

FSMLock: Sequential Logic Locking Case Study

Jacob LaPietra, Michael Kurdziel

L3Harris Technologies

jacob.lapietra@L3Harris.com, Mike.Kurdziel@L3Harris.com

Marcin Łukowiak

Department of Computer Engineering

Rochester Institute of Technology

mxleec@rit.edu

SUMMARY

FSMLock is a sequential logic locking technique that has been proposed for protection of intellectual property (IP) of finite state machine (FSM) circuits. FSMLock is applied to a sequential circuit by abstracting a flattened version of its distinct state entry table (SET) into a binary data file, which can then be encrypted and stored in non-volatile memory. The encrypted flattened state entry table (FSET) representation is read in partitions such that, at run-time, only a subset of the sequential logic is in scope. A high level architecture of the FSMLock is illustrated in Figure 1. While this technique provides security advantages over other sequential logic locking techniques, one major drawback it brings is the large amount of memory required for storing data of all states, transitions, and outputs. Finite state machines with input multiplexing (FSMIM) is an optimization methodology and tool set that was proposed for efficient mapping of FSMs into memory. This is primarily achieved by reducing the number of effective inputs to the FSM, and converting it from Moore's to Mealy's machine for minimizing the number of states. This paper discusses our work on integrating these two techniques in a practical case study of converting an existing state machine into an implementation using FSMLock with input multiplexing.

The state machine used in this case study has 55 states, 71 transitions, 45 inputs and 63 outputs, and its behavior was originally modeled in Verilog HDL. We analyzed configurations with one partition, where the entire state machine would fit into Scoped FSET memory, with two partitions, and with four partitions. The memory based architecture together with the original model were simulated in a VHDL test bench in order to ensure that functionality of the original FSM was preserved. The hardware tests were conducted on a Digilent Nexys4 development board with an Artix 7 100T FPGA

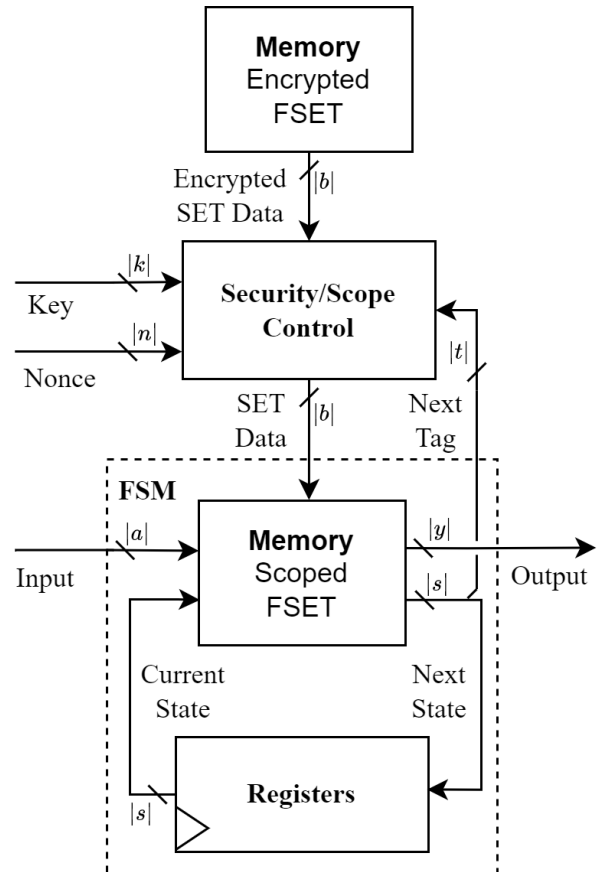


Fig. 1. The FSMLock primitive.