# Input Vector Control for Post-Silicon Leakage Current Minimization in the Presence of Manufacturing Variability

Yousra Alkabani[1], Tammara Massey[2], Farinaz Koushanfar[1,3], Miodrag Potkonjak[2]
[1]Computer Science and [3] Electrical & Computer Engineering Dept(s)., Rice University
[2]Computer Science Dept., University of California, Los Angeles

## ABSTRACT

We present the first approach for post-silicon leakage power reduction through *input vector control (IVC)* that takes into account the impact of the *manufacturing variability (MV)*. Because of the MV, the integrated circuits (ICs) implementing one design require different input vectors to achieve their lowest leakage states. We address two major challenges. The first is the extraction of the gate-level characteristics of an IC by measuring only the overall leakage power for different inputs. The second problem is the rapid generation of input vectors that result in a low leakage for a large number of unique ICs that implement a given design, but are different in the post-manufacturing phase. Experimental results on a large set of benchmark instances demonstrate the efficiency of the proposed methods. For example, the leakage power consumption could be reduced in average by more than 10.4%, when compared to the previously published IVC techniques that did not consider MV.

## Categories and Subject Descriptors

B.6.3 [**Logic Design**]: Design Aids; B.7.2 [**Integrated Circuits**]: Design Aids

## General Terms

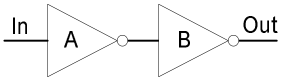
Design, Algorithm

## Keywords

Input Vector Control, Low Power, Manufacturing Variability

## 1. INTRODUCTION

There is a wide consensus that the intrinsic IC MV fundamentally alters synthesis and analysis procedures, and must be taken into account [13]. For example, the common model for approximating the leakage power is a lognormal [6], that has an exponential growth for small variations in gate oxide thickness and effective gate length. Thus, leakage power variations could increase by large factors in presence of MV. However, to the best of our knowledge, no existing solution is available for power optimization while considering the post-silicon MV. Probably the best way to clarify the importance of considering the MV impact is to use a small illustrative example. Figure 1 shows a very small design that consists of two inverters A and B where the output of inverter A is the input to inverter B. The second row of the Table in Figure 1 shows leakage power for an inverter of nominal size in 0.18um model from MOSIS that is simulated using the leakage current in Cadence Spectre [15]. The third row shows the ten times increased leakage power as the consequence of the MV.



| Input | 0 | 1 |
|---|---|---|
| Leakage | 100 nA | 227 nA |
| 10x leakage | 1000 nA | 2270 nA |

**Figure 1: Series inverters and their leakage currents.**

If A has the nominal leakage current, and B is ten times larger, the primary input value 1 must be used so that overall leakage current is 1227nA. On other hand, if B has the nominal leakage current, and A is ten times larger, one should use the value 0 as the primary input so that the overall leakage current is 2370nA. The incorrect decision results in 93% larger leakage current. Thus, it is important to consider MV to find the best inputs for leakage power minimization of each IC.

Post-silicon leakage power optimization using IVC poses two major and conceptually new synthesis challenges. The first is the extraction of the leakage power for each gate in an IC using only the potentially noisy overall leakage power measurements. For this task, we have developed an accurate and polynomial time generic linear programming solution [3].

The second problem is the rapid creation of power efficient IVC for a large number of somewhat similar, but unique ICs due to MV. We address this challenge using coordinated application of statistical clustering and iterative improvement. The importance of both problems permeates beyond post-silicon leakage power minimization because many synthesis and analysis tasks that consider the MV impact have to conduct gate level characterization and post-processing CAD optimizations. For example, supply and threshold assignment can be addressed using the presented procedures.

## 2. RELATED WORK

In the current and pending CMOS technologies, the existence of MV in device parameters results in high variability in the delay and power consumption of the VLSI circuits [13]. A number of research efforts have focused on the direct and invasive measurement and extraction of the variable parameters [8]. Assuming that the measurement results are available, several techniques for representing the spatial variations as well as higher level stochastic and probabilistic models have been studied [13]. MV can also be used for system security [2, 10].

