International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 11 (2017) pp. 2907-2915
© Research India Publications. http://www.ripublication.com

Design and Implementation of Four-Phase Sequences on FPGA using Modified Particle Swarm Optimization for Radar Applications

Mr. S. Srinivasa Rao[1]
Department of Electronics and Communication Engineering, Mahatma Gandhi Institute of Technology, Hyderabad, Telangana-500075, India.

[1]***ORCID: 0000-0002-4497-8137***

Dr. P. Siddaiah[2]
Department of Electronics and Communication Engineering, Acharya Nagarjuna University College of Engineering and Technology, Guntur, Andhra Pradesh- 522510, India.

Abstract:

Radar applications require sequences with individually peaky autocorrelation and low cross-correlation. Obtaining such sequences is a combinatorial problem. Thus the problem of signal design mentioned above is a challenging task for which many global optimization algorithms like genetic algorithm, particle swarm optimization algorithm, simulated annealing, tunneling algorithm were reported in the literature. The paper aims at the design of optimal set of Polyphase Sequences using Modified Particle Swarm Optimization (MPSO) Algorithm which makes use of Hamming Scan Algorithm (HSA) for Mutation. The Algorithm is implemented on FPGA to obtain real-time performance. The proposed Algorithm provides better performance over Modified Genetic Algorithm [9] in terms of speed and hardware requirement.

Keywords: Auto-Correlation, Optimization, Modified Particle Swarm Optimization (MPSO), Modified Genetic Algorithm (MGA), Hamming Scan Algorithm (HSA).

INTRODUCTION

The polyphase signals find application in the high-frequency applications such as RADAR and the communication. The polyphase codes have more than two phases. The polyphase codes have better Doppler tolerance than the binary phase codes. Some of the polyphase codes examples are as follows frequency derived codes, frank code, Lewis and Kretschmer code, etc. The polyphase frank code finds application in the omnidirectional RADAR systems [8]. The major task in the design of the polyphase code is to derive the code with better coherence and correlation. The design of the polyphase code with the better coherency and the correlation can be taken as the optimization problem [8]. Some of the algorithms such as genetic algorithm (GA) and the particle swarm optimization (PSO) have been used in the existing papers [3-8] to improve the coherency and the correlation of the polyphase code. The existing models design the polyphase code by defining an individual fitness value with the use of the objective function. It searches the optimal solution from the legal area [2]. Unlike in genetic algorithms, in PSO, there is no selection operation which increases the speed and reduces complexity of the algorithm. As the sequences length increase, the genetic algorithm consumes more time. Hence global optimization techniques such as

PSO algorithm is used for the generation of sequences with good correlation properties.

POLY PHASE CODES

Consider a set of polyphase sequences of length N bits which can be is represented by the given complex number sequence

$$\{S_l(n) = e^{j\varnothing_l(n)},\ n = 1, 2, 3, \ldots\ldots, N\},\qquad l = 1,2,3, \ldots\ldots, L \tag{1}$$

where, $\varnothing_l(n)$ is the phase of nth bit of the sequence which lies between 0 to 2π and L is the set size. The phase of the polyphase code signal has M number of phases. The phases of the polyphase coded signal can only have the following admissible values.

$$\varnothing_l(n) \in \left\{0, \frac{2\pi}{M}, 2\frac{2\pi}{M}, \ldots\ldots, (M-1)\frac{2\pi}{M}\right\} \tag{2}$$
$$= \{\Psi_1, \Psi_2, \Psi_3 \ldots\ldots\ldots, \Psi_M\}$$

For example, if M=4, the values of $\{\Psi_1, \Psi_2, \Psi_3$ and $\Psi_4\}$ will be 0, $\pi/2$, π and $3\pi/2$ respectively.

Consider a polyphase code set S having code length N, set size L, and distinct phase number M, the phase values of S is given by the following L by N phase matrix.

$$S(L,N,M) = \begin{bmatrix} s_1(n) \\ s_2(n) \\ s_3(n) \\ \cdot \\ \cdot \\ s_l(n) \\ s_L(n) \end{bmatrix} = \begin{bmatrix} \emptyset_1(1) & \emptyset_1(2) & \emptyset_1(3) & \dots\dots\dots \emptyset_1(N) \\ \emptyset_2(1) & \emptyset_2(2) & \emptyset_2(3) & \dots\dots\dots \emptyset_2(N) \\ & & \vdots & \\ \emptyset_L(1) & \emptyset_L(1) & \emptyset_L(1) & \dots\dots\dots \emptyset_L(N) \end{bmatrix} \quad (3)$$

where the phase sequence in row l ($1 < l < L$), is the four-phase sequence of signal l. All the elements in the matrix can be selected from the given set of phases specified in equation (2). The autocorrelation and cross-correlation of orthogonal four-phase codes have the following properties:

$$A(s_l, k) = \begin{cases} \frac{1}{N}\sum_{n=1}^{N-k} s_l(n)s_l^*(n+k) = 0, & 0 \le k < N \\ \frac{1}{N}\sum_{n=-k+1}^{N} s_l(n)s_l^*(n+k) = 0, & -N < k < 0 \end{cases} \quad (4)$$

and,

$$C(s_p, s_q, k) = \begin{cases} \frac{1}{N}\sum_{n=1}^{N-k} s_l(n)s_l^*(n+k) = 0, & 0 \le k < N \\ \frac{1}{N}\sum_{n=-k+1}^{N} s_l(n)s_l^*(n+k) = 0, & -N < k < 0 \end{cases} \quad (5)$$

$$p \neq q, \quad p, q = 1,2,\dots.L$$

where, $A(s_l, k)$ is the aperiodic autocorrelation function of the sequence s_l and $C(s_p, s_q, k)$ is the cross-correlation function of sequences s_p and s_q respectively.

