WHITE PAPERS 7218

HOW PHYSICALLY UNCLONABLE FUNCTION (PUF) TECHNOLOGY PROTECTS EMBEDDED SYSTEMS

By: Kristopher Ardis

Abstract: Security experts have been excited about the promise of physically unclonable function (PUF) technology for many years. It wasn't until recently, however, that reliable, cost-effective ICs with integrated PUF technology became available on the market. What's driving all of the excitement over PUF? In this white paper, I'll demystify PUF and highlight how it benefits a variety of embedded systems.

Introduction

From security cameras to thermostats, appliances, and toys, our everyday electronics are getting smarter and connected. As such, they need to be protected from tampering, reverse engineering, and other malicious attacks. After all, by gaining entry into one poorly protected device, a hacker can find a way into the larger network, where, perhaps, even more sensitive data resides.

Cryptography with secret binary keys provides a measure of security, but only so long as the key remains safe. PUF technology delivers a more robust means to protect IoT applications and more because the key is not stored on the device. Plus, PUF technology is implemented in such a way that it defends against physical attacks.

Like humans with our unique fingerprints, chips also have their own unique fingerprints, created during their manufacturing process. PUF exploits minute differences in silicon that appear from chip to chip (even two chips side-by-side on the same wafer) to create a binary value. Bring together enough PUF cells, and you can create arbitrary-length numbers with good properties of randomly generated numbers. You can then exploit the chip-to-chip variances in such a way that those arbitrary-length numbers are practically unique no matter how many chips are manufactured.

How is this possible? Once you examine the physical design of a chip in great detail, you'll recognize that each piece of silicon manufactured has minute differences—despite highly accurate manufacturing processes. PUF technology amplifies these minor differences to create a unique number on each chip, kind of like how DNA is unique to each person.

What makes PUF technology valuable is its use for secret keys. Consider this analogy: PUF is like an insanely long password you have to enter to get into your house, ensuring that only trusted people can access the house. In this paper, I'll provide a more detailed look at the role of PUF in key protection.

Why Key Protection Is Essential in Our Digital World

Securing anything in the digital world calls for some form of encryption. The right cryptographic tools can help you implement:

- Confidentiality, protecting communications from point A to point B
- Integrity to detect whether messages received have been tampered with
- Authentication to prove that a device belongs to a particular group or network

Let's consider each of these areas. Confidentiality is what most people think about when they think about encryption. Encryption scrambles a message, so someone listening to the communication cannot intercept or understand it (**Figure 1**).

Figure 1. Encryption from point A to point B.

Integrity relates to potential message tampering. To maintain integrity, you could use a cryptographic algorithm called a hash, which takes an arbitrary-length data stream as input and outputs a constant-length number. A hash should be infeasible to reverse. A hash run over an input message AND a secret key, and verified by the recipient, can prove that the message was not tampered with by a third party.

Authentication techniques are used to prove that someone or something is part of your group and can be trusted. When a new device wants to join a group, the servers of that group must "challenge" it to see if it really belongs. A one-time random number (called a challenge or a nonce) is sent to the new device, and

again cryptographic algorithms are executed on the random number and the secret key to produce an answer. If the answer checks out at the server, the new device is admitted to the group.

Note that in all three of these scenarios, the secret key plays a pivotal role: an attacker who knows the secret key could impersonate a valid device, create fraudulent messages or tamper with legitimate messages, or listen to sensitive communications at will. If the data being transferred is deemed valuable or, similarly, if joining a certain group is considered worthwhile, attackers may be willing to spend significant sums of time and money to figure out your secret keys. Who is the attacker? It could be a criminal trying to take control of financial transaction equipment, competitors trying to create devices that can operate within your infrastructure or reverse-engineer your work to create clones, or even governments trying to illicitly access communications or break critical industrial equipment.

In summary, secret keys really are the most important asset in any security scheme. In silicon, the secret keys have to *be* somewhere, typically in some kind of memory cell. Let’s discuss secret key storage and the benefits and drawbacks of different methods.

How Secret Keys Are Stored Today

Some systems store secret encryption keys in external memories, such as non-volatile resources like NOR/NAND flashes or special external memory chips like battery-backed SRAMs. When the main system microcontroller or microprocessor needs to use that secret key, it must read it over a memory bus, where that key is transmitted in the clear. To protect that key, some systems implement extensive and expensive physical security methods to make it difficult for an attacker to monitor those clear-text transmissions. For example, they might hide the memory bus in the middle layers of a PCB, implement sensor circuitry around the sensitive area to detect attacks, or fill the empty space of the device with a plastic filler that is exceedingly difficult to remove. These kinds of approaches are, however, expensive and can often be defeated by patient attackers.

