A Program That Computes Optimal and Secure Physical Unclonable Function Implementations of Integrated Circuits

ANTHONY LOPEZ, University of California – San Diego
Miodrag Potkonjak, University of California – Los Angeles

Contact Information

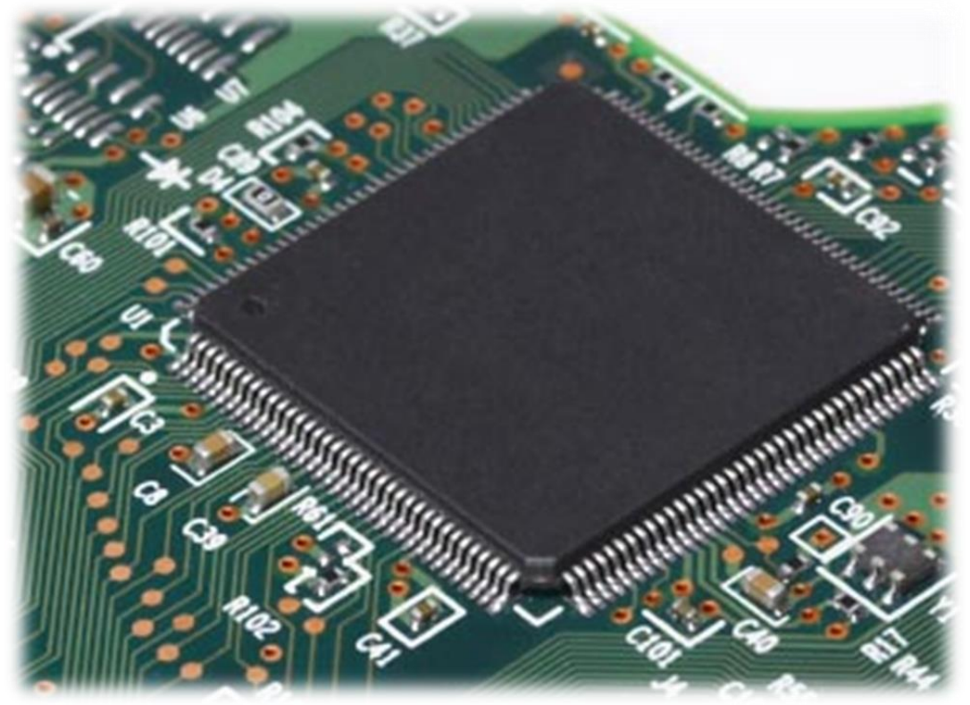
Introduction

The International Chamber of Commerce estimated the cost due to counterfeit and piracy in 2008 to be **777 billion dollars every year**.[1]

Physical Unclonable Functions (PUFs) promise cheap, efficient, and secure protection **against integrated circuit (IC) counterfeiting**.[2, 3]

However, **complexity and overhead** in terms of speed and area exist in PUF implementations.

This project aims to create a program that computes and evaluates a **theoretically minimal PUF implementation** for a given integrated circuit.



A Finished Version of an Integrated Circuit

Background

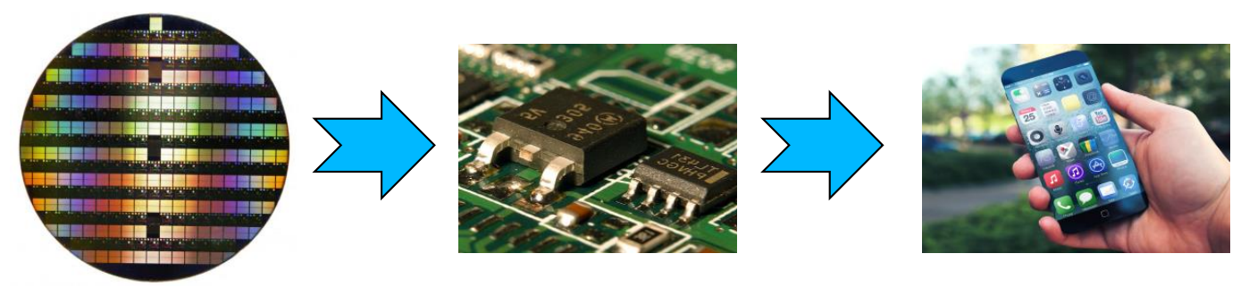
Process Variation (PV) as a result of the imperfectness of the manufacturing process in the physical level characteristics (such as effective transistor channel length and transistor threshold voltage) significantly affect delays and power in gates. [5]

