Minimization of Average Power Consumption in 3 Stage CMOS Ring Oscillator based on MSFLA, Fuzzy-MSFLA, GA, and Fuzzy-GA

Farshid Keivanian
Master of Science in Electronics Engineering
Department of Electrical and Computer Engineering
University of Birjand, South Khorasan, Birjand, Iran

ABSTRACT
This article is based on the application of heuristic algorithms to minimize the average power consumption in a VLSI circuit. The idea is to find the optimum layout and temperature for a 3 stage ring oscillator with minimal dynamic average power. The objective function is the same as average power ($P_{avg}$) of 3 stage ring oscillator with 6 CMOS inverters that depends on the temperature and the two different group of channel widths for NMOSs and PMOSs. ($W_1=W_3=W_5$ and $W_2=W_4=W_6$). These parameters make a three dimensional search space which is explored by search agents of algorithms. Motivated by the convergence of Modified Shuffled Frog Leaping Algorithm (MSFLA), Genetic Algorithm (GA) and the link of MATLAB with HSPICE Software the minimized average power of 3 stage ring oscillator is obtained. Based on MSFLA, Fuzzy-MSFLA, GA, and Fuzzy-GA algorithms the best resulting for $P_{avg}$ in 0.18µm Technology and the supply voltage of 5v is 1.19 µW based on Fuzzy-MSFLA.

General Terms

Keywords
3 stage CMOS ring oscillator, minimal average power, optimum layout and temperature, MSFLA, Fuzzy-MSFLA, GA, and Fuzzy-GA.

1. INTRODUCTION
Oscillators are used for synchronizing computation in a digital system, timing the sampling in a data converter, carrier synthesis and LO in RF systems [1]. The ring oscillators are widely used in integrated circuits. They are being used be semiconductor foundries to monitor the power dissipation, delay, and jitter of fabricated CMOS inverters. On the basis of the result of measured frequency pattern the IC is accepted or rejected. Ring oscillators also occupies less chip area as they do not have inductor as compared to LC tank oscillators [2]. The application of oscillators ranges from clock generation in microprocessor to carrier synthesis in cellular telephones requiring vastly different oscillator topologies and performance parameter [3]. Recently the explosive growth in wireless communication and the advances in complementary metal oxide semiconductor (CMOS) technology made it possible to implement high frequency oscillator with CMOS technology (MR20768). The oscillator are required to be tunable over a relatively wide frequency rang. The tenability is usually obtained by variable voltage and hence comes the name Voltage Controlled Oscillator (VCO). VCOs are key component in frequency synthesizer for Radio Frequency (RF) wireless application [4]. Voltage Controlled Oscillators (VCOs) constitute a critical component in many RF transceivers and are commonly associated with signal processing tasks like frequency Selection and signal generation. There are two types of VCOs such as I) Waveform oscillators: ring oscillator topology and relaxation oscillator (which has poor phase noise performance) II) Resonant oscillators: LC tank oscillator topology and Crystal oscillator (neither integrated nor tunable) [2]. The ring oscillator topology has the ability of easy integration with the absence of passive elements which reduces the die area. This property changes the ring oscillators to the favorite kind of oscillators in communication systems [5,6,7]. In this paper the heuristic algorithms such as Modified Shuffled Frog Leaping Algorithm (MSFLA) [8,9], Fuzzy-MSFLA, GA, and Fuzzy-GA are employed to find the best channel widths and temperature in which the 3 stage ring oscillator consumes the lowest average power.

2. THREE STAGE CMOS RING OSCILLATOR
The ring oscillator which is proposed in this article is comprised of three delay stages, with the output of the last stage fed back to the input of the first. To provide the DC inversion, an odd number of stages must be used. To see why 3 stage ring circuit will oscillate, assume that the output of the first inverter is a ‘0’. Therefore, the output of the Nth (3th) inverter, where N (N=”3”) is odd, must also be ‘0’. However, this output is also the input to the first inverter, so the first inverter’s output must switch to a ‘1’. By the same logic, the output of the last inverter will eventually switch to a ‘1’, switching the output of the first inverter back to ‘0’. This process will repeat indefinitely, resulting in the voltage at each node oscillating [10]. The 3 stage CMOS ring oscillator using single ended inverters is shown in Figure 1 and 2.

