# A Novel Threshold Voltage Defined Multiplexer for Interconnect Camouflaging

***Jae-Won Jang[1], and Swaroop Ghosh[2]***
Department of Computer Engineering
Pennsylvania State University
University Park, PA, USA
Contact Author Email: [1]jxj328@psu.edu, [2]szg212@engr.psu.edu

**Abstract:** *Semiconductor supply chain is increasingly getting exposed to variety of security attacks such as Trojan insertion, cloning, counterfeiting, reverse engineering (RE), piracy of Intellectual Property (IP) and side-channel analysis due to involvement of untrusted parties. Camouflaging of gates has been proposed to hide the functionality of gates. However, camouflaging is associated with significant area, power and delay overhead. In this paper, we propose camouflaging of interconnects using multiplexers (muxs) to protect the IP. A transistor threshold voltage-defined pass transistor multiplexer is proposed to prevent its reverse engineering since transistor threshold voltage is opaque to the adversary. The proposed pass transistor based multiplexer having more than one input, hides the original connectivity of the net. The camouflaged design operates at nominal voltage and obeys conventional reliability limits. A small fraction of nets can be camouflaged to increase the RE effort extremely high while keeping the overhead low. We propose controllability, observability and random-net based selection for camouflaging.* ***Simulation results indicate 32-81% area, 13-89% delay and 33-84% power overhead when 5-15% nets are identified and camouflaged using the proposed technique.***

**Keywords:** Reverse Engineering, Camouflaging, Threshold-Defined Multiplexer.

## Introduction

Reverse Engineering (RE) of an Intellectual Property (IP) [1-2] is a process of identifying its design, functionality, and structure. In the RE, the adversary de-layers the Integrated Circuit (IC), determines the gate functionalities and their connectivity information, and, reconstructs the netlist (Fig. 1). This technique has been originally used by industries with the mindset of gathering information on its competitors, to confirm the functionality of their own design, and to ensure the legitimacy of circuits against piracy. However, well-equipped adversaries can exploit this technique with an ill-intention to steal, and pirate a design to siphon large profits.

Camouflaging of gates have been proposed [3] [4] for hiding the logic functionality and making the RE economically non-profitable or extremely difficult. The primary objective of camouflaging gates is to hide the functionality of *few chosen gates* (since camouflaged gates are typically area, delay and power intensive) to increase RE effort of adversary while minimizing power, performance and area overhead. The camouflaged gates can assume functionalities such as AND, OR, XOR, etc. Although the exact gate functionality is hidden, the adversary can still create a partial netlist with other known gates and go through a guess-and-validate process to RE the missing gate functionality. This is achieved by making a guess on the gate function, finding reasonable test patterns to confirm the guess, and then applying these patterns to both a partial netlist and a golden chip. If the outputs match then the guess is correct, else the adversary guesses a new gate functionality and repeats the steps. This procedure is shown in Fig. 2(a). The RE effort is also shown which involves the time needed to identify all camouflaged gate functionalities.

It has been shown that careful camouflaging of ~10-40% gates can increase the RE effort significantly [11]. Therefore, significant research is being devoted to develop methodologies for IC camouflaging ranging from using dummy contacts [1-2] to programmable standard cells [5] and filler cells [6]. A new camouflaging technique based on the transistor threshold voltage ($V_T$) programmable switch that turns ON/OFF based on $V_T$ assertion is also proposed for camouflaging [7] [8].

