S*FSM: A Paradigm Shift for Attack Resistant FSM Designs and Encodings

Mike Borowczak and Ranga Vemuri
School of Electronics and Computing Systems
The University of Cincinnati
Emails: borowcm@email.uc.edu, ranga.vemuri@uc.edu

Abstract

While hardware design focuses on creating minimally-sized circuits, this paper proposes that security-centric designs require a departure from this mentality. The need for built-in protection mechanisms at all levels of design is paramount to providing cost-effective secure systems. We focus on the high-level design of sequential circuits by targeting Finite State Machines (FSMs) and their vulnerability to non-invasive, side channel based, attacks. The unconventional paradigm shift needed is justified by showing that conventional, minimalism-based, FSM synthesis and encodings allow direct correlation between state/transitions and Hamming Models. A two-fold method, involving structural modifications and specific encoding strategies, is proposed for side-channel secure FSM (S*FSMs). Preliminary high-level simulations show the effectiveness and potential for security driven S*FSM synthesis methods to mitigate the relationship between attack models and underlying hardware implementations.

1. Introduction

The underlying concepts used in performing power based side channel attacks have been used since the dawn of the 20th century: seven decades before the radio-communication security concerns highlighted in TEMPEST [1] and a century before their discovery and formal presentation by Kocher, Jaffe and Jun in their seminal work describing Differential Power Analysis based attacks [2]. In 1903, Nobel Prize winning physiologist Willem Einthoven began capturing and correlating electrocardiogram signals to numerous cardiovascular conditions [3]. The foundation of his work as well as side channel attacks is based on the relative ease of modeling the relationship between a target’s internal operation and its byproducts.

Regardless of the potential target, whether it is biological, electronic, or even a social collective, to ensure its protection from side channel attacks the relationship between the models and the byproduct must be eliminated. Figure 1 depicts a common target and a side channel model and their use in a generic side channel attack. A typical side-channel attack, on a device $T$, implementing a function $F$ with input $A$, mask $K$ (e.g. a secret key), output $B$, and side channel $T_{SC}$ requires three main components:

1) Access to a target device’s side channel: $T_{SC} \propto F$;
2) A model of the side channel: $M_{SC}(K_G, A|B)$, where $K_G$ is a mask guess; and
3) A relationship between $T_{SC}$ and $M_{SC}$ allowing ranking of mask guesses.

With these three components in mind, one of the most prevalent side channels due to its accessibility and low capture cost, is a device’s power consumption [2]. Second, when targeting the power side channel, two generic power models exist. The first, described using Equation (1), is the Hamming Weight (HW) of internal register $S$. The second, described by Equation (2), is the Hamming Distance (HD) between two successive states of register $S_t$ and $S_{t+1}$. Variability in either of the Hamming models implies a correlated variability in the power consumed, and thus a vulnerability of the mask $K$ [4].

$$\text{HW}(S[x..0]) = \sum_{i=0}^{x} S[i] \quad (1)$$

$$\text{HD}(S_t[x..0], S_{t+1}[x..0]) = \sum_{i=0}^{x} S_t[i] \oplus S_{t+1}[i] \quad (2)$$

Most of the present day research effort focuses on removing the variability of the side channel itself. If no variability exists within the side channel then no useful information is transmitted and any correlation...
between the model and the side channel will be random in nature - $T_{SC} \not \propto M_{SC}$.

These low level approaches generally involve creating customized dual routed, balanced cells [5]–[7]. Unfortunately, while recent advances in the use of charge recovery logic in dual-rail logic styles such as SABL have mitigated power penalties [8], [9], low-level approaches typically come at significant hardware cost in terms of standard cell restrictions, layout area, speed and design effort. Other research has focused on masking the relationship between the side channel model $M_{SC}$ and the underlying implemented function $F$. These algorithm-specific solutions are both impractical for widespread, generic use and are prey to higher-order side channel attacks.

The major contribution of this paper is the elimination of any correlation between $T_{SC} \not \propto M_{SC}$ in FSMs by removing the variability of both Hamming Models. Thus, we provide a high-level method to protect any FSM based devices against power side channel attack.

2. Motivational FSM Example

FSMs represent a computational model typically used heavily in the design of circuit level sequential hardware and computing programs found in everything from locking mechanisms and vending machines to communication protocols. While the security of a vending machine is hardly comparable to that of a communication protocol, motivation for Secure FSMs (S*FSMs) starts with a basic FSM rather than one as complex as a communication protocol.