A more practical approach to design a polyphase code set from the above equations (4) and (5) is to numerically search for the best polyphase sequences by minimizing a cost function that measures the degree to which a specific result meets the design requirements. For the design of polyphase code sets, used in radar applications, the cost function depends on the auto-correlation sidelobe peaks and cross-correlation peaks. Therefore, from equations (4) and (5) the cost function can be

$$E = \sum_{l=1}^{L} \left(\max_{k\neq 0} |A(s_l,k)|\right)^2 + \lambda \sum_{p=1}^{L-1}\sum_{q=p+1}^{L} \left(\max_{k} |C(s_p,s_q,k)|\right)^2 \quad (6)$$

where λ is the weighting coefficient between auto-correlation and cross-correlation.

MPSO ALGORITHM

The Particle Swarm Optimization algorithm is a novel population-based stochastic search algorithm and an alternative solution to the complex non-linear optimization problem. This algorithm optimizes a non-linear and multidimensional problem which usually obtains better solutions efficiently, requiring minimal parameterization. The PSO algorithm was first introduced by Dr. Kennedy and Dr. Eberhart in 1995 and it's basic idea was originally inspired by simulation of the social behavior of animals such as bird flocking, fish schooling and so on [1]. It is based on group communications to share individual knowledge when a group of birds or insects are in search of food or migrate and so forth in a searching space, although the birds or insects do not know where exactly the best position lies. But from the nature of the social behavior, if any of them finds a desired path to go, the rest of the members will follow automatically. Although PSO finds solutions much faster than most of the contemporary search techniques like Evolutionary and Genetic Algorithm, it may lose local optimum value because of the random search process. In the present work we have used HSA for mutation along with PSO. HSA searches in all directions in the vicinity of the point. In HSA, each element of the sequence is mutated with all other possible elements in the sequence. If the cost is reduced after mutation, then the new element is accepted, else the original element is retained. This recursive process is applied to all elements in the sequence. Thus, HSA performs search among all the neighbors of the sequence and selects local minimum.

WORKING OF MPSO ALGORITHM

In MPSO each particle keeps track of its coordinates in the solution space which is associated with the best solution (fitness) that it has achieved so far. This value is called *pbest* (p_i). Another best value that is tracked by the PSO is the best value obtained so far by any other particle in the neighbourhood of that particle. This value is called *gbest* (p_g). In this algorithm we have a completely connected swarm, that is, that all the particles share information, where every particle knows what is the best position ever visited by other particles in the swarm.

The basic steps of MPSO algorithm for the design of binary sequences is as follows

Step1: (Initialization): Generate initial particles by randomly generating the position and velocity for each particle.

Step 2: Evaluate each particle's cost (E), which is the ratio of the amplitude of main peak of the auto correlation to the absolute maximum amplitude of the side lobes.

Step 3: For each particle, if the cost (E) is less than its previous best (p_i) fitness, *pbest* should be updated.

Step 4: For each particle, if the cost (E) is less than the best of all particles (*gbest*), *gbest* should be updated.

Step5: For each particle

a) Generate a new particle according to the given equations

$$\boldsymbol{v_i[t+1] = w \cdot v_i[t] + c_1 r_1 \cdot (p_i[t] - x_i[t]) + c_2 r_2 \cdot (p_g[t] - x_i[t])} \quad (7)$$

$$\boldsymbol{x_i[t+1] = x_i[t] + v_i[t+1]} \quad (8)$$

where $v_i[t]$ and $v_i[t+1]$ are the previous and the current velocities of the i^{th} particle. Here $x_i[t]$ represents the position of the i^{th} particle. $0 \leq w < 1$ is an inertia weight which determines how much the previous velocity is preserved (chosen $w = 0.99$). This signifies that the previous velocity is almost preserved, but not completely, to avoid escaping from the optimum value. c_1 and c_2 are the accelerating constants assigned a random value picked between 0 and 1 from an uniform distribution, and finally r_1 and r_2 are uniformly distributed random numbers ranging between [0, 1].

b) Generate a new particle x^1 by mutating the elements in x with HSA.

c) Compare x with x^1 and select the one with low cost value.

Step6: If the stop criterion is satisfied, then stop, else go to Step 3.

PROPOSED ARCHITECTURE

In this paper, FPGA implementation of MPSO has been proposed for the synthesis of four-phase sequences with good correlation properties. Four-phase sequence includes the elements +1, -1 +i and –i. Because, the proposed architecture is targeted to FPGA, the elements are coded as 001, 011, 101 and 111 respectively. The implementation of MPSO includes Random Sequence Generator, Fitness Module, PSO module and Hamming Scan Module.