Split manufacturing [1] is another technique to protect the IP. However, it addresses the RE of IP and Trojan insertion during manufacturing process. In contrast to the regular manufacturing procedure, the front-end

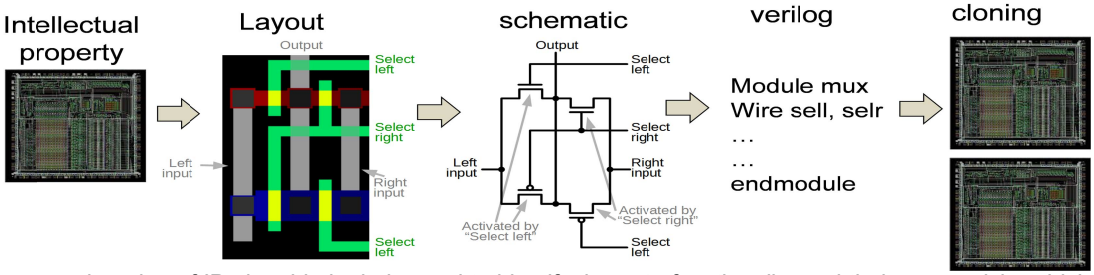



**Fig. 1.** Reverse engineering of IP: the chip is de-layered to identify the gate functionality and their connectivity which is used to reconstruct the schematic and netlist. The objective is to clone the design.

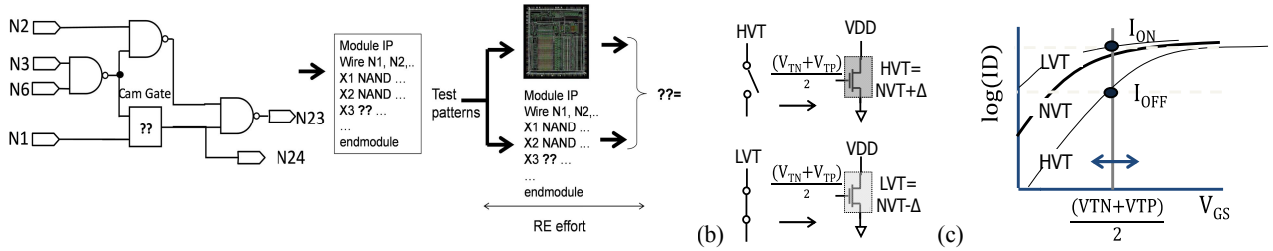


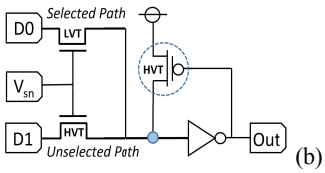

**Fig. 2** (a) When a gate is camouflaged, the adversary extracts the partial netlist, guesses the missing gate functionality ("??") and applies specific test pattern to match the output against actual chip to confirm the guess. The RE effort is the time invested by adversary to find appropriate test pattern and identify the camouflaged gate functionality; (b) $V_T$ programmable switch. HVT: OFF, LVT: ON. PMOS works similarly; and (c) cartoon of I-V curves of NVT, HVT and LVT transistors. The $I_{ON}$ and $I_{OFF}$ depends on the LVT and HVT values as well as on gate voltage biasing.

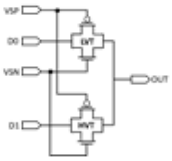
(transistors) is manufactured in an untrusted foundry whereas the back-end (interconnect) is manufactured in trusted facility. This makes the RE and Trojan insertion more challenging for the adversary present in untrusted foundry since the connectivity information is hidden. Furthermore, since the front-end fabrication cost is higher than the back-end, the cost benefit of outsourcing the fabrication is still preserved without increasing the security risks. Although this technique is effective in preventing RE, it can be susceptible to yield loss during stacking due to via misalignment. Furthermore, it still requires trusted foundries and costly assembly processes.

In contrast to gate camouflaging, we propose an interconnect camouflaging technique to hide the connectivity information between gates. Conceptually, the effect is similar to split manufacturing. However, this technique does not require splitting of layers between trusted and untrusted foundry; and only few selected nets are camouflaged to incur small overhead while increasing RE effort of adversary. The proposed camouflaging is achieved by inserting our novel RE-resistant multiplexers (mux design based on $V_T$-defined switches [7] which requires no select bit and leaves no layout trace) in the design.

