SVD-Based Ghost Circuitry Detection

Michael Nelson[1], Ani Nahapetian[1], Farinaz Koushanfar[2], and Miodrag Potkonjak[1]

[1] Computer Science Department, UCLA, Los Angeles, CA 90095, USA
{ani,miodrag}@cs.ucla.edu
[2] Computer Science Department, Electrical and Computer Engineering Department, Rice University, Houston, 77005 TX
farinaz@cs.rice.edu

Abstract. Ghost circuitry (GC) insertion is the malicious addition of hardware in the specification and/or implementation of an IC by an attacker intending to change circuit functionality. There are numerous GC insertion sources, including untrusted foundries, synthesis tools and libraries, testing and verification tools, and configuration scripts. Moreover, GC attacks can greatly compromise the security and privacy of hardware users, either directly or through interaction with pertinent systems, application software, or with data. GC detection is a particularly difficult task in modern and pending deep submicron technologies due to intrinsic manufacturing variability. Here, we provide algebraic and statistical approaches for the detection of ghost circuitry. A singular value decomposition (SVD)-based technique for gate characteristic recovery is applied to solve a system of equations created using fast and non-destructive measurements of leakage power and/or delay. This is then combined with statistical constraint manipulation techniques to detect embedded ghost circuitry. The effectiveness of the approach is demonstrated on the ISCAS 85 benchmarks.

Keywords: Hardware Trojan horses, gate characterization, singular value decomposition, manufacturing variability.

1 Introduction

Ghost circuitry (GC) insertion is an intentional hardware alteration of the design specification and IC implementation. The alterations only affect the circuit's functionality in a few specific circumstances and are hidden otherwise. GC is more difficult to detect than design bugs or manufacturing faults, since it is intentionally implanted to be unperceivable by the current debugging and testing methodologies and tools. The vast number of possibilities for inserting GC further complicates detection.

In a GC insertion attack, the adversary adds one or more gates such that the functionality of the design is altered. The gates can be added so that no timing path between primary inputs and flip-flops (FFs) and primary outputs and FFs is altered. However, leakage power is always altered. Even if the attacker gates the added circuitry, the gating requires an additional gate.

Our goal here is to detect the insertion of GC, specifically added gates, in the face of low controllability and observability of gates. However, the GC detection approach

S. Katzenbeisser and A.-R. Sadeghi (Eds.): IH 2009, LNCS 5806, pp. 221–234, 2009.

is generic enough that it can easily be retargeted to other circuit components, such as interconnect by considering more comprehensive timing and/or power models. The main technical obstacle to GC detection is manufacturing variability, which can have a significant impact on gate timing and power characteristics across ICs.

The basis for our approach is gate-level characterization using a set of non-destructive timing and/or power measurements. The measurements are treated as a set of linear equations and are processed using singular value decomposition (SVD) to fingerprint the circuit. The detection of additional ghost circuitry is carried out by imposing additional constraints on the linear equations in such a way that the results indicate whether circuitry was added. Essentially, we ask if the characterization of gates is significantly more consistent under the assumption of added circuitry.

Gate-level characterization (GLC) has emerged as a premier synthesis, analysis, watermarking, cryptography, and security task in current and even more pending deep submicron silicon technologies subject to manufacturing variability. Two major GLC ramifications are widely addressed post-silicon customization where the pertinent integrated circuit (IC) is differently operated as a function of its gate-level characteristics and design under uncertainty where the design is synthesized in such a way that the consequent impact of manufacturing variability is considered and compensated. Hardware and in particular gate-level and physical design watermarking is greatly impacted in at least two ways: through potential negative impact on the performance overhead and sharply increased detection difficulty. Recently, it has been shown that GLC can be used for hardware-based secret and public-key cryptography that eliminates physical and side channel attacks. Finally, manufacturing variability greatly complicates the defense against hardware security attacks such as gate resizing and addition of GC. In gate resizing, the attacker changes the size of one or more gates in such a way that under general or specific circumstances (e.g. a specific input vectors) the energy consumption is excessive or timing correctness is violated.

While all of the itemized tasks are of paramount importance, our primary goal is detection of gate resizing and GC hardware attacks. The emphasis is on GLC that directly solves gate resizing attack by calculating their power, timing, and other characteristics. GLC, augmented with statistical techniques, is also basis for GC detection.

