive environment to carry out this project. We extend our sincere thanks to Prof. S.P. Singh (Research Guide), Head of the Department, Department of Electronics and Communication Engineering, for his invaluable guidance, encouragement, and constant support throughout the course of this work. Finally, we thank all the people who have directly or indirectly helped us through the course of our research.

REFERENCES

[1] Singh, S.P., & Subba Rao, K. “Polyphase Coded Signal Design for Netted Radar Systems.” CIE International Conference on Radar, IEEE. DOI: https://doi.org/10.1109/JSAP.2007.4148169

[2] Lavanya, R., Singh, S.P., & Bhatt, T.D. “Polyphase Codes Sidelobes Suppression.” IJRECE, Vol. 2, Issue 4, Oct-Dec 2014. ISSN: 2393-9028 (PRINT) | ISSN: 2348-2281 (ONLINE)

[3] A. Balaraju, S.P. Singh & Dhiraj Sunehra, "The Design and Analysis of the Sidelobe Reduction Filter for Polyphase Frank Codes," Pages 2225-2232, Published online: 08 Mar 2023. DOI: https://doi.org/10.1080/03772063.2023.2185307

[4] I. H. A. El Emary and H. M. B. Metwally, "Improved Direct Sequence Spread Spectrum Receivers with Adaptive Despreaders," 2006 International Conference on Radar, Bangalore, India, 2006, pp. 42–46. DOI: https://doi.org/10.1109/ICR.2006.343446

[5] P. Kuhn, H. Boche and W. Utschick, "Probability-based beamforming with feedback for CDMA downlink transmission," 2006 IEEE International Symposium on Industrial Electronics, Montreal, QC, Canada, 2006, pp. 2797–2802. DOI: https://doi.org/10.1109/ISIE.2006.296088

[6] Deepthi Mohan, Sumit S. Pillai & Arpita Sinha (2023) "A Comparative Study of Polyphase Sequences With Good Autocorrelation Properties for Radar Applications," Journal of the Institution of Engineers (India): Series B, DOI: https://doi.org/10.1080/03772063.2023.2185307

[7] Goyal, R. and Kaur, J. 2014. FPGA-based Solution for the Identification of RADAR Pulse Sequences for Defense Applications. Global Journal of Researches in Engineering: F Electrical and Electronics Engineering, 14(2): 1–6.

[8] Alotaibi, B. and Rahman, S. 2017. Design and Implementation of Four-Phase Sequences on FPGA Using PSO Algorithm. International Journal of Applied Engineering Research, 12(11): 2859–2865.

[9] Pattnaik, A. and Pattnaik, R. 2019. Optimal Sequence for MIMO Radar using Modified PSO Algorithm. International Journal of Innovative Technology and Exploring Engineering, 8(12): 2310–2314.

[10] Chitra, D. and Selvi, V. 2015. Design and Implementation of Quadriphase Sequences with Good Correlation Properties. Indian Journal of Science and Technology, 8(24): 1–5.

[11] Weng, L. and Gong, Y. 2005. A Class of Polyphase Codes with Good Correlation Properties Based on Frank Sequences. IEEE Transactions on Aerospace and Electronic Systems, 41(1): 354–358.