### Threshold Voltage Defined Switch

We use the programmable switch proposed in [7] that turns ON/OFF based on $V_T$ asserted on it. In this work, the switch is optimized to suit the mux application. The switch is realized by using conventional NMOS and PMOS transistors with the gate biased at the mid-point between nominal NMOS and PMOS threshold voltages i.e., $0.5(V_{TN}+V_{TP})$. Therefore, the switch conducts when low $V_T$ (LVT) is assigned during manufacturing. This is due to the fact that $V_{GS}= 0.5(V_{TN}+V_{TP}) >$ LVT. The switch stops conducting when high $V_T$ (HVT) is assigned ($V_{GS}$ < HVT). This is depicted in Fig. 2(b) for NMOS transistor. The cartoon of transistor I-V curves for NVT, LVT and HVT transistor is shown in Fig. 2(c). The $I_{ON}$ and $I_{OFF}$ that can be obtained by assigning LVT and HVT is also shown. A good $V_T$ defined switch should offer high ON current and low OFF current. The gate voltage, HVT, LVT values and transistor sizes are tuned to maximize the $I_{ON}/I_{OFF}$ ratio. For NMOS-switch, higher HVT values and lower gate voltage is good for $I_{OFF}$ (leakage) whereas lower LVT and higher gate voltage is good for $I_{ON}$ (performance). Vice-versa is true for PMOS-switch. The switch optimization in presence of these conflicting requirements is described in Section IIC.




(c)

| Type | Delay (ps) | Power (uW) | Area (um²) |
|---|---|---|---|
| 2:1 NMOS MUX | 33.5 | 2.89 | 0.598 |
| 4:1 NMOS MUX | 39.5 | 3.372 | 1.098 |
| 8:1 NMOS MUX | 50.7 | 4.31 | 2.1 |
| 16:1 NMOS MUX | 72.42 | 6.21 | 4.1 |

| Type | Delay (ps) | Power (uW) | Area (um²) |
|---|---|---|---|
| 2:1 TR MUX | 43.6 | 8.1 | 1.31 |
| 4:1 TR MUX | 64.96 | 12.2 | 2.2 |
| 8:1 TR MUX | 98.3 | 28.6 | 4.27 |
| 16:1 TR MUX | 155.6 | 44.3 | 8.41 |

**Fig. 3** (a) the proposed pass transistor NMOS-only based 2:1 mux; (b) the transmission gate based 2:1 MUX; and (c) attributes of the proposed N:1 MUX for NMOS-only MUX and Transmission(TR)-gate MUX

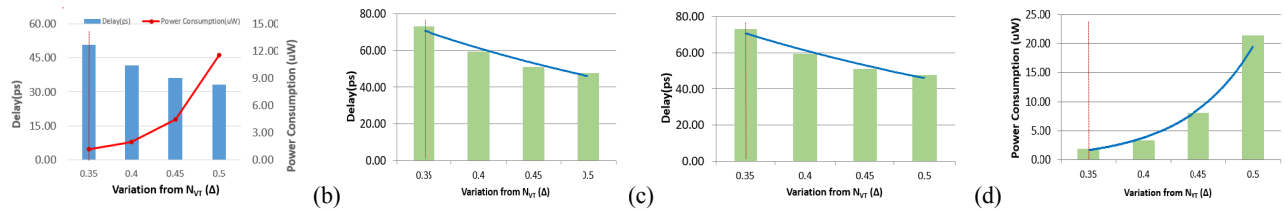



**Fig. 4.** Selection of $V_{SN}$ and offset from NVT: (a) 8:1 mux delay and leakage vs offset; (b) 8:1 mux delay vs $V_{SN}$; (c) 16:1 mux delay vs offset; and, (d) 16:1 mux leakage vs offset. The optimal choice of offset is also shown by dashed lines.