Fig 1: The 3 Stage Ring Oscillator using single ended inverters
by increasing temperature T the mobility of electrons, $\mu_n$ increases [14]. The effect of temperature T and channel widths W on average power, $P_{avg}$ are shown in Relation 7. Thus by changing the parameters of channel widths (W) and temperature (T) the average power consumption ($P_{avg}$) varies. In this article all parameters such as NMOS and PMOS channel widths and temperature are defined as independent variables that make a search space in which the optimum solution (value) for NMOS channel widths ($W_1=W_3=W_5$), PMOS channel widths ($W_2=W_4=W_6$), and the temperature (T) is found. These solutions meet the least average power value. The pseudo code for average power is illustrated in Box 1.

**Box 1. The Pseudo-Code of Objective Function which is defined in this Article for MSFLA and GA Algorithms**

- Open the input netlist file ('T3b.sp')
- Seek the NMOSs, PMOSs' channel widths and temperature and allocate the positions of search agents to them ("Assign the Position(1) to W1, W3, W5., Position(2) to W2, W4, and W6, and Position(3) to Temperature value")
- Run HSPICE as a fitness evaluator with input Netlist File ('T3b.sp')
- Read data from output file ('T3b.lis') and seek the average power ('avgpower') from the output file ('T3b.lis') and return it to the main program that has called this objective function

### 3.2 Shuffled Frog Leaping Algorithm (SFLA)

The Shuffled Frog Leaping Algorithm (SFLA) which is firstly proposed by Eusuff and Lansey. The overall structure in SFLA is shown in Figure 3 which is the same as structure of shuffled complex evolution (SCE-UA) [15-18]. The flowchart of SFLA in Figure 4 is similar to the operation of SCE-UA except that the frog leaping algorithm (FLA) is used as local search in SFLA while in SCE-UA for local search the competitive complex evolution (CCE) is used. One of differences between FLA and CCE is that the parameter of maximum search step, $S_{max}$ is defined in FLA to limit the length of search step because it is randomly produced. The vector calculations for producing solutions in standard SFLA are in Relations 8, 9, and 10 and they are shown in Figure 5.

\[
S = \text{round} \left( \text{rand} \times C \times (X - Pw) \right), X \in \{P_B, P_C \}, C > 0 \quad (8)
\]

\[
|S| = \min \left( |S|, S_{\text{max}} \right) \quad (9)
\]

\[
U_q = Pw + S \quad (10)
\]

The Relation 8 shows that X is either the best solution in complex, PB, or the best global solution, PX, and the deference of X and the worst solution of complex, Pw, the search step of S is obtained. The parameter C commonly equals to 2 (C=2). The maximum search step length, $S_{max}$ is obtained in Relation 9 and finally the new vector of solution $U_q$ is obtained in 10. In Figure 3 the allocation of population to complexes, selection of q members of complexes with triangular distribution of $\pi_i$, modification or improvement of the last member in each of complexes, placement of improved complex in the main population and sorting the populations in the new complex, then repeat the entire process for all complexes. The results of global search are used in local search. Figure 5 shows the sampling process in SFLA.
The process of sampling in the search space are determined by probability distribution. For example, if \( B = X_k \) and \( D = X_{m+1} \), then the best solution \( P_B \) in complex \( B \) is evolved towards the global best solution \( P_X \). If \( B = X_q \) and \( D : X_{q'} \), the best solution \( P_B \) in complex \( B \) is evolved towards \( P_X \). A random solution is created in search space if the best solution in complex \( B \) is not better than the global best solution.