It was shown that under certain assumptions, the delays of all paths in a circuit can be expressed as a linear combination of the

*DAC 2008*, June 8–13, 2008, Anaheim, California, USA

delays of a small subset of paths [12]. However, the focus of such methods has been on testing the delay faults and upper bounds on the critical path (CP) delays. To the best of our knowledge, our research is the first that extracts the device parameters by non-invasive methods using the power measurements, and by only applying a specific set of input vectors and solving a large optimization. The technique considers the impact of measurement error and MV that was not included in prior testing literature. Unlike the previous work where the goal of extraction was to find a global circuit-level model or testing, the extracted characteristics are individually used for the subsequent post-silicon leakage power optimization.

Scaling has a profound impact on the static power consumption of the circuits [4, 11]. To reduce the dynamic power, supply voltages are aggressively scaled down with each process generation. To sustain performance, the threshold voltages ($V_t$) are scaled, producing a significant subthreshold leakage. Other variable factors such as reverse junction bias also contribute to the leakage.

While initially, the emphasis was on switching power reduction [5], recently several device, circuit, and system level techniques have been proposed [1, 7, 13, 15]. Input vector control (IVC) is an effective way to minimize the leakage, because in the circuit's sleep state, the leakage current strongly depends on the input vector combination [1, 5, 7, 15]. Since IVC is an NP-complete problem, both exact and heuristic methods have been suggested [1, 15]. Also, simultaneous IVC and gate replacement were proposed [7, 15]. This paper is the first to perform customized post-silicon IVC for each chip. Gate replacement is not a viable option in this phase. We introduce a method for IVC, but more importantly, we introduce techniques for clustering the inputs for the different ICs.

# 3. PRELIMINARIES

**Variability models.** Process variations are generally divided into two categories: (i) inter-die variations (denoted by $\delta_{inter}$), that are die-to-die fluctuations, and; (ii) intra-die variations denoted by $\delta_{intra}$) that are the variability present inside one chip. The assumption is that inter-die variations similarly affect all the gates on one chip, while the intra-die variations may differently influence various devices. What complicates the analysis and modeling of intra-die variations is that there is a significant spatial correlation among the close-by gates that cannot be ignored. We use the equations from [6, 13] for intra-die variations, they model a parameter $p$ located at $(x, y)$ as, $p$=$\bar{p}$+$\delta_x x$+$\delta_y y$+$\epsilon$; $\bar{p}$ is the nominal value of the parameter at the $(0, 0)$ die location; $\delta_x x$ and $\delta_y y$ are the gradients of the spatial variations of the parameter in $x$ and $y$ directions; and $\epsilon$ is the random intra-chip variation component. The multivariate normal distribution (MVN) is used for modeling the vector of all random components across the chip and the intra-chip correlations among them. Furthermore, the grid model that partitions the space into grids was used, where devices within the same grid are highly correlated and devices in further grids are correlated proportional to their distances.

**Leakage power model.** The leakage current is a function of the process variations. We use the recent leakage model proposed by Chang and Sapatnekar [6]. Their model takes into account the subthreshold leakage ($I_{sub}$) and the gate tunneling leakage ($I_{gate}$).

**Measurement error model.** Power is measured from the external pins. Leakage current is read out in the steady state, where the dynamic power is not present. Such power measurement methods have been previously utilized, for example in power analysis of embedded software [14]. Environmental conditions, noise, packaging, thermal effects and many other complex phenomena affect the external current readings and cause measurement errors. The errors may vary from one design to the next, e.g., because of the differences in size, layout, and environment.

**Global flow.** The approach consists of two main phases: (i) gate level characterization for each integrated circuit (IC); and (ii) post-processing for identification of the input vector that minimizes the leakage power. Each phase has four steps.