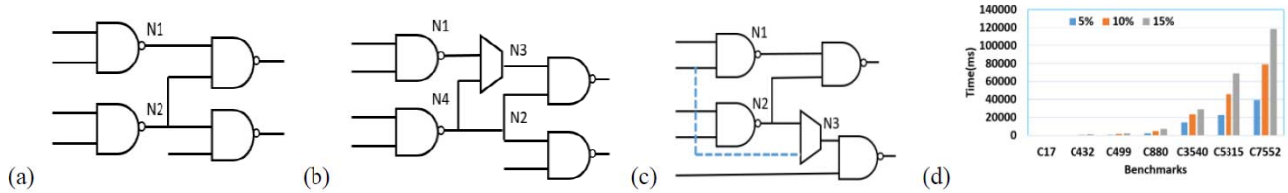



**Fig. 5.** Example of qualified nets: (a) original circuit; (b) single-fan-out net which is selected for mux insertion. Since N1 cannot float, the adversary can easily guess that N1 connects to N3. We disqualify such nets to prevent mux insertion; (c) multi-fan-out net N2 is selected for mux insertion. The adversary cannot figure the connection between N2 and N3. Such nets are qualified to enable selection; and, (d) RE effort with percentage camouflaging for 2:1 mux based on controllability and observability of nets

### Threshold Voltage Defined Multiplexer

In this section, we extend VT defined switches to design a novel VT defined multiplexer for interconnect camouflaging and provide design space.

The $V_T$ defined switch [7] is optimized to suit the mux application. In the proposed mux, the real path contains LVT pass transistor and the fake paths contain HVT pass transistor (Fig. 3(a)). This eliminates the need of a mux select input as $V_T$ value inherently determines the input selection. Since an NMOS transistor cannot pass a strong input '1', we incorporate a level restoring weak HVT PMOS transistor (highlighted with dashed-circle in Fig. 3(a)) to pull the NMOS pass transistor output to full-rail. The level restoring transistor helps full voltage swing of the degraded input and improves the low-to-high transition. Furthermore, it eliminates the static current from the *output inverter*. The sizing of this level restoring PMOS transistor is done carefully so that it does not fight with mux inputs. The alternative design technique to avoid level restoring transistor is to use full transmission gates (with NMOS and PMOS in parallel as shown in Fig. 3(b)). This method will allow both strong input '0' and '1' to be passed through the muxs, but incurs enormous power, delay, and size overhead due to requirement large PMOS transistors. As Fig. 3(c) shows the comparison result, this design increases area and power overhead especially for wide input mux designs. The pass transistor NMOS-only mux logic allows the proposed design to be compact without incurring significant overhead.

For the design space exploration, we have used Nangate 45nm technology [9]. The LVT and HVT values are chosen by determining their offset (Δ in Fig. 2(b)) from NVT value. For example, if the NVT of NMOS transistor is 0.62V an offset of 0.1V (i.e., Δ = 0.1V) means that the LVT is 0.52V and HVT is 0.72V. We sweep both offset and gate voltages (VSN) and calculate the delay and leakage power. The offset voltage (Δ) is swept from 0.30V to 0.45V in steps of 0.05V for NMOS as well as PMOS. The VSN is swept from 0.1V to 0.5V in steps of 0.05V. Fig. 4(a) shows the delay and leakage power values with offset and Fig. 4(b) shows the delay when VSN are varied. From these two plots, we choose the optimum values of Δ (= 0.35V), and VSN (=0.7V) which are used for simulating 2:1, 4:1, 8:1, and 16:1 muxs. A similar trade-off study is conducted between the delay and leakage power for 16:1 mux (Fig. 4(c)-(d)). A Δ of 0.4V, VSN of 0.7V is selected. Fig. 3(b)-(c) shows the area power and delay of (NMOS-only / Transmission-gate) 2:1, 4:1, 8:1 and 16:1 muxs for simulation of ISCAS85 [10] benchmarks. Note that VSP is shown in the exploration since we are not using VT defined PMOS transistor in the proposed pass transistor mux.