**Fig 3:** The overview of SFLA

**Fig 4:** The Flowchart of SFLA

**3.3 Modified Shuffled Frog Leaping Algorithm (MSFLA)**

All modified versions of SFLA (MSFLAs) maintain the general structure of SFLA. Some of the recent articles propose some changes in parameters like \( n \), number of solutions in each complex and others that lead to slight improvement. In other modified versions of SFLA (MSFLA) the parts of local search and the process of sampling in the search space are modified [19-21].

### 3.3.1 Modified Shuffled Frog Leaping Algorithm with Cognitive Behavior

The Modified SFLA (MSFLA) with cognitive behavior is partly based on PSO algorithm as there are two components of individual factor \( C_1 \) and the social factor \( C_2 \) in both of them. In Relation 11 the social factor \( C_2 \) is multiplied by the difference of solution \( W \) with the best solution in complex \( P_X \) or with the best global solution in population, \( P_B \). The individual factor \( C_1 \) is multiplied by the difference of the solution \( W \) and the best solution of complex \( P_{W_{best}} \) that is experienced. Parameters \( r_1 \) and \( r_2 \) are randomly produced in MATLAB and have uniform probability distribution. For applying the maximum step length the way of standard SFLA is also used for Relation 12. The candidate solution is made by adding the search step with the position of solution \( S \) in Relation 13.

\[
S = round \left( r_1 c_1 W_{best} - P_W + r_2 c_2 (X - P_W) \right) \tag{11}
\]

\[
r_1, r_2 \sim U(0, 1), X \in \{ P_B, P_X \}
\]

\[
S = \min (S, S_{max}) \tag{12}
\]

\[
U_q = P_W + S \tag{13}
\]

After implementation of Modified SFLA with cognitive behavior on the objective function which is defined the average power consumption in 3 stage ring oscillator the Figures 6 is obtained. It shows the convergence of algorithm with the least average power of 2.4 * 10^6 watt for the specific values of temperature, NMOS channel widths, and PMOS channel widths that are all shown in Table 1.

---

**Table 1:**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Temperature</td>
<td>300K</td>
</tr>
<tr>
<td>NMOS channel widths</td>
<td>10nm</td>
</tr>
<tr>
<td>PMOS channel widths</td>
<td>10nm</td>
</tr>
</tbody>
</table>
The Relation 14 shows the total size of population in S. Also the maximum search step, $S_{max}$, is obtained through the
Relation 15.

$$S = (\text{maxVars} - \text{minVars})$$  \hspace{1cm} (14)

$$S_{max} = 0.45 \times (\text{maxVars} - \text{minVars})$$  \hspace{1cm} (15)

The probability of selection of the members of the complex A to be evolved is shown in Relation 16.

$$p_i \leftarrow \frac{2(n+1-i)}{n(n+1)}, \quad i = 1, \ldots, n$$  \hspace{1cm} (16)

The pseudo code for the function of evolve_towards_pxp is shown in Box 2. These evolutions are based Relations 11-13.

**Box 2. Pseudo-Code of Evolve_Toward_PxPB Function**

- Set parameters $C_1 = 1$, $C_2 = 2$
- Calculate the search step : $S = \text{max} (\text{min}(C_1 \ast \text{rand} \ast (\text{Pw.Pbest.X} - \text{Pw.X}) + \ldots, C_2 \ast \text{rand} \ast (\text{Pb.X-Pw.X}),S_{max}) - S_{max})$ (based on relations 11, 12)
- Add the search step to the position of the solution to produce a new solution : (based on relation 13)
- Return the produced new random solution to the main program that has called this function

The improvements of SFLA are colored in purple within pseudo codes in Box 3.