The first step of the gate characterization phase is creation of simulation models for gate-level leakage characteristics for each IC that takes into account spatial, inter-, and intra-chip correlations. In an industrial set-ups, where the actual ICs are manufactured, this step is bypassed. In the second step, we simulate the leakage power measurements for the specified gate variations on one chip while superimposing errors. Again, in industrial set-ups, the actual measurements are conducted. If the measurements are conducted at higher temperatures, the required accuracy of the measurements is lowered because of the increased leakage. Once the measurements are available, we conduct the gate-level characterization of the pertinent ICs [3].

In the second phase, the first step is the generation of a relatively large number (denoted by $N$), of model instances corresponding to manufactured ICs of the pertinent design that differ because of the MV. The objective is to find the best IVCs that results in leakage power minimization. This step is accomplished using a novel large neighborhood iterative improvement (LNII) algorithm. Next, we identify similar solutions and select a small set of *representative input vectors* (denoted by $R$), in such a way that any non-selected IC has at least one IC with a representative input within its specified Hamming distance. Leveraging on the relatively small size of the representative inputs and their structure, this problem is optimally solved using integer linear programming (ILP). In the fourth step, we iteratively apply the LNII algorithm $R$ times, to find the IVC that minimizes the leakage for a newly characterized IC. Each time, we start from one of the $R$ representative inputs that greatly reduces the runtime. Lastly, the post-silicon IVC method is evaluated on a comprehensive set of benchmarks.

# 4. LEAKAGE POWER MINIMIZATION

In this section, we present new algorithms for leakage power minimization using MV-aware IVC. Because of space limitations, we omit the details of the gate-level characterization of ICs in this paper. For a full description, we refer the readers to [3]. The outcome of the gate-level characterization are the estimated scaling factor of the gates that are used for modeling the exact post-silicon leakage variations of an IC. The best IVC is then found for the estimated characteristics.

We first discuss the sources of difficulty in addressing the new problem and our overall algorithmic strategy. Next, we introduce very Large-scale Neighborhood search Iterative Improvement-based (LNII) algorithm for solving the optimization that finds the best IVC. We conclude the section by discussing our clustering approach that finds a small set of representative inputs. The representative set serves as the starting point for invoking the LNII algorithm for each new IC with unique gate characteristics along with our strategy for fast optimization of each unique instance.

## 4.1 Problem formulation, and methodology

Post-silicon leakage power minimization using IVC can be defined in the following way.

**Instance.** Given a combinational netlist with $G$ logic gates $g_i$, with no cycles and a set $S$ of scaling coefficients $s_i$ corresponding to each gate $g_i$, $(i = 1, \ldots, G)$. Also given a table $T$ of nominal values for leakage power consumption with entries $t_{k,j}$, where $k$ is the gate type (e.g., NAND, NOR) and $j$ is the ordered input combination to the gate. The coefficient $s_i$ indicates the scaling ratio between the leakage power of the gate $g_i$ to the nominal value of the leakage for the same gate type and the same input from $T$. The assumption is that the ratio is the same over different inputs. Note that, If the ratio is not constant over all gate inputs, one can easily generalize the approach, by introducting a new two-dimensional scaling factor $s'_{i,j}$ that characterizes the scale of the gate $g_i$ for the input $j$, where $j$ belongs to the set of the ordered inputs.

**Objective.** Generate the logic values assigned to the primary inputs of the circuit (i.e., IVC) so that the sum of the leakage powers for all of the gates is minimized.

**Complexity.** Majority of power optimization and synthesis problems are difficult because of their NP-complete structure [15]. Post-silicon leakage power minimization using MV-aware IVC has an additional source of difficulty: one has to solve a large number of instances. It is common that for each design, millions of ICs are manufactured. To address these challenges, we decided to follow the paradigm of separation of concerns. We also emphasize the software and results reusability paradigms. At an intuitive level, one would expect that a significant number of ICs with similar gate characteristics have identical or similar best input vectors.

Following the separation of concerns paradigm, we divided the overall approach is three steps as shown in Pseudocode 1.