Random Sequence Generator (RSG):

Random sequences are generated using linear cellular automata (LCA). It is adopted to generate necessary individuals as it generates better random sequences. The most popular rules i.e Rule 150, is used to generate random sequences. Fig 1 shows the architecture of 8-bit RSG.

PSO Module:

For hardware implementation, the PSO algorithm is carried out by folowing operations: calculate fitness, update particle's best position, global best position, update the velocity and update position. The position for particle i, x_i is fetched from memory in the first stage. Then in the second stage, the fitness module evaluates fitness, $f(x_i)$ based on the position of particle i. The current *pbest* values, p_i and $f(p_i)$ are also fetched from *pbest* memory. In the third stage, the *gbest* and *pbest* are updated. Both the update *gbest* and *pbest* modules are passed x_i and $f(x_i)$. In addition, the update *pbest* module is passed p_i and $f(p_i)$ and the update *gbest* module is passed p_g and $f(p_g)$. Each module selects the lowest fitness and the associated positions for their output, which are the now updated *gbest* and *pbest*. In addition the old velocity, v_i is fetched from the velocity memory. Now in the fourth stage, the new p_i and $f(p_i)$ are stored in the *pbest* memory. The velocity update module uses v_i, x_i, p_i and p_g to evaluate the new velocity, v_i. In the fifth stage, the update position module uses x_i and v_i to evaluate the new position, x_i. The new velocity v_i, is stored in the velocity memory. In the last stage, the new position x_i is stored in the position memory. Because updating of *gbest* and *pbest* are done in parallel, the five operations can be implmented in a 6-stage pipeline, together with initial fetch and final write stages.

Fitness Evaluation

The procedure adopted here is, calculating the autocorrelation of individual sequences first, followed by the main lobe amplitude, maximum of all side lobes and the ratio of main lobe to the side lobe gives the fitness value. The above procedural computation has been done using the hardware structure shown in the Fig 3.

Hamming Scan Module

Hamming scan algorithm is a traditional optimization technique which searches in all directions in the vicinity of point to reduce the fitness value. It has a faster convergence rate over particle swarm optimization (PSO).

Fig 4 depicts the architectural block of hamming scan technique. [For example, in case of a four-phase sequence, the +1 can be mutated by -1, +1 and -i. Similarly, -1 can be mutated by +1, +i and -i and so on. Hence by hamming scan operation, three new sequences will be generated. For all these sequences side lobe amplitudes are computed

and compared by a comparator. Finally, the sequence with optimal ASP value is selected.