In the remainder of the paper, we present some background on manufacturing variability, gate-level characterization, and ghost circuitry detection. Then we present our SVD-based approach for carrying out gate-level characterization and the related simulation results. Finally, we present the ghost circuitry detection approaches utilizing the gate-level characterization obtained from our procedure, which incorporates SVD and some post-processing of the results.

2 Related Work

Manufacturing variability is the result of intense CMOS technology scaling, which results in a high degree of variability across ICs from the same design and even from the same wafer, with regards to gate sizing, power consumption, and timing characteristics. A new generation of security techniques based on manufacturing variability (MV) has been developed [8][9][12].

Gate level characterization in the face of manufacturing variability has been examined before in other settings and with other techniques, but some important extensions and differences exist in our work. [18] and [19] use compressed sensing to determine gate level characteristics: power and timing, respectively. However, they make a simplifying assumption necessary for employing their technique; specifically that manufacturing variability is correlated across adjacent gates. In practice, the implantation of dopants is done individually which results in weak local correlation. Our techniques make no assumptions about the correlation of manufacturing variability across gates, and thus our approach is more robust and realistic. [17] uses convex programming to determine gate level characteristics. Its applicability is limited to butterfly networks, where a path exists from each input to each output. [15] and [16] utilize a linear programming approach, whereas, our work uses singular value decomposition (SVD), post-processing of results, and most importantly utilization of gate characterization for GC detection.

Manufacturing variability aware gate-level characterization can be used to optimize manufacturing yield, carry out remote enabling and disabling of ICs [25], determine higher quality IC for appropriate distribution [8], in addition to its applicability to ghost circuitry detection.

Kuhn [23] presents ghost circuitry detection strategies that utilize mask comparison. In our attack model we assume that the embedding of the ghost circuitry is carried with the aim of obfuscating the embedding as manufacturing variability. Additionally, we do not assume that we have access to the masks, as the foundry may be a source of the GC insertion.

3 Preliminaries

3.1 Manufacturing Variability Model

Manufacturing variations are due to the intense industrial CMOS feature scaling. With scaling of feature sizes, the physical limits of the devices are reached and uncertainty in the device size increases [5]. Variations in transistor feature sizes and thus, in gate characteristics, e.g., delay or power, are inevitable. In present and pending technologies, the variation is large compared to the device dimensions. As a result, VLSI circuits exhibit a high variability in both delay and power consumption. In this work, manufacturing variation in gates is modeled as a multiplicative scaling factor.

3.2 Measurement Model

To carry out the ghost circuitry detection using gate level characterization, a limited number of nondestructive measurements are taken. After manufacturing, the original design is available, and it is possible to provide input vectors to the input pins of the manufactured chip and obtain the respective outputs from the output pins. Additionally, it is possible to measure the IC's leakage power consumption. To measure the individual path delays, an input vector is provided to the IC, and then a single input bit is flipped. With knowledge of the IC design, the delay incurred between the input vector application and the output vector change can be used to calculate the delay value of the path.

We assume that it is possible to have measurement error in every single measurement taken, i.e. every single application of an input vector and measurement of it power and delay characteristics at the output. This error, however, has shown to be small on the order of 1% as cited in selected previous literature [24][26]. We model this value in our linear equations and have examined a uniform model for the measurement error in our work.

Generally, the model may be affected by aging and temperature. However, as these circuits are tested shortly after fabrication, aging is not a factor. Temperature is a factor, but normalization with repeated measurements can handle this. Additionally, control of environmental factors such as room temperature and working from cold boot can eliminate the variation that may be witnessed in the circuit behavior due to temperature variation across measurements. The measurements are made in a control environment after fabrication, so it is very easy to eliminate factors such as humidity, dust, presence of electromagnetic radiation, etc.

3.3 Threat Model

Since semiconductor manufacturing demands a large capital investment, the role of contract foundries has dramatically grown, increasing exposure to theft of masks, attacks by insertion of malicious circuitry, and unauthorized excess fabrication [1]. The development of hardware security techniques is difficult due to reasons that include limited controllability and observability (50,000+ gates for each I/O pin in modern designs) [7], large size and complexity (the newest Intel processor has 2.06B transistors), variety of components (e.g., clock, clock distribution interconnect, and finite state machine), unavoidable design bugs, possibility of attacks by non-physically connected circuitry (e.g., using crosstalk and substrate noise), many potential attack sources (e.g. hardware IP providers, CAD tools, and foundries), potentially sophisticated and well-funded attackers (foundries and foreign governments), and manufacturing variability that makes each IC coming from the same design unique [5][11].