**PSEUDO 1: Post-silicon leakage power minimization using MV-aware IVC algorithm**

```
1. Find the best solutions for $N$ ICs;
2. Cluster the best solutions and identify representative inputs;
3. Use representatives to find efficient inputs for the new ICs;
```

## 4.2 Generation of initial IC-specific solutions

The first step is generation of initial individual solutions for each of the $N$ ICs. This step has three goals: (i) creation of the initial starting point for the consequent optimization of a large number of ICs; (ii) learning the solution space in terms of its topology and how long the LNII algorithm should be executed; (iii) comparison of the IVC performance with the IVS resulting from the restricted search-based LNII, for the new ICs.

At the intuitive level, the LNII algorithm starts by assigning a random input vector as the current solution. LNII then iteratively looks for switching the value of an input, so that the leakage power is reduced. As soon as it finds such an input, it updates the current solution accordingly. The procedure is iteratively repeated until no alternation of a single inputs improves the current solution. If there is no such solution, the LNII algorithm enlarges the search space by looking for an improvement where two inputs are simultaneously altered. Any time an improvement is found, the scope is rested to an alternation of a single input. Although, in principle, one can keep enlarging the scope, the run time constraints usually limit the enlargement size to a few inputs. The LNII algorithm is summarized using the following pseudocode.

**PSEUDO-2: LNII algorithm**

```
Generate a random initial solution;
while (stopping criteria-overall == NO) {
    scope = 1;
    while (stopping criteria-scope == NO) {
        search for better solution within the scope;
        if (no-better-solution is found);
        scope++;
    }
}
```

## 4.3 Clustering

The goal of the clustering step is to identify a small number of representative input vectors that have the property that every other IC has a low leakage power input vector solution that is only a short Hamming distance from at least one representative input. The Hamming distance between a pair of inputs is calculated as the number of binary inputs that have to be altered to produce one of the solutions from another. Although numerous statistical clustering techniques are readily available [9], we decided to create a new ILP-based technique to leverage the relatively small number of instances and more importantly, the relatively sparse structures of the constraints.

We now formulate the clustering problem as an instance of the integer linear program (ILP).

**Given:** A set of $N$ input vectors $\mathcal{I}$ with elements $I_n$, $n = 1, \ldots, N$; a number $K$; a Hamming distance matrix $H_{\{N \times N\}}$ with elements $h_{nm}$, s.t.

$$h_{nm} = \begin{cases} 1, & \text{If Hamming}(I_m,I_n) \leq K \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

**Variables:** a vector $X_{\{N\}}$ with elements $x_n$ s.t.

$$x_n = \begin{cases} 1, & \text{If input vector } I_n \text{ is in the representative set} \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

**Objective Function:** The objective function is to minimize the number of representative inputs in the set, i.e., for $n = 1, ..., N$, $\min \sum_n I_n$.

**Constraints:** The set of constraints ensures that every input vector is either in the representative set, or it has a Hamming distance smaller than $K$ to an input vector in the representative set. For $n = 1, ..., N$, $I_n + \sum_m H_{mn} I_m \geq 1$.

## 4.4 Generation of final IC-specific solutions

The final step is generation of an input vector for a new IC once when the representative set is available. For this task, we use LNII that starts from the input vectors in the representative set. One can envision many more sophisticated versions where one uses the gate characteristics to identify the best starting point, but our extensive experimentations with several such strategies indicate that those methods significantly increase the run time, with no performance benefits in terms of leakage power optimization.

# 5. EXPERIMENTAL RESULTS

Our methods were tested on circuits from the MCNC'91 benchmarks. The synthesis was done with SIS and the LPs were solved using CPLEX. The variation model was described in Section 3. The technology parameters are the ones used in [15], who reported the leakage values for the 0.18$\mu$m in their table: we emphasize that the only important matter is the ratio between the leakage powers of the library cells and not the absolute values. The spatial correlation is such that the correlation value decreases with the distance between the grids, and there are 20 grids. On each chip, we randomly selected 5 center grids where the variations are the highest and the variations of the other grids are computed by correlations to those centers. The 3$\sigma$ of the parameter variations of $T_{ox}$ and $L$ was set to 25% (corresponding to the 90nm technology). The inter-die is about 75% of the variations, while the intra-die is about 25%.