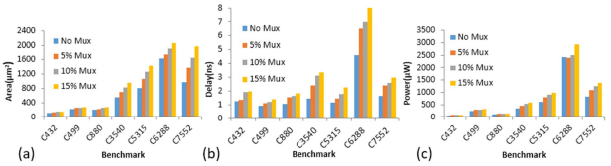



Fig. 6. Area, delay, power values of different benchmarks based on the percentage replacements (5 to 15%) of nets by 2-1 mux; (a) area; (b) delay; and (c) power of net selection methodology.

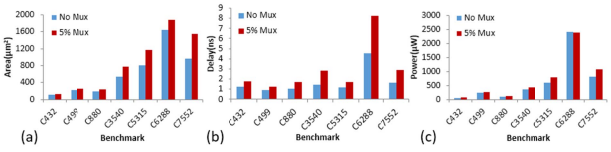



Fig. 7. Area, delay, power values of different benchmarks based on the percentage replacements (5%) of nets by 4-1 mux; (a) area; (b) delay; and (c) power of net selection methodology.

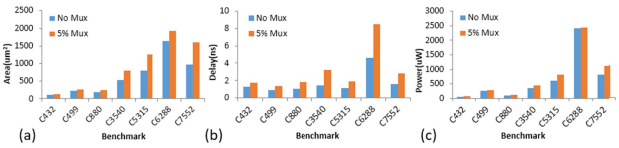



Fig. 8. Area, delay, power values of different benchmarks based on the percentage replacements (5%) of nets by 8-1 mux; (a) area; (b) delay; and (c) power of net selection methodology.

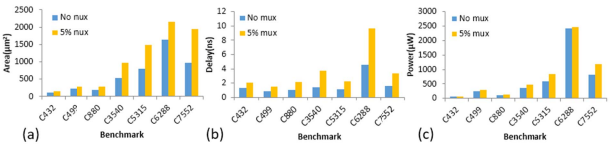



Fig. 9. Area, delay, power values of different benchmarks based on the percentage replacements (5%) of nets by 16-1 mux; (a) area; (b) delay; and (c) power of net selection methodology.

**Table 1.** Average percentage overhead of N:1 mux for 5% camouflaged nets

| Gate | Area | Delay | Power |
|---|---|---|---|
| 2:1 MUX | 15% | 20% | 14% |
| 4:1 MUX | 22.43% | 38.12% | 13.99% |
| 8:1 MUX | 25.32% | 41.08% | 16.23% |
| 16:1 MUX | 34.99% | 49.29% | 19.41% |

## Threshold Voltage Defined Interconnects

In contrast to previous works revolving around camouflaging gates to maximize the RE effort, we camouflage the nets that cannot be reverse engineered through simple intuition. For example, if net N1 (which is a single fan-out net) in Fig. 5(a) is camouflaged using a mux as shown in Fig. 5(b) then reverse engineering becomes straightforward. This is due to the fact that N1 cannot float in a valid design. This leaves the adversary to conclude that N1 and N2 are connected without running any simulation. We discard such single fan-out nets from the selection algorithm. However, if a multi-fan-outs net such as N2 is selected for mux insertion, then the adversary cannot figure out the connection between N2 and N3 (Fig. 5(c)). Such nets are considered qualified nets in the proposed camouflaging procedure.

In addition to utilizing interconnects, our technique also identifies interconnects based on quantifiable values to maximize the RE effort. We first compute the controllability (CC) and observability (Obs) values for every nets and its number of fan-outs in a circuit. The '0' and '1' controllability (CC0 and CC1) and observability values provides a relative difficulty of controlling and observing a logic signal of a particular net. By selecting a net with low CC0, CC1 and Obs values, it is possible to increase the adversary RE. Note that, the controllability and observability of the net is assigned the same value as the controllability and observability of the gate that is driving the net. For the nets with fan-outs, the controllability and observability is propagated to all fan-out nets. Fig. 5(d) shows the RE effort for 2:1 mux using controllability / observability based selection. This was evaluated using Synopsis Design Compiler for ISCAS85 benchmarks [10]. Since threshold-defined muxs are not included in standard cell library, we have created a liberty file of the proposed muxs with values characterized using HSPICE simulation.