Imagine the classical computer architecture problem of designing a branch predictor. Figure 2 shows the branch predicting FSM using the expected 2-bit saturating counter: each state of the counter is assigned a minimal binary encoding. The implementation is a generic four-state Moore machine: if the previous branch was taken ($T = 1$), the branch-predictor moves towards (or remains in) the Strongly Taken state (ST), otherwise when the previous branch is not taken ($T = 0$) the predictor moves towards the Strongly Not Taken state (SNT). The two intermediate states, Weakly Taken and Weakly Not Taken (WT and WNT), aide in providing an increased prediction history or memory, greatly increasing its accuracy [10], [11].

![Branch predictor with transition conditions and state encodings.](image)

Figure 2. A branch predictor implemented as a two-bit saturating counter.

The minimally encoded two-stage branch predictor
shown in Figure 2 is summarized in the extended transition table in Figure 2(b) which shows both the HW and HD ranging from 0 to 2.

3. S*FSM

In order to effectively resist attack, a side-channel hardened FSM must satisfy several criteria. To fully specify these hardening requirements an FSM is represented as a directed graph $G(V,E)$ with vertices $V$ and edges $E$. Let $V(G)$ contain the set of all vertices $v_i$, where each vertex is defined by a unique encoding. For clarity, we also include state labels in our diagrams. Similarly, $E(G)$ contains all directed edges $e_i = \{v_x, v_y, t\}$, each triple consists of two vertices, a source $v_x$ and destination $v_y$, and the conditions for transition $t$. Note, that later in our discussion on secure encodings we simplify without loss of generality, our model to an undirected graph. This will thereby remove the need for transitional conditions, thus edges will simply be the unordered pairing of vertices, $e_i' = \{v_a, v_b\}$.

We present two conditions for Hardened FSMs that render Hamming based side-channel models ineffective:

1) $\forall v : v \in V \rightarrow HW(v) = c_1$.
2) $\forall e : e(v_x, v_y, t) \in E, v_x, v_y \in V \rightarrow HD(v_x, v_y) = c_2$.

Simply stated, the first condition says that each FSM state (vertex $v$ in $G$) must have the same HW. This can easily be achieved by selecting an appropriate encoding strategy in which every encoding has the same number of bits turned on - notice that this immediately increases the minimal encoding bit length. The second condition states that each transition (edge $e$ in $G$) should have the same HD. This requirement puts additional strain on finding an appropriate state encoding. Note that while both conditions impact the encoding of the Hardened FSM, only the second condition impacts the FSMs physical structure.

In order to create S*FSMs, we propose a two-part FSM hardening process which includes 1) structural modifications to physical topology and 2) the use of intelligent encoding schemes.

3.1. Physical Topology

The need for structural modification in FSM is easily demonstrated when considering the need for a constant HD between all state transitions. Consider first a single node FSM with one self-looping transition. Albeit not useful, regardless of its encoding, it will always have a constant HW as well as a HD of zero. Clearly, it satisfies both conditions needed for a secure FSM. On the other hand, the two-node FSM in 3(a) can never satisfy the second condition. Regardless of encoding selected for states A and B, the HD between two unique states can never be zero, while the HD between any node and itself is always zero.

The only solution capable of eliminating this conflict is the unrolling of self-loops - essentially the opposite of state collapsing which is often used to reduce FSM complexity [12]. Thus, in order to satisfy the second condition, a multi-state, side-channel hardened, FSM can not contain any self-loops. Algorithm 1, FSM Loop Remove, is a simple yet effective method of removing self-loops for all FSMs. To achieve this goal it takes each node, checks for a self-loop, and upon finding one, removes it from the transition list. A new state is added with two corresponding edges, each with the same transition condition(s). Finally all out-going edges are replicated, maintaining the original functionality of the FSM. Without loss of generality, we assume that self-loops should not be present in any hardened FSM.

### Algorithm 1 FSM Loop Unroll

1: procedure LOOPREMOVE($V, E$)
2: for each $v \in V$ do
3: if $\exists e : e(v, v, t) \in E$ then $\triangleright$ Self-loop w/cond. $t$
4: $E \leftarrow E - e$
5: $v' \leftarrow v$ $\triangleright$ Create new node
6: $V \leftarrow V \cup \{v'\}$
7: $E \leftarrow E \cup \{\{v, v', c\}, \{v', v, c\}\}$
8: for each $u \in V | u \neq v, e(v, u, c) \in E$
9: $E \leftarrow E \cup e(v', u, c)$
10: end for
11: end if
12: end for
13: end procedure

3.2. Encoding

The main challenge to guarantee security in S*FSMs is applying an intelligent, state encoding to the secured structure. Table 1 compares three encoding strategies for the structurally modified branch predictor shown in Fig. 4. The first two strategies, Minimal or Binary Encoding (BE) and One-Hot Encoding (OH), are available in any standard synthesis flow. The final strategy, is our hybrid encoding scheme for optimal
secure encodings: S*Optimal (S*O). Notice that both the OH and S*O columns show constant HW and HD measurements regardless of any transition vector applied, while the minimal BE can range from [0-2] and [1-3] respectively. Our experimental results highlight the attackability of arbitrary minimal encodings.