The experimental results for nondestructive gate-level characterization, creation of initial IC-specific solutions, and clustering are reported in [3]. Once we find a representative set of input vectors from the initial $N$ chips, we apply the input set on new ICs. Table 1 shows the resulting leakage current for a batch of 100 new ICs. In

Table 1, the first column is the benchmark, the second column is the resulting leakage current by using the MV-aware method. The third, fifth, and seventh columns show the resulting leakage from the DP, r10K and r1K IVC methods respectively. The MV-aware algorithm uses the initial solutions from the representative set, along with only one round of 1 bit flipping in the LNII algorithm for IVC. The representative inputs are from the Hamming distance of 5. An average improvement of 12.3%, 24%, and 21% in leakage power is obtained over the DP, r10K, and r1K respectively. The results show a few percentage improvement over those power saving for initial solution, while the runtime is significantly reduced [3].

| **circuit** | **MV (uA)** | **DP (uA)** | % | **r10K (uA)** | % | **r1K (uA)** | % |
|---|---|---|---|---|---|---|---|
| **9symml** | 56.64 | 62.57 | 10.46 | 64.99 | 14.74 | 64.99 | 14.74 |
| **lal** | 25.39 | 27.70 | 9.11 | 32.20 | 26.85 | 32.20 | 26.85 |
| **c8** | 45.32 | 49.27 | 8.72 | 57.13 | 26.08 | 50.68 | 11.83 |
| **term1** | 94.34 | 103.30 | 9.50 | 100.81 | 6.86 | 101.65 | 7.74 |
| **too_lrg** | 189.25 | 206.50 | 9.11 | 220.94 | 16.74 | 229.37 | 21.20 |
| **x1** | 73.02 | 81.73 | 11.93 | 95.78 | 31.17 | 82.44 | 12.90 |
| **i8** | 613.93 | 836.52 | 36.26 | 752.91 | 22.64 | 684.88 | 10.96 |
| **x3** | 172.63 | 191.84 | 11.13 | 197.17 | 14.22 | 199.47 | 15.55 |
| **pair** | 428.24 | 461.81 | 7.84 | 487.64 | 13.87 | 471.35 | 10.07 |
| **i7** | 99.92 | 121.70 | 21.80 | 175.33 | 75.48 | 177.63 | 77.78 |
| **max** | 428.24 | 461.81 | 23.58 | 487.64 | 75.48 | 471.35 | 77.78 |
| **min** | 25.39 | 27.70 | 7.84 | 32.20 | 6.86 | 32.20 | 7.74 |
| **avrg** | 127.83 | 142.20 | 12.32 | 154.01 | 24.16 | 151.42 | 21.03 |

**Table 1: MV aware solution using representative inputs vs the non MV-aware methods**

Table 2 outlines the average root mean square error while simulating 100 different chips considering 1%, 2%, 5%, and 10% measurement errors. An average error of 0.2%, 0.38%, 0.94%, and 1.91% in the computed leakage are obtained in the case of 1%, 2%, 5%, and 10% errors respectively. The results are very significant in light of the gate-level extraction error results we obtained in the non-invasive characterization of ICs. Our results show that even in presence of large measurement errors, the gate level characterization is very accurate [3].

## 6. CONCLUSION

We address the new problem of post-silicon leakage power minimization by selecting an input vector for each IC that suits the IC's specific characteristics, because of its specific manufacturing variability. We solve two conceptually new technical problems. The first problem of the extraction of gate-level characteristics is solved optimally on the practical instances using a new linear programming formulation. The second problem is rapid generation of high quality solutions for large number of unique ICs. Experimental results demonstrate improvements by an average factor of more than 10% can be achieved over the previously published techniques that generate input vectors for leakage power optimization without considering manufacturing variability.