**Box 3. Pseudo-Code of MSFLA with Cognitive Behavior**

- Set the SFLA parameters for the problem of 3 Stage Ring Oscillator, nVars = 3 (Number of Variables), maxVars = [2 2 50] as upper band of NMOS W, PMOS W, and temperature, minVars=[0.2 0.2 1] as lower band of NMOS W, PMOS W, and temperature, Smax=0.45*(maxVars-minVars), Cost Function=Objective Function (“defined in Box1”), m = 5 (number of complexes), n = 5 (number of solutions in each complex), S = m * n, q = 4 (number of members in sub-complex), Ns = 2 (number of shuffling), Nt = 1 (number of iteration for evolution), nIter = 10 (total number of iteration for main loop, VTR = 0 (the lowest value to reach)
- Generate Initial Population, Evaluate the cost value, make a memory for the best solution in each complex by now
- Do While iteration number (it) < Maximum Iteration (nIter) and best cost > VTR
  - Sort population and $P_X$ is determined. ($P_X = \text{pop}(1)$)
  - For $k = 1$ to m
    - Allocation of the population into k complexes , $A_k$ (Complex Evolution : FLA)
    - For $j = 1$ to Nt
      - Evolve Pw towards $P_B$ (PB: The best global solution)
        - and produce new solution $r$ (Call evolve_towrads_pxp function)
        - if solution $r$ is not within minVars, maxVars then
          - Evolve Pw towards $P_X$ (The best position in complex by now, based on memory content) and produce new solution $c$ (Call evolve_toward_pxp function)
          - if solution $c$ is not within minVars, maxVars then
            - Create new random solution $z$
            - Update the $P_B$ and $P_X$
  - End Ns
- End Do
- Plot Best Costs

3.4 Fuzzy Modified Shuffled Frog Leaping Algorithm with Cognitive Behavior (Fuzzy-MSFLA)

By controlling the parameter $C_1$ (individual factor) and $C_2$ (global factor) the performance of MSFLA is improved. The Fuzzy Rules for this purpose are shown in Table 2.

<table>
<thead>
<tr>
<th>W1=W3=W5 (µm)</th>
<th>W2=W4=W6 (µm)</th>
<th>Temperature (centigrade)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.6657126869238</td>
<td>0.63834494370498</td>
<td>46.533917536174</td>
</tr>
</tbody>
</table>

The Relation 14 shows the total size of population in S. Also the maximum search step, $S_{max}$, is obtained through the
Relation 15.

$$S = (\text{maxVars} - \text{minVars})$$  \hspace{1cm} (14)

$$S_{max} = 0.45 \times (\text{maxVars} - \text{minVars})$$  \hspace{1cm} (15)

The probability of selection of the members of the complex A to be evolved is shown in Relation 16.

$$p_i \leftarrow \frac{2(n+1-i)}{n(n+1)}, \quad i = 1, \ldots, n$$  \hspace{1cm} (16)

The pseudo code for the function of evolve_towards_pxp is shown in Box 2. These evolutions are based Relations 11-13.

**Box 2. Pseudo-Code of Evolve_Toward_PxPB Function**

- Set parameters $C_1 = 1$, $C_2 = 2$
- Calculate the search step : $S = \text{max} (\text{min}(C_1 \ast \text{rand} \ast (\text{Pw.Pbest.X} - \text{Pw.X}) + \ldots, C_2 \ast \text{rand} \ast (\text{Pb.X-Pw.X}),S_{max}) - S_{max})$ (based on relations 11, 12)
- Add the search step to the position of the solution to produce a new solution : (based on relation 13)
- Return the produced new random solution to the main program that has called this function

The improvements of SFLA are colored in purple within pseudo codes in Box 3.