In this paper, we assume the attackers can embedded ghost circuitry, even as little as a single gate. This insertion can occur at various stages of the IC manufacturing process, including through CAD tools, through the use of outside IP, and at the foundry during the fabrication process. The attacker can carry out many different types of hardware attacks, including gate resizing, removing gates, and allowing crosstalk. However, in this paper, we consider ghost circuitry attacks that obtain information from the IC, implying that at least one gate is inserted.

4 Singular Value Decomposition for Gate-Level Characterization

4.1 Problem Formulation

Manufacturing variation in power and delay behavior of gates is modeled by associating each gate with a scaling factor, α, which multiplies both delay and leakage current. Measurements of total leakage power and path delay for various circuit inputs gives rise to linear equations with the scaling factors as the unknowns. Each set of measurements produces a linear system $Ga = m+e$ where

- a is the vector of scaling factors, also referred to as the α-values, and related to gate size
- $m+e$ is a vector of measured values
- m would be the measured value if there is no measurement error
- e is the measurement error associated with each measured value
- G is derived from the expected power and/or delay characteristics of the gates.

For N_g number of gates in the circuit and N_m number of measurements, G is $N_m x\, N_g$, a is $N_g x\, 1$, and m is $N_m x\, 1$.

More abstractly, one can imagine the circuit's gate characteristics split into two components represented by G and a. G represents the characteristics of gate classes, i.e. 2-input NANDs power and delay characteristics for a given input vector, and it is inherent the circuit design. This information is readily available and in our experiments we have used the values provided by [21] for delay and [20] for leakage power.

The vector a, which is a vector of α-values for all the gates in the circuit, represents the unknowns in the equation. In other words, a is the fingerprint for the circuit just as the α-value is the fingerprint for the individual gate. Due to manufacturing variability, gate sizes are not exactly matched to the design specifications. The size of each gate in the circuit of each fabricated IC can have a variety of values. All circuits accordingly will have a large variety of sizes for most or all of their gates, and hence the extremely large combinations of possibility for a results in a unique fingerprint for each circuit.

Splitting each manufactured circuit into an invariant and into a variant component results in, G, which is universal across all circuits of the same design for the same set of input vectors, and a, which represents the unique characteristics of the fabricated circuit.

A large set of measurements are taken for the total circuit. As we can only access the input and output pins of the circuit, all the measurements made, represented by $m+e$, are made from a global circuit or path level and not at the individual gate level. Obviously, if we were able to measure these values at the gate level, we would easily be able to solve for each gate's α-value.

We do consider error in the formulation, as measurement error is possible when measuring total leakage power for the circuit and total delay along a path of the circuit from input to output pin. This is represented by e, which is the error that may be introduced in the measurement for each input vector or pair of input vectors.

A singular value decomposition $G = U\Sigma V^T$ is used in the following way. G^+, the pseudo-inverse of G, gives a least-squares solution to the system, a', an approximation of the scaling factors given the possibility of measurement errors being introduced. The procedure for fingerprinting circuits, i.e. determining the α-values as accurately as possible is the following: (1) Choose a set of circuit inputs. (2) Compute G and G^+ . (3) Perform measurements on a circuit to produce $m+e$. (4) Compute the fingerprint $a' = G^+(m+e)$.

In this formulation, a' represents the fingerprint that we deciphered from the SVD. It does not necessarily match a, due to the measurement error and also due to gate correlations that hinder gate-level characterization.

In the next subsections, we provide not only the power and delay models, but also a complete example that we solve to demonstrate more clearly procedure followed to accomplish gate-level characterization.

4.2 Power Model

The total leakage power consumed by a circuit is the sum of the leakage power of its gates [20]. For a particular circuit input i and a measurement ML_i of total leakage power with input i, we have the equation, $\sum_g GL_{gi} \cdot \alpha_g = ML_i$ where GL_{gi} is the expected leakage current for gate g when the global input is i. Each equation contributes a row to G and an entry to m in the overall system, $Ga = m+e$.