## 7. ACKNOWLEDGEMENT

This work is partly supported by NSF, under the grant number 0716674. The work by Rice authors is also partially supported by the DARPA/MTO Young Faculty Award (YFA), grant number W911NF-07-1-0198.

| **B.M.** | **1%** | **2%** | **5%** | **10%** |
|---|---|---|---|---|
| **9symml** | 0.28 | 0.57 | 1.72 | 2.91 |
| **lal** | 0.3 | 0.64 | 1.59 | 3.37 |
| **c8** | 0.26 | 0.51 | 1.22 | 2.28 |
| **term1** | 0.21 | 0.35 | 0.9 | 2.18 |
| **too_large** | 0.18 | 0.31 | 0.72 | 1.63 |
| **x1** | 0.38 | 0.49 | 1.06 | 1.89 |
| **i8** | 0.19 | 0.36 | 0.97 | 1.84 |
| **x3** | 0.13 | 0.26 | 0.59 | 1.39 |
| **pair** | 0.1 | 0.2 | 0.42 | 0.88 |
| **i7** | 0.15 | 0.27 | 0.67 | 1.67 |
| **Average** | 0.20% | 0.38% | 0.94% | 1.91% |

**Table 2: Root mean square error of final leakage reduction for circuits with 1%,2%,5%, and 10% leakage estimation error.**

## 8. REFERENCES

[1] A. Abdollahi, F. Fallah, and M. Pedram. Leakage current reduction in CMOS VLSI circuits by input vector control. *IEEE Trans. VLSI*, 12(2):140–154, 2004.

[2] Y. Alkabani and F. Koushanfar. Active hardware metering for intellectual property protection and security. In *USENIX Security*, pages 291–306, 2007.

[3] Y. Alkabani, T. Massey, F. Koushanfar, and M. Potkonjak. Input vector control for post-silicon leakage current minimization under manufacturing variations. Technical report, Rice University, Electrical and Computer Engineering Department, 2008.

[4] S. Borkar, T. Karnik, S. Narendra, J. Tschanz, A. Keshavarzi, and V. De. Parameter variations and impact on circuits and microarchitecture. In *DAC*, pages 338–342, 2003.

[5] A. Chandrakasan, M. Potkonjak, R. Mehra, J. Rabaey, and R. Brodersen. Optimizing power using transformations. *IEEE Trans. on CAD*, 14(1):12–31, 1995.

[6] H. Chang and S. Sapatnekar. Full-chip analysis of leakage power under process variations, including spatial correlations. In *DAC*, pages 523–528, 2005.

[7] L. Cheng, L. Deng, D. Chen, and M. Wong. A fast simultaneous input vector generation and gate replacement algorithm for leakage power reduction. In *DAC*, pages 117–120, 2006.

[8] P. Friedberg, Y. Cao, J. Cain, R. Wang, J. Rabaey, and C. Spanos. Modeling within-die spatial correlation effects for process-design co-optimization. In *ISQED*, pages 516–521, 2005.

[9] A. Jain, M. Murty, and P. Flynn. Data clustering: A survey. *ACM Computing Survey*, 31:264–323, 1999.

[10] F. Koushanfar and M. Potkonjak. CAD-based security, cryptography, and digital rights management. In *DAC*, pages 268–269, 2007.

[11] S. Mukhopadhyay, A. Raychowdhury, and K. Roy. Accurate estimation of total leakage in nanometer-scale bulk CMOS circuits based on device geometry and doping profile. *IEEE Trans. on CAD*, 24(3):363–381, 2005.

[12] M. Sharma and J. Patel. Finding a small set of longest testable paths that cover every gate. In *ITC*, pages 974–982, 2002.

[13] A. Srivastava, D. Sylvester, and D. Blaauw. *Statistical Analysis and Optimization for* VLSI*: Timing and Power*. Series on Integrated Circuits and Systems. Springer, 2005.

[14] V. Tiwari, S. Malik, and A. Wolfe. Power analysis of embedded software: a first step towards softwarepower minimization. 2(4):437–455, 1994.

[15] L. Yuan and G. Qu. A combined gate replacement and input vector control approach for leakage current reduction. *IEEE Trans. on VLSI*, 14(2):173–182, 2006.