**Box 3. Pseudo-Code of MSFLA with Cognitive Behavior**

- Set the SFLA parameters for the problem of 3 Stage Ring Oscillator, nVars = 3 (Number of Variables), maxVars = [2 2 50] as upper band of NMOS W, PMOS W, and temperature, minVars=[0.2 0.2 1] as lower band of NMOS W, PMOS W, and temperature, Smax=0.45*(maxVars-minVars), Cost Function=Objective Function (“defined in Box1”), m = 5 (number of complexes), n = 5 (number of solutions in each complex), S = m * n, q = 4 (number of members in sub-complex), Ns = 2 (number of shuffling), Nt = 1 (number of iteration for evolution), nIter = 10 (total number of iteration for main loop, VTR = 0 (the lowest value to reach)
- Generate Initial Population, Evaluate the cost value, make a memory for the best solution in each complex by now
- Do While iteration number (it) < Maximum Iteration (nIter) and best cost > VTR
  - Sort population and $P_X$ is determined. ($P_X = \text{pop}(1)$)
  - For $k = 1$ to m
    - Allocation of the population into k complexes , $A_k$ (Complex Evolution : FLA)
    - For $j = 1$ to Nt
      - Evolve Pw towards $P_B$ (PB: The best global solution)
        - and produce new solution $r$ (Call evolve_towrads_pxp function)
        - if solution $r$ is not within minVars, maxVars then
          - Evolve Pw towards $P_X$ (The best position in complex by now, based on memory content) and produce new solution $c$ (Call evolve_toward_pxp function)
          - if solution $c$ is not within minVars, maxVars then
            - Create new random solution $z$
            - Update the $P_B$ and $P_X$
  - End Ns
- End Do
- Plot Best Costs

3.4 Fuzzy Modified Shuffled Frog Leaping Algorithm with Cognitive Behavior (Fuzzy-MSFLA)

By controlling the parameter $C_1$ (individual factor) and $C_2$ (global factor) the performance of MSFLA is improved. The Fuzzy Rules for this purpose are shown in Table 2.
Table 2. Fuzzy Rules in Modified SFLA with cognitive behavior (Fuzzy-MSFLA)

<table>
<thead>
<tr>
<th>If</th>
<th>Then</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iternormalized is Low and</td>
<td>C1 is Low and C2 is High</td>
</tr>
<tr>
<td>Bestnormalized is High</td>
<td></td>
</tr>
<tr>
<td>Iternormalized is Medium and</td>
<td>C1 is Medium and C2 is Medium</td>
</tr>
<tr>
<td>Bestnormalized is Medium</td>
<td></td>
</tr>
<tr>
<td>Iternormalized is High and</td>
<td>C1 is High and C2 is Low</td>
</tr>
<tr>
<td>Bestnormalized is Low</td>
<td></td>
</tr>
<tr>
<td>Iternormalized is High and</td>
<td>C1 is Low and C2 is High</td>
</tr>
<tr>
<td>Bestnormalized is High</td>
<td></td>
</tr>
<tr>
<td>Iternormalized is Low and</td>
<td>C1 is High and C2 is Low</td>
</tr>
<tr>
<td>Bestnormalized is Low</td>
<td></td>
</tr>
</tbody>
</table>

The best result for minimization of average power based on Fuzzy-MSFLA is shown in Figure 7 and Table 3.

The Least Average Power

Fig 7: Average Power = $1.19 \times 10^{-6}$ Watt, VDD=5v, Temperature= 46.5339175361742 °C

Table 3. Channel Widths and Temperature based on Fuzzy Modified SFLA with cognitive behavior (Fuzzy-MSFLA)

<table>
<thead>
<tr>
<th>W1=W3=W5 (µm)</th>
<th>W2=W4=W6 (µm)</th>
<th>Temperature (centigrade)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.297848989765905</td>
<td>0.322296653819349</td>
<td>29.1031851838616</td>
</tr>
</tbody>
</table>

3.5 Genetic Algorithm (GA)

In Genetic Algorithm the search agents are the chromosomes. The best chromosomes have the best objective function. Based on pseudo in Box 4 the program worked and the least average power which obtained by GA was shown in Figure 8.