Physical Unclonable Function(PUF)[4] is a function that must be:

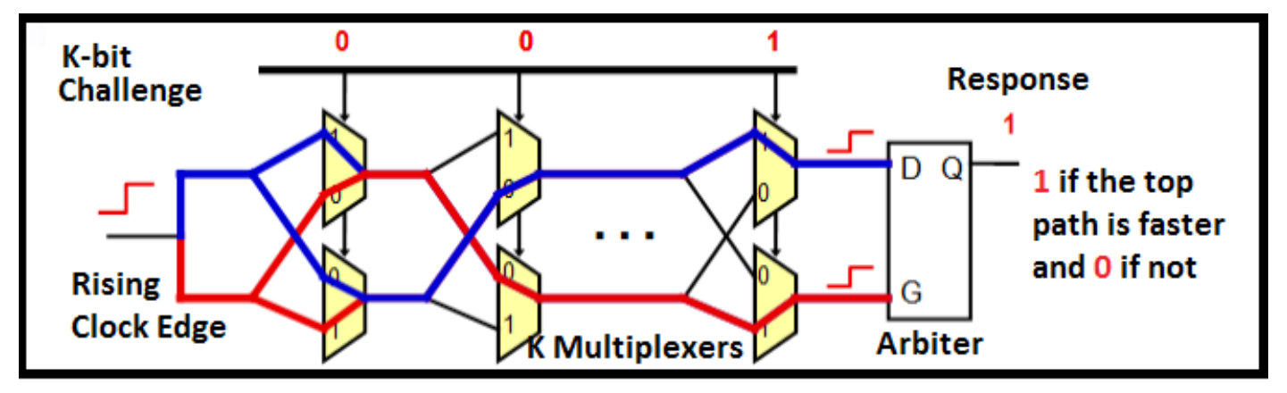
1. Fast
2. Unpredictable
3. Tamper resistant

An **Arbiter PUF** uses a function based on the delay differences as a result of random PV to map challenges to responses.[4]

A **PUF Implementation** is a solution to an integrated circuit which performs like the original circuit but with a different design.



Simplified Overview of Manufacturing Process



Arbiter PUF with K Challenge Inputs

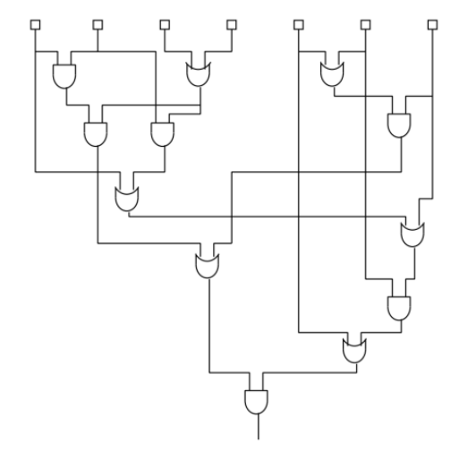
A **directed acyclic Boolean network[3]** is a graph with no cycles where each node represents a logic gate and each incoming edge represents an input.

Approach and Related Material

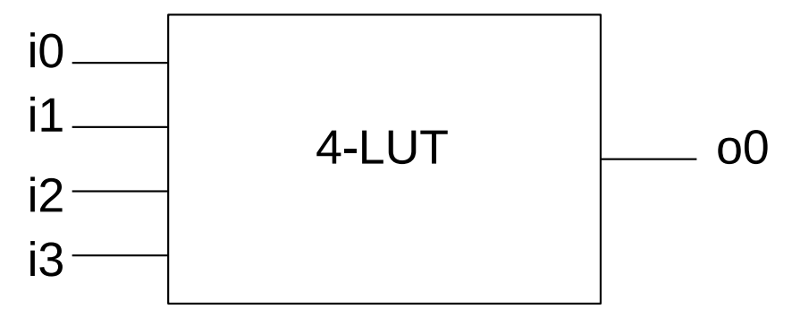
We relate the **problem at hand to minimizing the critical (longest) path and number of gates of a PUF implementation**. We therefore adapt an an algorithm called HeteroMap.

HeteroMap[3] is an algorithm that creates minimal delay and area mapping for a Look-Up Table(LUT) implementation of a directed acyclic Boolean network.