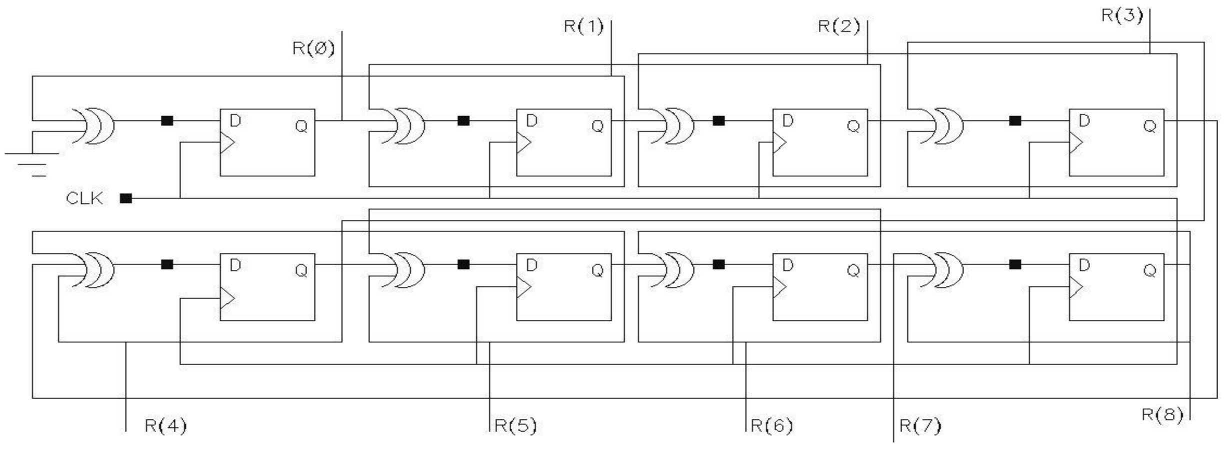


Figure 1: Random Sequence Generator

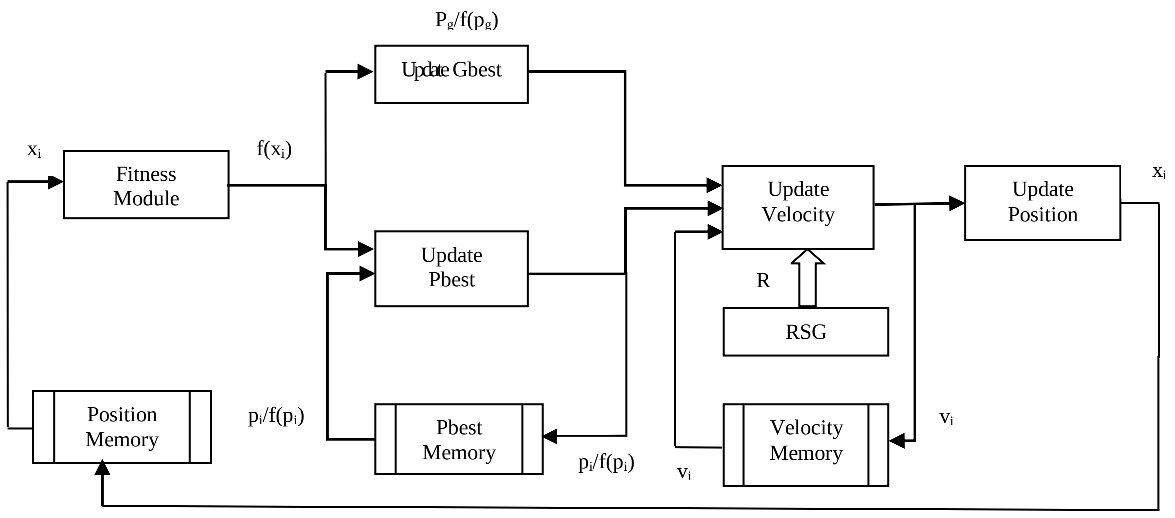


Figure 2: PSO Module

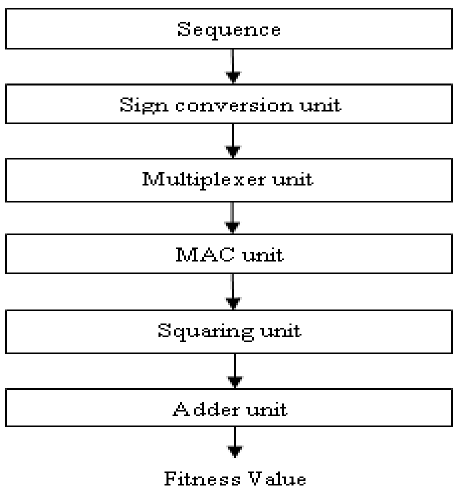


Figure 3: Fitness Computation Module

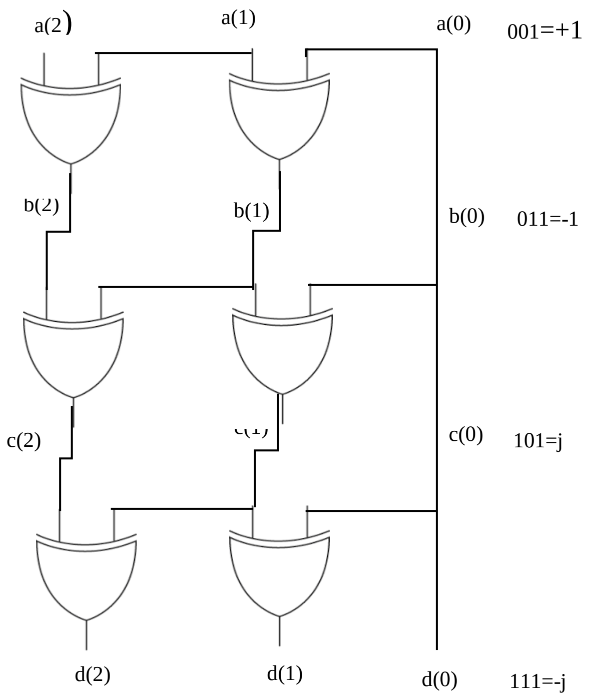


Figure 4: Hamming Scan Module

TECHNOLOGY AND TOOLS

The modules in MPSO are developed using Verilog HDL. The hardware design has been simulated, synthesized and implemented on Xilinx Spartan-6 FPGA by the tool Xilinx ISE XST foundation 14.1. The synthesis tool has been configured to minimize chip area and to obtain high performance.

RESULTS AND DISCUSSIONS

The hardware implementation of proposed algorithm (MPSO) based on FPGA has been used for generation of Four-phase sequences to verify its functionality. The simulation waveform obtained for an optimal four-phase sequence of length 16 is shown in the Fig 5 and the sequence is 111 11 101 001 011 101 101 101 001 101 111 101 0111 011 111 011 (-i -1 i 1 -1 i i i 1 i -i i -1 -1 –i 1). The side lobe amplitude of above sequence is 2. The top level schematic diagram of MPSO after synthesis is shown in Fig 6. The hardware utilization report of an optimal sequence of length 16 is presented in the Table 2. It can be observed that the hardware utilization is very low (Registers 1% and LUTs 4%), so the same FPGA can be used for the implementation of higher lengths of the codes. The timing analysis of proposed algorithm is shown in Table 4 and it indicates that the performance of MPSO is better than the reported in literature [9]. The auto-correlation and cross-correlation values of poly phase codes are shown in Tables 1 and 2 and the cross-correlation values obtained using proposed algorithm is far better than reported in literature [8-9].The average ASP and CP values of sequences are shown in Fig 7. The normalized ASP and CP values for N=40 is about 0.14 and 0.2 and for N=128 is 0.09 and 0.11 respectively.