### 4. Experimental Results

In this section, we validate our 2-fold method for S*FSMs by demonstrating the variability of two Hamming models on two different FSMs. The first is the unsecured FSM shown in Figure 3, while the second is the unrolled structurally modified FSM in Figure 3(b). Each FSM is encoded using a traditional BE method as well as secure encoding methods when they are feasible (OH and/or S*O).

Our experiment targets the worst-case variability of the side channel models: if the model itself shows no variability, it is rendered useless. In order to measure the variability of both the HW and HD models, a simple event driven FSM simulator was constructed. The high-level simulator applied random transition vectors (ranging in size from 10 - 50000 transitions) to a given FSM. During each transition, the current state and transition path was recorded and the HW and HD were computed. While an over-simplification of a hardware system, recall the goal is to whether a relationship between the two recorded events (State and Transition) and the models (HW and HD) exists.

Table 2 summarizes the simulation results for the standard FSM as well as the structurally secure S*FSM for three unique encoding schemes including: BE, OH and S*O. Each entry shows the average correlation over 100 simulation runs. The results highlight first and foremost that a proper encoding strategy is key: FSMs, regardless of structure are insecure using typical binary encoding schemes. Secondly, the only way to eliminate variability of both the HD and HW models is through a combined use of a secure encoding scheme.

#### Table 1. Three encoding strategies applied to the structurally secured FSM seen in Fig. 4.

<table>
<thead>
<tr>
<th>State</th>
<th>Minimal/Binary</th>
<th>One-Hot</th>
<th>S*Optimal</th>
</tr>
</thead>
<tbody>
<tr>
<td>ST1</td>
<td>101</td>
<td>100000</td>
<td>0101</td>
</tr>
<tr>
<td>ST2</td>
<td>100</td>
<td>010000</td>
<td>0110</td>
</tr>
<tr>
<td>WT</td>
<td>011</td>
<td>001000</td>
<td>1100</td>
</tr>
<tr>
<td>WNT</td>
<td>010</td>
<td>000100</td>
<td>1001</td>
</tr>
<tr>
<td>SNT2</td>
<td>001</td>
<td>000010</td>
<td>0011</td>
</tr>
<tr>
<td>SNT1</td>
<td>000</td>
<td>000001</td>
<td>1010</td>
</tr>
<tr>
<td>HW Range</td>
<td>[0-2]</td>
<td>[1]</td>
<td>[2]</td>
</tr>
<tr>
<td>HD Range</td>
<td>[1-3]</td>
<td>[2]</td>
<td>[2]</td>
</tr>
</tbody>
</table>

Figure 3. Motivating structural modification: (a) Fails Condition 2 for hardened FSMs regardless of encoding; (b) Unrolled, structurally secure FSM.

Figure 4. Original two-stage branch predictor after a structural modification algorithm is applied.
(S^*O or OH) and structural modification.

5. Conclusions and Future Work

We have identified two conditions that enhance the side channel attack-resistance of FSMs from a model-centric perspective. A generic, two-stage branch predictor motivated and highlighted that even simple side channel models can be correlated to the internal FSM states. Our goal, to remove this correlation, was achieved by applying simple transforms to create S^*FSMs. Our experimental results support our method, showing that both structural modification and encoding are necessary and sufficient to remove the correlation between Hamming models and an FSMs state or the transitions between its states.

To validate the sufficiency of our method, and its integration with existing low level protection mechanisms, the FSM models and encoding will be implemented and characterized in Spice using both standard CMOS and custom side-channel resistant logic cells. Additionally, to demonstrate the real-world practicality and integration with existing low level protection methods, these ideas are being applied to larger more complex FSMs - specifically those found within standard logic synthesis processes.

Table 2. Correlation (when it exists) between HW/HD Model and State/Transition for three encoding strategies on Standard and S^*FSMs.

<table>
<thead>
<tr>
<th>State/HW</th>
<th>Standard FSM</th>
<th>S^*FSM</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>BE</td>
<td>OH</td>
</tr>
<tr>
<td>10</td>
<td>0.71</td>
<td>—</td>
</tr>
<tr>
<td>500</td>
<td>0.62</td>
<td>—</td>
</tr>
<tr>
<td>5000</td>
<td>0.63</td>
<td>—</td>
</tr>
<tr>
<td>50000</td>
<td>0.63</td>
<td>—</td>
</tr>
</tbody>
</table>

### References