Table 2 shows a matrix G computed from the example circuit in Figure 1, using input-dependent leakage values from Yuan and Qu [20], shown in Table 1.

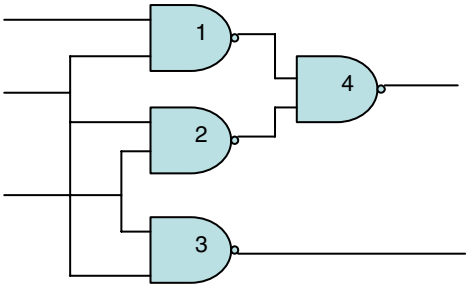

Fig. 1. Example circuit with NAND gates used to demonstrate SVD-based GLC

Table 1. Input-dependent leakage current for a 2-input NAND gate

00	37.84 nA
01	95.17nA
10	100.3 nA
11	454.5 nA

Table 2. Power matrix for example circuit given in Figure 1

Input	Gate 1	Gate 2	Gate 3	Gate 4
000	37.84	37.84	37.84	454.5
001	100.3	37.84	37.84	454.5
010	95.17	100.3	100.3	454.5
011	454.5	100.3	100.3	95.17
100	37.84	95.17	95.17	454.5
101	100.3	95.17	95.17	454.5
110	95.17	454.5	454.5	100.3
111	454.5	454.5	454.5	37.84

4.3 Delay Model

Total delay along paths through the circuit is measured by changing one input to the circuit and waiting for the change to propagate to the outputs. For a pair of circuit inputs *i* and *j*, the "before" and "after" inputs, zero or more output measurements will be made. For any measurement MD_p whose output is connected to the changed input by a unique path p, we have the equation $\sum_g GD_{gij} \cdot P_{gp} \cdot \alpha_g = MD_p$ where GD_{gij} is the expected gate delay for gate g when the global input transitions from *i* to *j*, and P_{gp} is an indicator function (0 or 1) that tells whether gate *g* is on path *p*. Each equation contributes a row to *G* and an entry to *m+e* in the overall system, Ga = m+e. Some output measurements will not have unique paths, and in this case, we do not know which path had the shortest delay, even though we do have a lower bound on the delay for these paths.

One can compute the matrix G for the example circuit in Figure 2.1, using delay values from Ercegovac, et. al. [21], shown in Table 3. There do exist newer models for delay characteristics of gates. Our model is independent of these values. In fact, the work can be easily be extended for new and changing model of gate characteristics in terms of delay, leakage power, or other gate characteristics.

Table 3. Delay for a 2-input NAND gate where L is the fanout

0 →1	0.05+0.038L ns
1→0	0.08+0.027L ns

4.4 Computing α-Values

The equations generated from leakage and/or delay measurements are combined into the system, *Ga = m+e*. Again recall that N_g is the number of gates in the circuit, and N_m is the number of measurements. A singular value decomposition of *G* has the form $U\Sigma V^T$, where *V* is *Ng x Ng* and orthogonal, *U* is N_m *x* N_m and orthogonal, and Σ is *Nm x Ng* and diagonal; the entries on its diagonal are the singular values. The rank of *G* is equal to the number of nonzero singular values; by convention, we assume that the nonzero singular values are in the leftmost columns of Σ.

The pseudoinverse of *G* is $G^+ = V \Sigma^+ U^T$, where Σ^+ is derived from Σ by replacing each nonzero singular value σ with its inverse $1/\sigma$. Performing the multiplication $G^+(m+e)$ gives our fingerprint *a'*, the vector in the column space of *G* for which the norm of *Ga' - (m+e)* is minimized. The fingerprint vector *a'* has the following properties: (1) If *G* has rank N_g, then *a'* is an approximation of *a*. (2) If *G* has rank < N_g, then *a'* is an approximation of the portion of *a* which is not annihilated by *G*.

Table 4 shows an example *a* for the circuit in Figure 1. The measurement vector *m* computed from this *a* and the power matrix in Table 1 and the resulting fingerprint vector *a'*. Because this matrix is not full rank, some α-values are inaccurate, even though we did not add any measurement error.