A K-**Look-Up Table (K-LUT)** can perform the function of any K (or less than K)–variable Boolean function. We **treat a Look-Up Table similar to an arbiter PUF** for this reason.

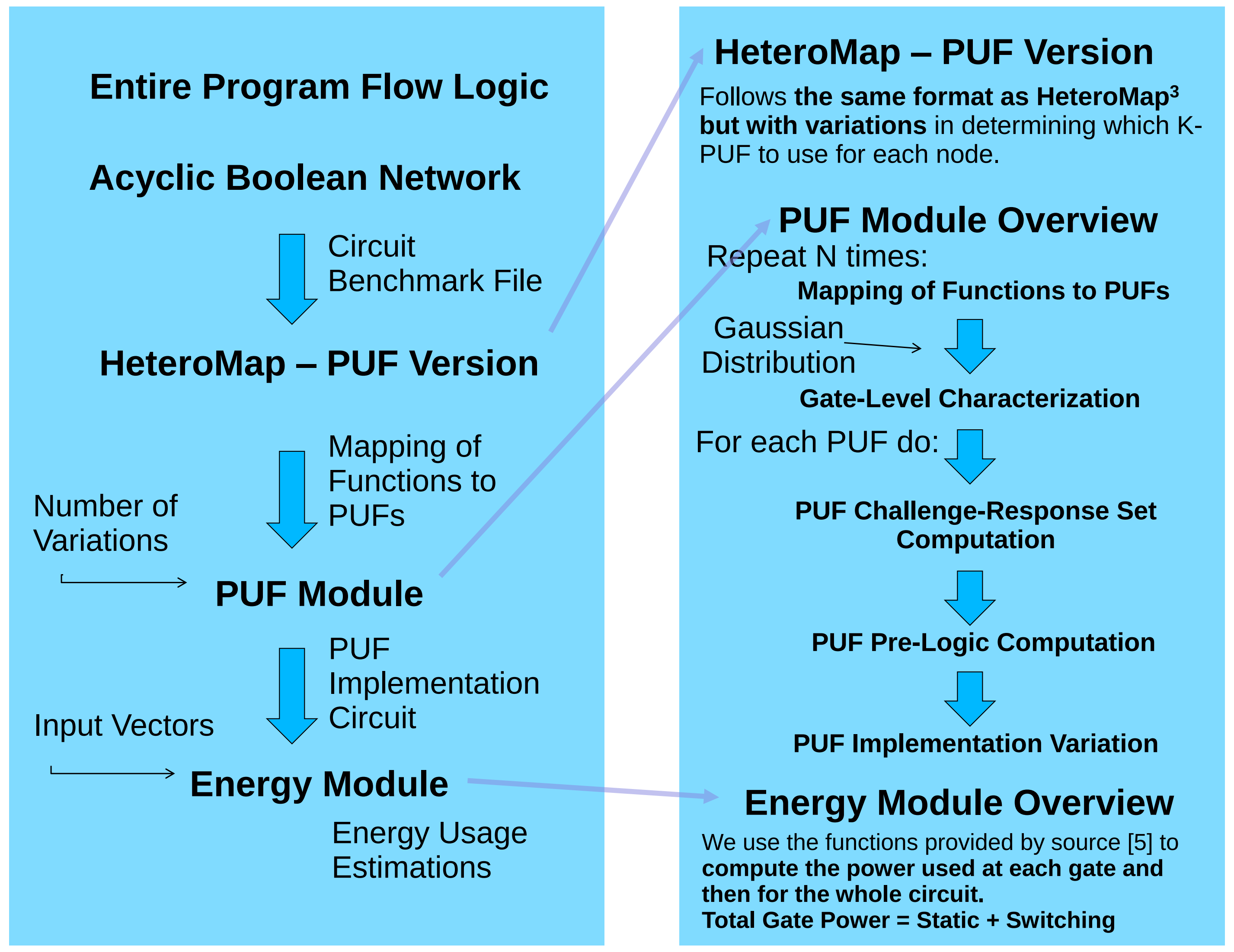


A Boolean Network



A 4-LUT Where K = 4

Techniques and Flow Logic



Entire Program Flow Logic

Acyclic Boolean Network

Circuit Benchmark File

HeteroMap – PUF Version

Mapping of Functions to PUFs

Number of Variations

PUF Module

PUF Implementation Circuit

Input Vectors

Energy Module

Energy Usage Estimations

HeteroMap – PUF Version

Follows **the same format as HeteroMap[3] but with variations** in determining which K-PUF to use for each node.