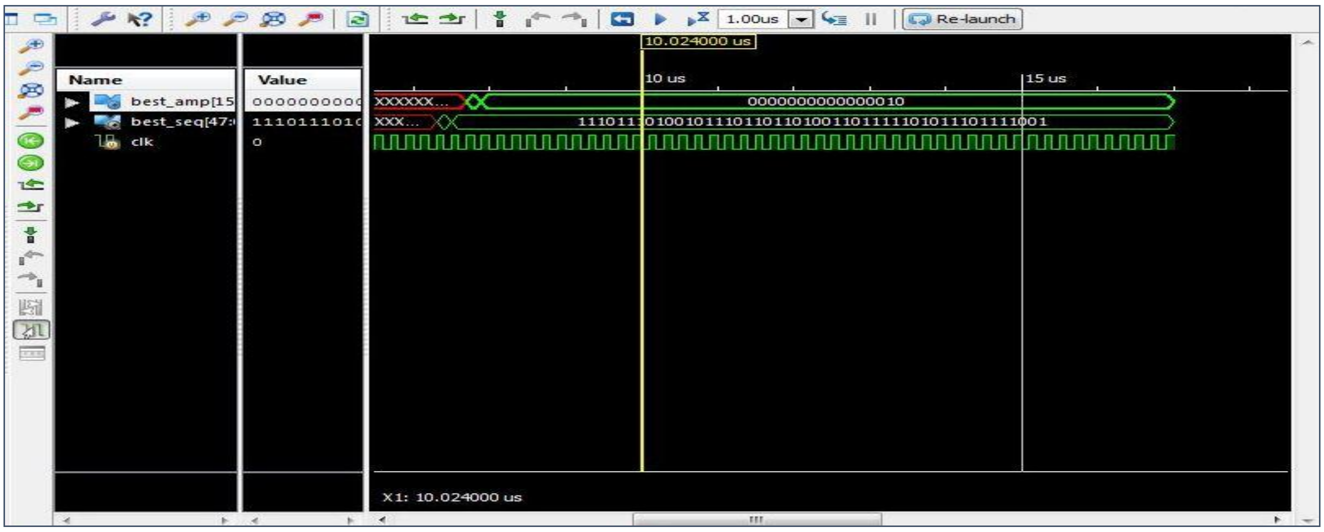


Figure 5: Simulation Wave Form for a Sequence of Length 16

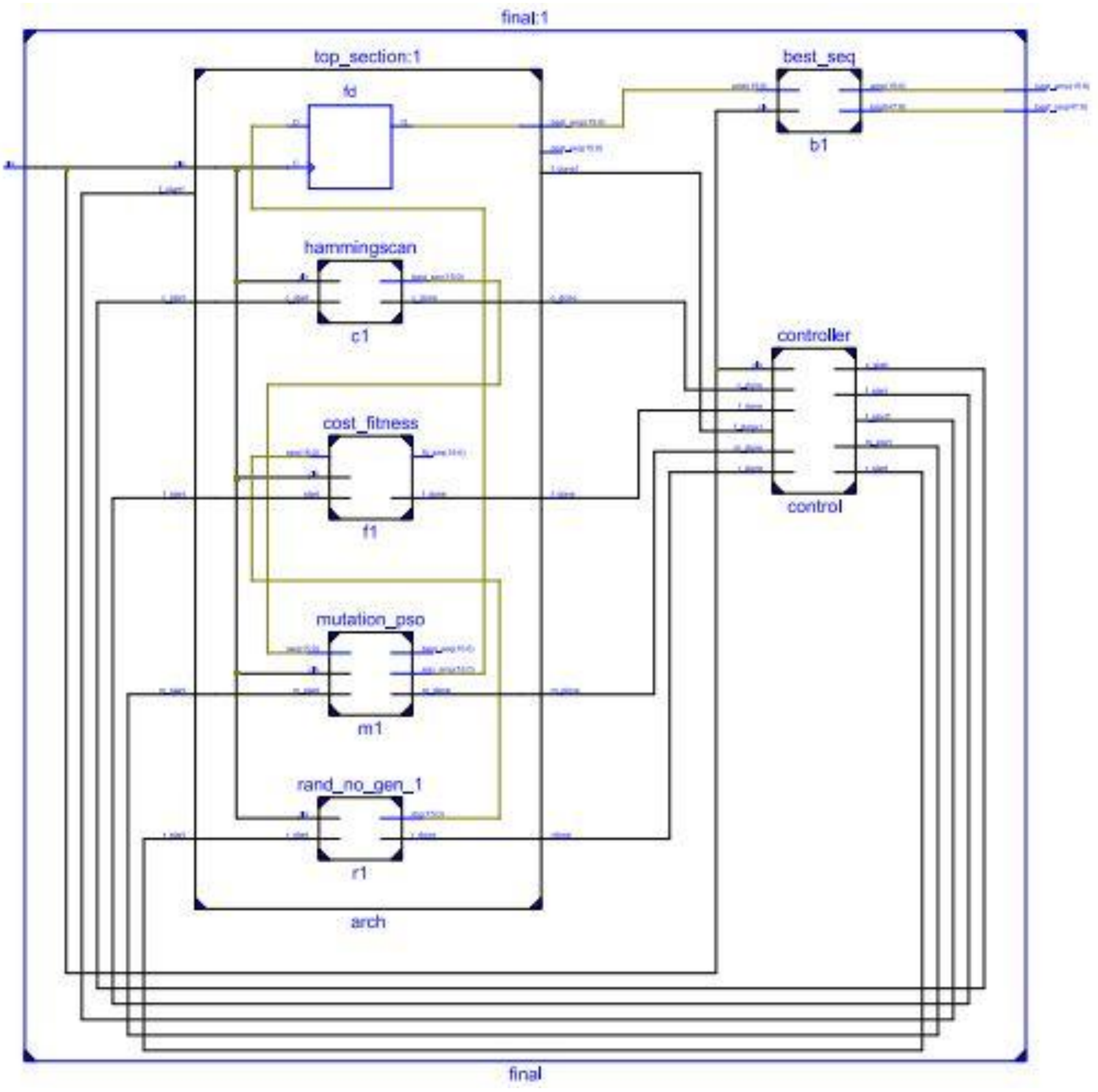


Figure 6: Top Level Schematic Diagram of MPSO

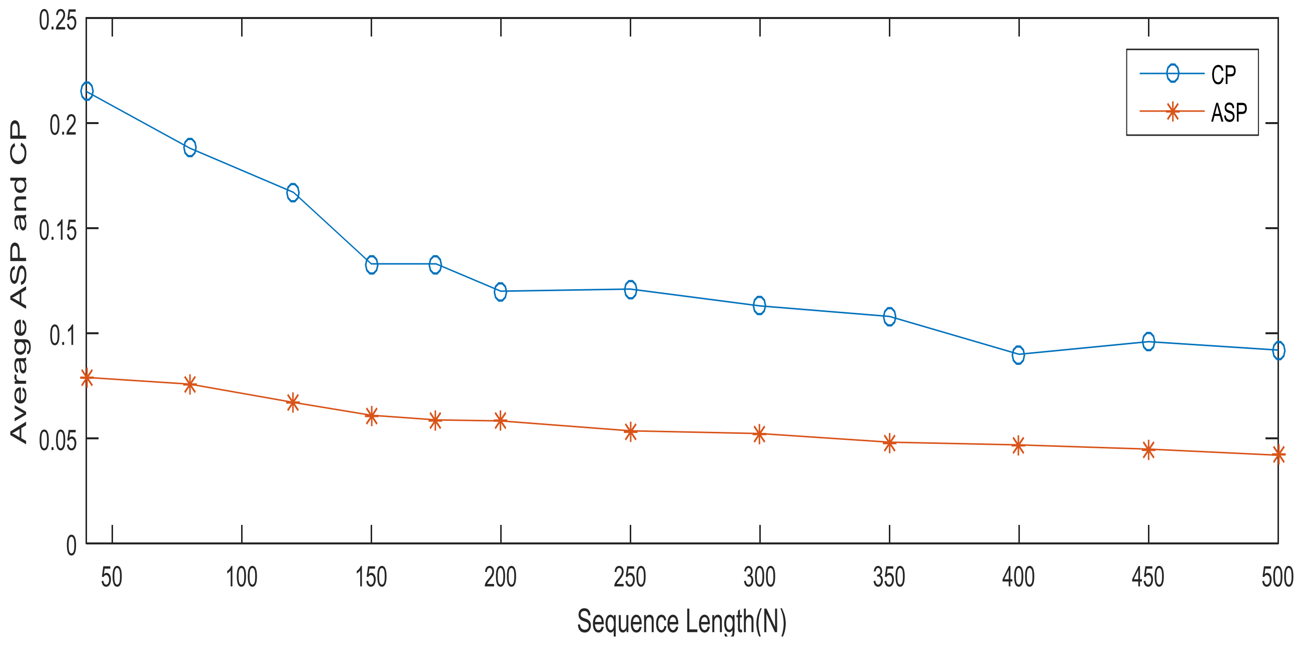


Figure 7: Average value of Autocorrelation sidelobe peaks (ASPs) and Cross-Correlation (CPs) Peaks of sequences for various values of N.

Table 1: Autocorrelation Sidelobe Peaks (Diagnol Terms) and Cross-Corellation Peaks (Off-Diagnol Terms) for Synthesized Codes with N=40, L=3 and M=4.

Codes	Code 1	Code 2	Code 3
Code 1	0.1451	0.2012	0.1902
Code 2	0.2012	0.1411	0.2021
Code 3	0.1902	0.2021	0.1346

Table 2: Autocorrelation Sidelobe Peaks (Diagnol Terms) and Cross-Corellation Peaks (Off-Diagnol Terms) for Synthesized Codes with N=128, L=3 and M=4.

Codes	Code 1	Code 2	Code 3
Code 1	0.0841	0.1094	0.1119
Code 2	0.1094	0.0891	0.1105
Code 3	0.1119	0.1105	0.0941

Table 3: Hardware utilization summary for a sequence of length 16.