As shown in Figure 1, gates 2 and 3 are both 2-input NANDs and they both have the same input vector in all possible measurements, as they both have by design the

same input vectors. As a result, it is not possible to separately characterize gates 2 and 3 since their G matrix entries will be same for all inputs vectors. The best that is achievable is to characterize the sum of their α-values, which in this case is 2.050, and it has been properly characterized. This demonstrates how even without measurement error it is possible to not properly fingerprint a circuit in some cases.

Table 4. α-value fingerprint obtained for example circuit

a	m	a'
1.015	537.4	1.015
1.103	600.7	1.025
0.9473	723.6	1.025
0.9271	755.0	0.9271
	654.9	
	718.2	
	1121	
	1428	

The task of determining the inputs vectors applied for which measurements are taken is not as straightforward as it seems. First, due to the prohibitive size of the input vector domain, an exhaustive search can only applied to the smallest of circuits. Secondly, certain input vectors will maximize the solution quality, while others may be redundant or even obfuscate the true value. For large circuits, a set of input vectors must be chosen that maximizes the rank of G. We have used the following heuristics in our work in this paper. (1) Start with an empty G (2) Choose a random input vector and compute its matrix row 3) If the row is independent of the existing rows of G, add it to G (increasing G's rank) (4) Repeat from step (2).

Since we do not know the maximum possible rank in advance, this process must be repeated until some arbitrary stopping condition is met, such as some number of failed choices in a row. For numerical robustness, N_m should be larger than $rank(G)$, and more random inputs can be added afterward to accomplish this.

5 Gate-Level Characterization (GLC)

Simulations were performed on the ISCAS 85 benchmark circuits [19]. Though the benchmarks are combinatorial circuits, the approach can be extended to sequential circuits simply by scanning the flip-flops as we are doing with the outputs and inputs. Custom software was written in C++ to construct input sets and compute G.

Table 5 shows the number of solvable α-values for each input set. Not all values can be recovered due to gate correlations. Additionally, some sets of α-values can be thought of as a single unit; those for which the sum of the set is solvable, but none of the differences between members are.

Table 6 illustrates the average accuracy of SVD-based GLC results, for five ISCAS85 benchmarks and different measurement modalities. We vary the average percentage of the measurement error for leakage power, timing, and both power and

Table 5. Number of scaling factors for gates recoverable using SVD (Table entries marked n/a were not computed due to their prohibitive size)

Circuits	# of Gates	Power	Delay	Both
c432	160	80	139	153
c499	202	154	136	202
c880	383	113	164	383
c1355	546	18	116	408
c1908	880	26	354	559
c2670	1193	138	287	594
c3540	1669	101	202	N/A
c5315	2307	429	1057	N/A

Table 6. SVD-based GLC for leakage power, delay, and both power and delay, for five ISCAS85 different benchmarks

Accuracy of Recovered Scaling Factor Values (α-Values)				
Measure Err	**0.1%**	**0.5%**	**1%**	**10%**
c432 Power	0.059	0.295	0.584	5.90
c432 Delay	0.035	0.175	0.351	3.48
c432 Both	0.011	0.054	0.107	1.07
c499 Power	0.372	1.84	3.69	0.362
c499 Delay	0.012	0.060	0.121	1.19
c499 Both	0.009	0.043	0.085	0.852
c880 Power	1.78	8.76	0.179	0.018
c880 Delay	0.016	0.078	0.156	1.57
c880 Both	0.069	0.350	0.689	6.95
c1355 Power	1.24	6.32	0.126	0.013
c1355 Delay	0.056	0.281	0.554	5.56
c1355 Both	0.049	0.244	0.483	4.83
c1908 Power	0.865	4.20	8.72	0.867
c1908 Delay	0.136	0.679	1.34	0.135
c1908 Both	2.78	0.141	0.282	0.028

timing. The results indicate that the quality of GLC is better than the measurement error, implying a successful characterization of the α-values of benchmark circuit gates.

The experiments presented in Table 6 were improved dramatically by averaging the results over several runs. Table 7 shows our results after averaging 100 and 1000 runs, for three different measurement errors. The table's values demonstrate that with post-processing the results can be improved in terms of reducing the average GLC error. In some cases, the post-processing has an effect as large as a factor of 12 and in some cases it drives the error down to 0.