PUF Module Overview

Repeat N times:

Mapping of Functions to PUFs

Gaussian Distribution

Gate-Level Characterization

For each PUF do:

PUF Challenge-Response Set Computation

PUF Pre-Logic Computation

PUF Implementation Variation

Energy Module Overview

We use the functions provided by source [5] to **compute the power used at each gate and then for the whole circuit.**

Total Gate Power = Static + Switching

Experiments

- We computed different function-to-PUF mappings by providing HeteroMap – PUF version with **different values for K that the implementation could use**.
- A **delay of a K-PUF is calculated as (K-mink)*.25.** For our purposes we consider **minK to be 5.**
- We compute the number of PUFs used for each test run first by using **minK = 5 and maxK = circuit input size and then by using mink = 5 and maxK = approx. Num of Inputs/2**
- The **results of the PUF and energy modules** in the last two **columns demonstrate the final size and power dissipation** of each graph.

PUF Implementations of Benchmark Circuits

ISCAS 85 Circuit	Critical Path Delay	Herero Map Params: MinK - MaxK	Circuit Size (#gates)	PUF Implementation Size (#PUFs)	PUF Implementation Size (#gates) (pre-PUFModule, EnergyModule)	Final PUF Implementation Size (#gates) (post-PUFModule, EnergyModule)	Power Dissipation (In watts) (of one PUF implementation)
C17	1.00	5-11	11	2	8	12	$4.05961*10^{-18}$
C499	4.00	5-11	202	64	784	795	$7.45094*10^{-16}$
C880	7.50	5-11	383	105	592	655	$5.85386*10^{-16}$
C1355	4.00	5-11	546	64	1128	795	$7.45094*10^{-16}$
C6288	18.00	5-11	2461	366	6137	7277	$7.23626*10^{-15}$
C17	1.00	5-5	11	2	8	12	$4.05961*10^{-18}$
C499	4.00	5-20	202	64	784	795	$7.45094*10^{-16}$
C880	7.50	5-30	383	105	592	655	$5.85386*10^{-16}$
C1355	4.00	5-20	546	64	1128	795	$7.45094*10^{-16}$
C6288	17.50	5-14	2461	301	8481	29050	$2.90227*10^{-14}$

Table demonstrates the impact of using the range of K on the final PUF implementation size.

Preliminary Results and Discussion

- The **resulting energy values indicate that circuit is protected against energy reading side channel attacks**.
- Using **HeteroMap minimizes critical path** in final PUF implementation circuit.
- Total PUF implementation size to original circuit size ratio is still large.
- For large minK and maxK values, the PUF Module and Energy Module programs will not be able to handle computing large pre-logic functions.

Future Work

- To **improve** pre-logic minimization for each individual PUF.
- To **expand** the PUF Module and Energy Modules by using many variations for a single circuit.
- **To optimize** the program in terms of lines of **code and speed**.
- To **test** results of this project to hardware and determining if the product is successful.

Summary

- **Successfully converts a given digital logic circuit into a single variation of an almost minimal PUF implementation circuit protected from energy reading attacks.**
- Future work will focus on the PUF and Energy Modules and the physical simulation of this program by using a Field Programmable Gate Array.

References

1.Chardonnal D (2011) Impacts of counterfeiting and piracy to reach US$1.7 trillion by 2015.
2.Morozov, S., Maiti, A., Schaumont, P.: A Comparative Analysis of Delay Based PUF Implementations on FPGA. IACR ePrint tbd/2009 (submitted December 19, 2009).
3.C. Jason and X. Songjie. Delay-Optimal Technology Mapping for FPGAs with Heterogeneous LUTs. In *Proc. 35th ACM/IEEE Design Automation Conference*, 1998.
4.B. Gassend, "Silicon Physical Random Functions," ACM, 2002.
5.Sheng Wei, Farinaz Koushanfar, Miodrag Potkonjak: Integrated circuit digital rights management techniques using physical level characterization. Digital Rights Management Workshop 2011: 3-14.

Acknowledgements

Christopher Murphy, Amalia Castaneda, Devin Horton, Kim Tran

UC LEADS
UNIVERSITY OF CALIFORNIA
LEADERSHIP EXCELLENCE THROUGH ADVANCED DEGREES PROGRAM