# Memory Cost Analysis for Sequential Logic Locking FSMLock

John Evans, Marcin Lukowiak
Department of Computer Engineering
Rochester Institute of Technology
jbe5115@rit.edu, mxleec@rit.edu

SUMMARY

FSMLock is a form of sequential logic locking that obfuscates FSM transition and output logic through classical encryption. Through the use of memory, FSMLock stores encrypted state information (Encrypted SET)), and as the state machine traverses, in-scope state machine information is decrypted and stored as Scoped SET. A diagram representing this process for a Mealy-type state machine is shown in Figure 1.

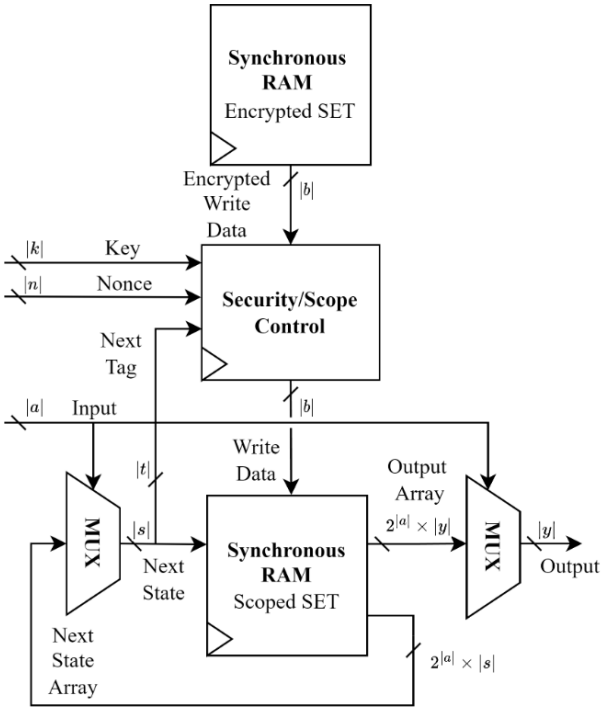


Fig. 1. FSMLock methodology for Mealy FSMs

To optimize the memory usage, the Finite State Machine Input Multiplexing (FSMIM) approach is integrated into FSMLock. The goal of FSMIM is to utilize the fact that not all inputs of an FSM are required in a certain state to transition to another state.

In Figure 1, several of the bit widths for each of the FSM parameters are shown, such as the input bit width $|a|$, state encoding bit width $|s|$, output bit width $|y|$, and cipher block bit width $|b|$. In addition, there are other FSM parameters not shown in Figure 1 that are specific to FSMIM, such as the effective input bit width $|ei|$ and the input selector bank (ISB) bit width $|isb|$. The in-scope and out-of-scope memory sizes are provided in Equations 1 and 2.

$$Size_{out} = 2^{|s|}(2^{|ei|}(|s| + |y| + |isb|)) \tag{1}$$

$$Size_{in} = 2^{|i|}(2^{|ei|}(|s| + |y| + |isb|)) \tag{2}$$

The FSMLock methodology follows a specific memory structure to organize FSM state entries. Within the state entries, there is a precise partitioning that organizes all possible state transitions (next states), outputs, and input selectors for a given state. This partitioning is shown in Figure 2.

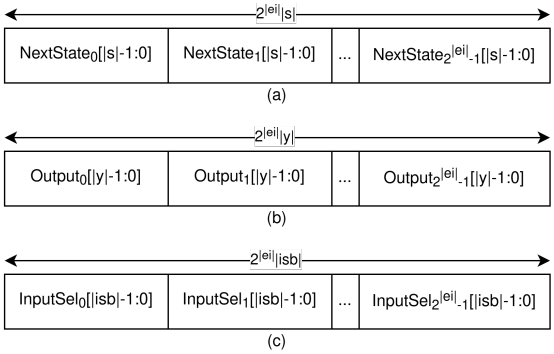


Fig. 2. FSMLock State Entry Partitioning for State Transitions (a), Outputs (b), and Input Selectors (c)

The objective of this work was to analyze the size, performance, and usability of the FSMLock methodology. Using the previous work & research done on the FSMLock, further analysis on the methodology was performed to derive a set of formulas/constraints on the parameters of a given FSM (input width, output width, state count, transition count) when given a desired memory size.