In Figure 2, we represent the result of varying the gate-size range. The graph demonstrates the accuracy with which we are able to characterize the gates, in terms of the

average percentage of difference between the original value and the recovered gate size values. The x-axis represents the range of possible gate size values chosen randomly from the uniform range. This graph demonstrates that this variation does not affect the gate-level characterization. The small level of variation is due to the randomly chosen input vector values that actually have an impact on the gate-level characterization accuracy. The graph demonstrates that the level of manufacturing variability does not help or hurt our approach. Rather, the attacker requires the presence of manufacturing variability to help hide its ghost circuitry.

Table 7. GLC accuracy given post-processing of data by averaging of runs

Accuracy of Recovered Scaling Factor Values (c432, c499)			
	# of Averaged Runs		
Measurement Error	**1**	**100**	**1000**
.01%	**(%)**	**(%)**	**(%)**
Power	0.006, 0.037	0.001, 0.004	0.0002, 0.001
Delay	0.003, 0.001	0, 0	0, 0
Both	0.001, 0.001	0, 0	0, 0
.1%	**(%)**	**(%)**	**(%)**
Power	0.059, 0.371	0.006, 0.037	0.002, 0.013
Delay	0.035, 0.012	0.003, 0.001	0.001, 0
Both	0.011, 0.009	0.001, 0.001	0, 0
1%	**(%)**	**(%)**	**(%)**
Power	0.584, 3.69	0.059, 0.374	0.019, 0.016
Delay	0.351, 0.121	0.033, 0.012	0.011, 0.004
Both	0.107, 0.085	0.010, 0.009	0.003, 0.003

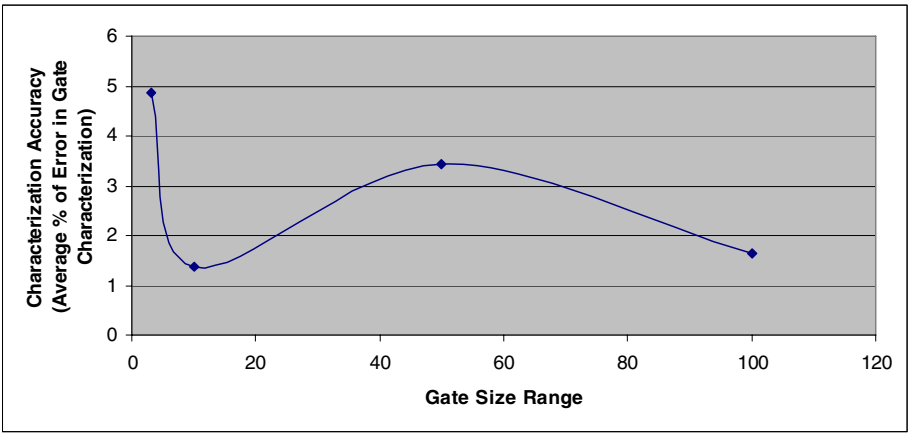

Fig. 2. For the c499 benchmark with 700 measurements the gate-level characterization accuracy for three different ranges of gates sizes, namely 0-3, 0-10, 0-50, and 0-100

6 Ghost Circuitry Detection

Our goal in this section is to address the detection of ghost circuitry using statistical techniques given the results of gate-level characterization. First, consider that there will be a possible shift in the scaling factors of the gates calculated by GLC when there is ghost circuitry present. The reason for this is that the contribution of the delay and the leakage power of the ghost circuitry will be attributed to the other gates in the circuit or the other gates along the paths where the GC is.

We analyze whether a systematic shift of all the scaling factors occurs when a group of ICs are analyzed. If ghost circuitry has been embedded, there is at least one gate that is not included in the linear equations for gate level characterization, and we can expect that a majority of gates will have scaling factors higher than the nominal factors as a result. In particular to conclude that ghost circuitry is present, we check whether the average α-values for all gates are above the average for other circuits and if it does not follow a Gaussian distribution that has been reported for the different silicon processes.

More formally, consider that we will try to solve for a assuming that we have $Ga = m+e$, but with ghost circuitry present we actually have $Ga = m+e+m_g$, where m_g is the vector representing the delay and/or the power contribution of the ghost circuitry. If ghost circuitry is present then α-value(s) of the ghost gate(s) will have an impact on the measurements obtained for the circuit in the case of power and for the applicable paths in the case of delay. However, the matrix G will not represent this ghost gate's contribution. Also, there will be no α-value accounted for in a. This will result in an increase in the average value calculated for the α-values of the other gates. The impact of the ghost circuit's power and delay will be distributed across one or more legitimate gates. As a result, if a shift is noted in the α-values, across different ICs of the same design or even as compared with the expected α-values, then we can use this as a predictor for determining the presence of ghost circuitry.