## Results and Discussions

Fig. 6(a-c) shows area, delay, and power of benchmarks replaced with 2:1 muxs for 5%, 10% and 15% camouflaging using the controllability/observability based net selection methodology. Comparing to the original ("No Mux") design, the average overhead is found to be 15% (area), 25% (delay) and 14% (power) for 5% camouflaging. The values for 10% camouflaging are 26%, 41% and 22%. For 15% camouflaging, the values are 33%, 44% and 29%. From these results, we can observe the linear relation of overhead with respect to the number of camouflaged nets. In order to further increase the RE effort (discussed next) and to test the flexibility of our proposed mux, we also tested our methodology using wider 4:1, 8:1 and 16:1muxes. For these simulations, we only replaced 5% of the nets. Fig. 7(a-c), 8(a-c), and 9(a-c) shows area, delay, and power values of benchmarks. The result of these overhead is shown in Table 1. From this, we can conclude that wider input muxs can incur affordable increase in design overhead.

## Conclusions

We propose threshold voltage-defined pass transistor based multiplexer to camouflage the interconnects of IPs both logically and physically. Compared to existing split manufacturing, the proposed interconnect camouflaging does not require any process change and does not incur extra assembly cost while promises to increase the RE effort. Careful selection of nets for camouflaging can mitigate the overhead compared to gate camouflaging technique.


## Acknowledgements

This paper is supported by Defense Advanced Research Projects Agency (DARPA) under award #D15AP00089


## References


1. F. Imeson, et al. "Securing computer hardware using 3D integrated circuit (IC) technology and split manufacturing for obfuscation." In Presented as part of the 22nd USENIX Security Symposium (USENIX Security 13), pp. 495-510. 2013.
2. Rajendran, Jeyavijayan, Michael Sam, Ozgur Sinanoglu, and Ramesh Karri. "Security analysis of integrated circuit camouflaging." In Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security, pp. 709-720. ACM, 2013.
3. SypherMedia, "Syphermedia library circuit camouflage technology." http://www.smi.tv/solutions.htm.
4. J. Rajendran, M. Sam, O. Sinanoglu, and R. Karri, "Security Analysis of Integrated Circuit Camouflaging," in *Proceedings of the 2013 ACM SIGSAC Conference on Computer and Communications Security*, CCS'13, (New York, NY, USA), pp. 709-720, ACM, 2013.
5. R. Cocchi, J. Baukus, B. Wang, L. Chow, and P. Ouyang. "Building block for a secure CMOS logic cell library." U.S. Patent 8,111,089, issued February 7, 2012.
6. L. Chow, L. Wai, J. Baukus, B. Wang, and R. Cocchi. "Camouflaging a standard cell based integrated circuit." U.S. Patent 8,151,235, issued April 3, 2012.
7. I A. Iyengar, and S. Ghosh. "Threshold Voltage-Defined Switches for Programmable Gates." GOMACTech, 2015
8. M. Mera, M. Massad, S. Garg. "Threshold-Dependent Camouflaged Cells to Secure Circuits Against Reverse Engineering Attacks" IEEE Computer Society Annual Symposium on VLSI (ISVLSI), 2016.
9. Predictive technology model, http://ptm.asu.edu/
10. http://www.pld.ttu.ee/~maksim/benchmarks/iscas85/verilog.
11. Y. Wang, U. Arslan, N. Bisnik, R. Brain, S. Ghosh, et al. "Retention time optimization for eDRAM in 22nm tri-gate CMOS technology." In 2013 IEEE International Electron Devices Meeting. 2013.