Box 4. GA Pseudo-Code in this Article

- Define the problem of 3 Stage Ring Oscillator, CostFunction=Objective Function ("in Box2"), nVar=6, VarMin: 0.2 for channel widths and 1 for temperature, VarMax: 2 for channel widths and 50 for temperature, MaxIt=10, nPop=4 (Population Size) , pc=0.5 (Crossover Percentage), pm=0.5 (Mutation Percentage)
- Generate Initial Population, Evaluate, Sort, and place them in the Best Solutions
- Do While iteration number (it) < Maximum Iteration (MaxIt)
  - Calculate Selection Probabilities
  - Select Parents indices and apply crossover and evaluate offspring, in Crossover Operator
  - Select Parents indices and apply mutation and evaluate Mutant, in Mutation Operator
  - Create Merged Population, sort population
  - Update and store the best solution and Cost that ever found
- End Do

Plot Results

GA is firstly implemented in MATLAB software and the optimum layout for the 3 stage CMOS ring oscillator is obtained with power supply of Vss=5v and the temperature of 25°. The channel widths ("BestSol.Positions") for NMOSs and PMOSs are shown in Table 4 also the convergence of power dissipation is illustrated in Figure 8.

Table 4. Channel Widths ("BestSol.Positions")

<table>
<thead>
<tr>
<th>W1=W3=W5 (µm)</th>
<th>W2=W4=W6 (µm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.50897150624236</td>
<td>0.640468867234475</td>
</tr>
</tbody>
</table>

For implementation of algorithms the attention is focused to the fact that the channel widths of PMOSs have the same values and the channel widths of NMOSs have the identical values.

Fig 8: Average Power=$6.69 \times 10^{-5}$, VDD=5v, Temperature=25°C

The values of channel width that were shown in Table 4, remain unchanged in the input netlist file ("V3.sp") in HSPICE Software. The optimization is performed to find the best solution for the temperature that for which the average power is more reduced. The convergence is shown in Figure 9 with the optimum temperature.
- Define the problem of 3 Stage Ring Oscillator,
  CostFunction=Objective Function (“in Box2”), nVar=6,
  VarMin: 0.2 for channel widths and 1 for temperature,
  VarMax: 2 for channel widths and 50 for temperature,
  MaxIt=10, nPop=4 (Population Size), pc=0.5 (Crossover
  Percentage), pm=0.5 (Mutation Percentage)

- Generate Initial Population, Evaluate, Sort, and place them
  in the Best Solutions

- Do While iteration number (it) < Maximum Iteration
  (MaxIt)
  - Calculate Selection Probabilities
  - Select Parents indices and apply crossover and evaluate
  off springs, in Crossover Operator
  - Select Parents indices and apply mutation and evaluate
  Mutant, in Mutation Operator
  - Create Merged Population, sort population
  - Update the best solution and the worst solution that ever
  found and Store Best Cost and Worst Cost
  - Normalization of variables : itnormalized = it / MaxIt,
    and BestCostnormalized = [ WorstCost - BestCost(it) ] / 
    WorstCost
  - Read Fuzzy Inference System File and Fuzzy Rules
    (Fuzzy_GA_FIS.fis)
  - Save the results (BestCost)
  - End Do
  - Plot Results

The optimum control of mutation percentage (“pm”),
crossover percentage (“pc”), and the number of population
(“nPop”) is obtained by Fuzzy-GA rules that are illustrated in
Table 5. The best solutions are shown in Table 6.