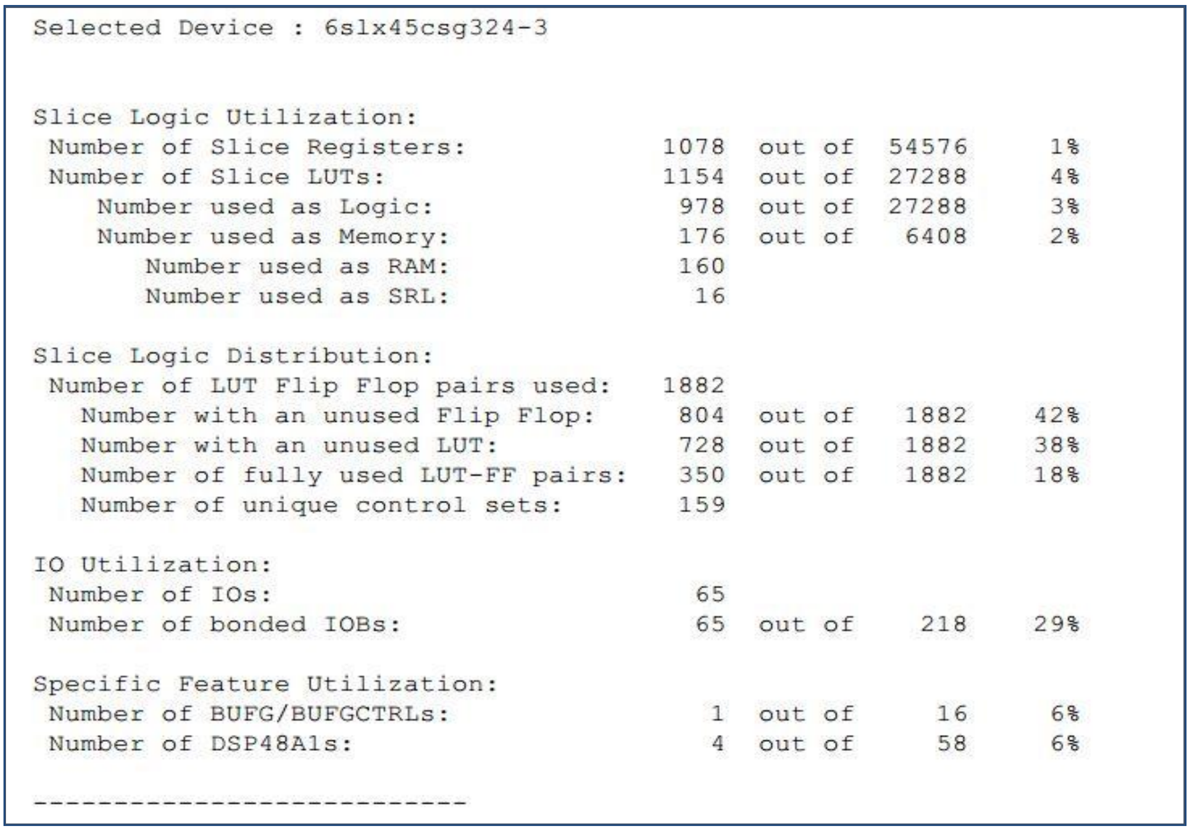
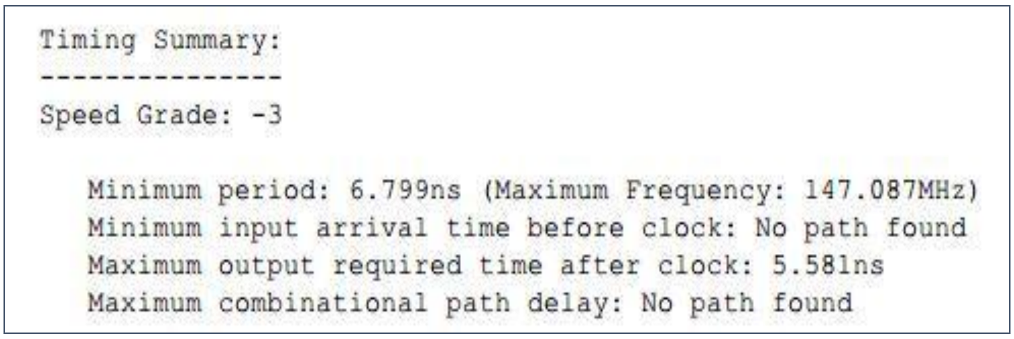

```
Selected Device : 6slx45csg324-3


Slice Logic Utilization:
 Number of Slice Registers:            1078  out of  54576     1%
 Number of Slice LUTs:                 1154  out of  27288     4%
    Number used as Logic:               978  out of  27288     3%
    Number used as Memory:              176  out of   6408     2%
       Number used as RAM:              160
       Number used as SRL:               16

Slice Logic Distribution:
 Number of LUT Flip Flop pairs used:   1882
   Number with an unused Flip Flop:     804  out of   1882    42%
   Number with an unused LUT:           728  out of   1882    38%
   Number of fully used LUT-FF pairs:   350  out of   1882    18%
   Number of unique control sets:       159

IO Utilization:
 Number of IOs:                          65
 Number of bonded IOBs:                  65  out of    218    29%

Specific Feature Utilization:
 Number of BUFG/BUFGCTRLs:                1  out of     16     6%
 Number of DSP48A1s:                      4  out of     58     6%

---------------------------
```

Table 4: Timing Analysis


```
Timing Summary:
---------------
Speed Grade: -3

   Minimum period: 6.799ns (Maximum Frequency: 147.087MHz)
   Minimum input arrival time before clock: No path found
   Maximum output required time after clock: 5.581ns
   Maximum combinational path delay: No path found
```

CONCLUSION

The objective of this paper is mainly to demonstrate the significance of the MPSO algorithm in the generation of Four-phase sequences with good correlation values. These sequences are widely used in radar and spread spectrum communications for improving system performance. The results obtained indicate that the proposed algorithm (MPSO) outperforms the existing algorithms like Genetic Algorithm. Unlike in Genetic Algorithms, in PSO, there is no selection operation which increases the speed and reduces complexity of the algorithm. As the sequences length increase, the genetic algorithm consumes more time. Hence global optimization techniques such as PSO algorithm are used for the generation of sequences with good correlation properties.