This technique places two significant assumptions: 1) The use of semiconductor processes does not induce unintentional bias, and 2) the measurement instruments do not have a positive systematic bias. The second assumption can also be eliminated if we add an additional factor to our linear equations notably a nonnegative variable to represent that bias. The advantage of this technique is that it is fast. On the other hand, it may not be applicable in all situations. An important challenge is finding a small number of gates in large circuits. As shown in the experiments below in the small benchmarks it is possible to notice the effects of adding a single ghost circuit, as SVD solution increases the α-value of other gates to compensate for the ghost circuit's delay and power side effects.

Table 8 presents the results of implementing this technique on the c17 and c432 benchmark of the ISCAS 85 benchmark suite. A single NAND gate was inserted randomly into certain circuits and nondestructive delay measurement were examined. The average α-values were compared between the two sets of circuits. As demonstrated there is a systematic positive measurement error, which is a strong indicator that ghost circuitry has been added.

Table 8. The average normalized α-value without the presence of ghost circuitry and the average normalized α-value with the presence of embedded ghost circuitry, along with the average increase in the α-values

Measurement Error	Average α-value without GC (c17, c432)	Average α-value with GC (c17, c432)	Average increase in α-value (c17, c432)
0%	2, 2.610	2.21, 2.629	10.5%, 0.72%
3%	1.94, 3.049	2.16, 3.057	11.3%, 0.262%
6%	1.88, 3.062	1.91, 3.181	1.6%, 3.89%

Table 9. The false positive and false negative rates of GC detection for 1000 different circuits, for two different types of thresholds for the c432 benchmark for 1% error rate with 200 constraints

	False Positive Rate	False Negative Rate
Threshold Set at Mean	25.1%	5.05%
Threshold Set at greater than a Known Non-GC Circuit Average Value	20.3%	5.95%

Table 9 presents the results for 1000 different circuits, 500 of which have a single NAND gate added near gate 25 in the center of the circuit. This would qualify as a very difficult case of GC detection. Given this scenario, if the threshold for determining if a ghost gate is present is set as whether the average of the α-values is greater than the mean of the α-values of the non-GC circuits, then there is only a 5.05% false negative rate. That means that about 95% of the ICs are properly characterized as having ghost circuitry. As expected, 25% of the time this results in a false positive rate. For this benchmark, with no measurement error and 200 constraints both the false positive rate and the false negative rate go to zero.

On the other hand if we use an average of a single non-GC circuit's α-values to set the threshold value, then the false negative rates increases to 5.95%, while the false negative rates decreases to 20.3%. Setting the threshold value is an important parameter in this analysis. We propose a minimization of the sum of the false positive and false negative rate for a learning set of ICs to determine the best threshold value.

In another GC detection technique, we manipulate constraints and the objective function in the nonlinear program, by adding an extra variable to the right side of each constraint. If the gates can be characterized in a more accurate and consistent manner with this addition, there is a strong indication of the presence of ghost circuitry.

To defeat this approach, the attacker would need to add gates such that they were always correlated with another gate. An automated search across the netlist could determine the best location for adding the gate(s). Another difficult attack to detect is if the attacker optimizes circuit design to use the saved delay and power characteristics to hide the ghost circuitry. In general, the attacker will need to carry out an optimization to determine the location to hide the ghost circuit to attempt to avoid diction by our techniques. Simple or random GC insertion will be detected.

7 Conclusion

We have developed a system of techniques for ghost circuitry detection. The techniques apply a system of non-destructive delay and/or power measurements followed by singular value decomposition for gate-level characterization. Once the GLC is completed, statistical data analysis is carried out to determine whether ghost circuitry has been added or not.