<table>
<thead>
<tr>
<th>If</th>
<th>Then</th>
</tr>
</thead>
<tbody>
<tr>
<td>Itnormalized is High and</td>
<td>pm is High and pc is Low and nPop is High</td>
</tr>
<tr>
<td>BestCostnormalized is High</td>
<td></td>
</tr>
<tr>
<td>Itnormalized is High and</td>
<td>pm is Low and pc is High and nPop is Low</td>
</tr>
<tr>
<td>BestCostnormalized is Low</td>
<td></td>
</tr>
<tr>
<td>Itnormalized is Low and</td>
<td>pm is High and pc is Low and nPop is High</td>
</tr>
<tr>
<td>BestCostnormalized is High</td>
<td></td>
</tr>
<tr>
<td>Itnormalized is Medium and</td>
<td>All pm, pc, and nPop are Medium</td>
</tr>
<tr>
<td>BestCostnormalized is Medium</td>
<td></td>
</tr>
</tbody>
</table>

Table 6. Channel Widths (“BestSol.Positions”)

<table>
<thead>
<tr>
<th>W1=W3=W5 (µm)</th>
<th>W2=W4=W6 (µm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.486219988184535</td>
<td>1.34202178834180</td>
</tr>
</tbody>
</table>

The values of channel widths that were shown in Table 6
remain unchanged as the input netlist file (“V3.sp”) in
HSPICE and the parameter of power supply is changed. The
optimization is performed to find the best solution for the

3.6 Fuzzy-Genetic Algorithm (Fuzzy-GA)
The best average power is 3.14*10^-5 Watt in Figure 11.

- Fig 9: Average Power = 1.76*10^-5, VDD=5v,
  Temperature= 29.8766397972101°C

- Fig 10: The Fuzzy Inference System in Fuzzy-GA

- Fig 11: Average Power=3.14*10^-5 Watt, VDD=5v,
  Temperature=25°C based on Fuzzy-GA

The pseudo code of Fuzzy-GA is shown in Box 5.

Box 5. Fuzzy-GA Pseudo-Code which is used in Article
temperature and the best result is shown in Figure 12.

![Graph showing average power comparison](image)

**Fig 12:** Average Power=1.17*10^-5, VDD=5v, Temperature= 27.907561137555°C

4. CONCLUSIONS

By implementation of meta heuristic algorithms such as Modified SFLA with cognitive behavior (MSFLA), Fuzzy-MSFLA, GA, and Fuzzy-GA the least average power is compared and it has been obtained for the specific values of the parameters such as temperature, NMOS channel widths, and PMOS channel widths. These parameters have effect on the average power consumption in 3 stage CMOS ring oscillator. In this article the least average power of 1.19 µW is obtained based on Fuzzy-MSFLA in technology L=0.18µm and VDD=5v. The comparisons are illustrated in Figure 13 and Table 7.

![Graph showing average power comparison](image)

**Fig 13:** The Graphical Comparison

Also the Fuzzy-MSFLA and Fuzzy-GA showed the better results since the fuzzy rules increase the performance of the algorithm. The Fuzzy-MSFLA and Fuzzy-GA showed that the lower average power is resulted in lower temperature.

**Table 7. The Comparative Results for Average Power in 3 Stage Ring Oscillator**

<table>
<thead>
<tr>
<th>Heuristic Algorithms</th>
<th>The Least Average Power, (P_{avg})</th>
<th>Temperature (Centigrade)</th>
<th>W_{NMOS} (µm)</th>
<th>W_{PMOS} (µm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fuzzy-MSFLA</td>
<td>1.19 µW</td>
<td>29</td>
<td>0.32</td>
<td>0.29</td>
</tr>
<tr>
<td>MSFLA</td>
<td>2.4 µW</td>
<td>46</td>
<td>0.63</td>
<td>1.66</td>
</tr>
<tr>
<td>GA</td>
<td>17.6 µW</td>
<td>29</td>
<td>0.64</td>
<td>1.5</td>
</tr>
<tr>
<td>Fuzzy-GA</td>
<td>11.7 µW</td>
<td>27</td>
<td>1.34</td>
<td>0.48</td>
</tr>
</tbody>
</table>

5. ACKNOWLEDGMENTS

Dr. Khaliji provided me with SFLA code in MATLAB.

6. REFERENCES