REFERENCES

[1]. J.KENNEDY and R. EBERHART, “Particle swarm optimization,” in Proc. IEEE Int. Conf. Neural logical Networks, vol. IV, Perth, ustralia, 1995, pp. 1942–press.1948.

[2]. Xiangneng Zeng, Yongshun Zhang, and Yiduo Guo, "Polyphase coded signal design for MIMO radar

using MO-MicPSO," Journal of Systems Engineering and Electronics, vol.22, no.3, pp.381–386, June 2011.

[3]. Sergio Gil-López, Javier Del Sera, Sancho Salcedo-Sanz, Ángel M. Pérez-Bellido, José Marı´a Cabero and José A. Portilla-Figueras, "A hybrid harmony search algorithm for the spread spectrum radar polyphase codes design problem," Expert Systems with Applications,vol.39, no.12, pp.11089–11093,15 September 2012.

[4]. Xiu Zhang and Xin Zhang, "A novel artificial bee colony algorithm for radar polyphase code and antenna array designs," EURASIP Journal on Wireless Communications and Networking, December 2016.

[5]. Li Hong, Qin Yuliang, Wang Hongqiang, Li Yanpeng and Li Xiang ,"A Fast Parameter Estimation Algorithm For Polyphase Coded CW Signals," Journal Of Electronics(China), vol.28, no.1, January 2011.

[6]. Paolo Ghelfi, Filippo Scotti, Francesco Laghezza, and Antonella Bogoni, "Phase Coding of RF Pulses in Photonics-Aided Frequency-Agile Coherent Radar Systems," IEEE Journal of Quantum Electronics, vol.48, no.9, pp. 1151 - 1157, 2012.

[7]. Antonio De Maio, Silvio De Nicola, Yongwei Huang, Zhi-Quan Luo, and Shuzhong Zhang, "Design of Phase Codes for Radar Performance Optimization With a Similarity Constraint," IEEE Transactions on Signal Processing, vol. 57, no. 2, pp.610 - 621, Nov, 2008.

[8]. S. P Singh, and K Subba Rao, "Orthogonal Polyphase Sequence Sets Design for Radar Systems," In Proceedings of the International Radar Symposium, pp. 1-4, 2006.

[9]. N.Balaji and K.Subba Rao, "VLSI-based Real-time Signal Processing Solution Employing Four-phse Codes for Spread-spectrum Applications", IETE Journal of research, Vol.58, Issue 1, pp. 57-64, Jan-Feb 2012

[10]. Merrill I Skolnik, "Introduction to Radar Systems," 2nd edition, 2008.

[11]. Nadav Levanon, Eli Mozeson, "Radar Signals", 1.st Editon Wilet-Interscience,2004.

[11] Changhe Li at.el, A Fast Particle Optimization Algorithm with Cauchy Mutation and Natural Selection Strategy. Proceedings of the 2nd international conference Advances in computation and Intelligence 2007, Wuhan, China, pp.334-343.

[12] Lindner, J. (1975) Binary sequences up to length 40 with best possible Autocorrelation functions. Letters, 11(1975), 507.

[13] Molin Jia, Xiao Cai, Syahrulanuar Ngah, Yuji Tanabe and Takaaki Baba, "A Pipeline Architecture of Particle Swarm Optimization for Real-Time Control," Journal of Signal Processing, Vol. 14, No. 6, pp. 405-414, Nov. 2010.

[14] H. Zhu, Y. Tanabe and T. Baba, "A random time-varying particle swarm optimization for the real time location systems", IEEE Transaction on Electronics, Information and Systems, Vol. 128-C, No. 12, pp. 1747-1760, 2008

ABOUT THE AUTHORS

Mr. S. Srinivasa Rao obtained his B. Tech degree in Electronics and communication Engineering from RVR & JC engineering college, Guntur in 1999 and M. Tech degree in" Digital Systems and Computer Electronics" from JNTU, Hyderabad, in 2003. Presently he is working as an Associate professor in the Dept of ECE of Mahatma Gandhi Institute of Technology, Gandipet, Hyderabad. Mr. Srinivasa Rao has more than 14 years of teaching experience. He is pursuing his Ph.D from Acharya Nagarjuna University, Guntur in Radar Signal Design. He is the member of ISTE. His areas of interests are Radar Signal Processing, Embedded Systems & Optimization Algorithms.

Dr. P. Siddaiah obtained B. Tech degree in ECE from JNTU Anantapur College of Engineering in 1988. He received his M. Tech degree from S. V. University, Tirupati. He did his Ph. D program in JNTU, Hyderabad. He is the chief investigator for several outstanding projects sponsored by Defence organizations, AICTE, UGC and ISRO. He is currently working as principal and professor in ECE Department of university college of Engineering and Technology, Acharya Nagarjuna University, Guntur, India. Several members successfully completed their Ph. D under his guidance. He has published several papers in National and International journals and Conferences. He is the life member of FISTE, IE and MISTE.