References

[1] Defense Science Board (DSB) study on high performance microchip supply (2006), http://www.acq.osd.mil/dsb/reports/2005-02-hpmsreportfinal.pdf

[2] Agrawal, D., Baktir, S., Karakoyunlu, D., Rohatgi, P., Sunar, B.: Trojan detection using ic fingerprinting. In: IEEE Symposium on Security and Privacy (SP), pp. 296–310 (2007)

[3] Anderson, R., Bond, M., Clulow, J., Skorobogato, S.: Cryptographic processors-a survey. Proceedings of the IEEE 94(2), 357–369 (2006)

[4] Anderson, R.J.: Security Engineering: A guide to building dependable distributed systems. John Wiley and Sons, Chichester (2001)

[5] Bernstein, K., Frank, D.J., Gattiker, A.E., Haensch, W., Ji, B.L., Nassif, S.R., Nowak, E.J., Pearson, D.J., Rohrer, N.J.: High-performance CMOS variability in the 65-nm regime and beyond. IBM Journal of Research and Development 50(4/5), 433–450 (2006)

[6] Hwang, D., Schaumont, P., Tiri, K., Verbauwhede, I.: Securing embedded systems. IEEE Security & Privacy 4(2), 40–49 (2006)

[7] Jha, N.K., Gupta, S.: Testing of Digital Systems. Cambridge University Press, Cambridge (2003)

[8] Koushanfar, F., Potkonjak, M.: CAD-based security, cryptography, and digital rights management. In: Design Automation Conference, DAC (2007)

[9] Lofstrom, K., Daasch, W.R., Taylor, D.: IC identification circuits using device mismatch. In: International Solid State Circuits Conference (ISSCC), pp. 372–373 (2000)

[10] Menezes, A., van Oorschot, P., Vanstone, S.: Handbook of Applied Cryptography. CRC Press, Boca Raton (1997)

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

[12] Su, Y., Holleman, J., Otis, B.: A 1.6J/bit stable chip ID generating circuit using process variations. In: International Solid State Circuits Conference, ISSCC (2007) (to appear)

[13] Suh, G., Devadas, S.: Physical unclonable functions for device authentication and secret key generation. In: Design Automation Conference (DAC), pp. 9–14 (2007)

[14] Yablonovitch, E.: Can nano-photonic silicon circuits become an intra-chip interconnect technology? In: IEEE/ACM International Conference on Computer-Aided Design (ICCAD), p. 309 (2007)

[15] Alkabani, Y., Koushanfar, F., Kiyavash, N., Potkonjak, M.: Trusted integrated circuits: A nondestructive hidden characteristics extraction approach. In: Solanki, K., Sullivan, K., Madhow, U. (eds.) IH 2008. LNCS, vol. 5284, pp. 102–117. Springer, Heidelberg (2008)

[16] Alkabani, Y., Massey, T., Koushanfar, F., Potkonjak, M.: Input vector control for post-silicon leakage current minimization in the presence of manufacturing variability. In: Design Automation Conference (DAC), pp. 606–609 (2008)
[17] Dabiri, F., Potkonjak, M.: Hardware aging-based software metering. In: Design, Automation, and Test in Europe, DATE (2009)
[18] Koushanfar, F., Boufounos, P., Shamsi, D.: Post-silicon timing characterization by compressed sensing. In: IEEE/ACM International Conference on Computer-Aided Design (ICCAD), pp. 185–189 (2008)
[19] Shamsi, D., Boufounos, P., Koushanfar, F.: Noninvasive leakage power tomography of integrated circuits by compressive sensing. In: International Symposium on Low power electronics and design (ISLPED), pp. 341–346 (2008)
[20] Yuan, L., Qu, G.: A combined gate replacement and input vector control approach for leakage current reduction. IEEE Trans. Very Large Scale Integr. Syst. 14(2), 173–182 (2006)
[21] Ercegovac, M.D., Lang, T., Moreno, J.H.: Introduction to Digital Systems (1999)
[22] Kocher, P., Jaffe, J., Jum, B.: Differential Power Analysis. In: International Cryptology Conference on Advances in Cryptology (1999)
[23] Kuhn, M.: Trojan hardware – some strategies and defenses. Slides from the Schloss Dagstuhl (2008),
http://www.cl.cam.ac.uk/~mgk25/dagstuhl08-hwtrojan.pdf
[24] Rajsuman, R.: Iddq testing for CMOS VLSI. Proceedings of the IEEE 88(4), 544–568 (2000)
[25] Alkabani, Y., Koushanfar, F., Potkonjak, M.: Remote Activation of ICs for Piracy Prevention and Digital Right Management. In: IEEE/ACM International Conference on Computer Aided Design, ICCAD (2007)