A more effective solution is to store secret keys in the same place they will be used. In embedded systems, this commonly means storing those keys in a non-volatile memory. They are programmed into flash or EEPROM, or possibly even manufactured into a ROM. While the secret keys can then remain on chip, there are still physical attacks that can access those keys. For example, the plastic package around a piece of silicon can be decapsulated to allow microprobing of the memory busses (**Figure 2**).

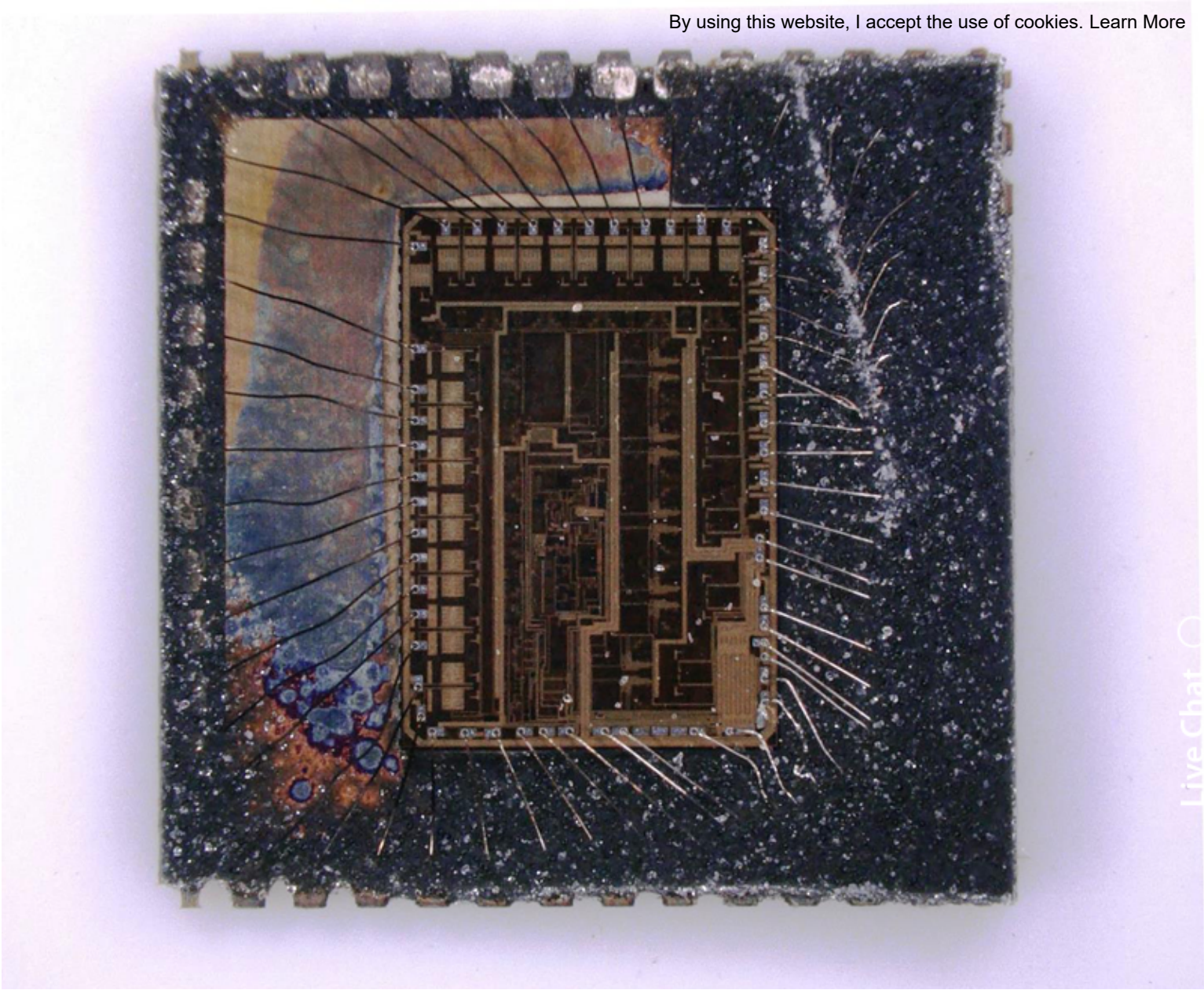

Figure 2. Decapsulated chip.

More physical security can be implemented on the chip itself, perhaps by adding top layers to the silicon above the memories so they can be accessed directly by a microprobe. This makes it more challenging to conduct an attack, but it's still possible to carefully remove those layers of silicon to access the deeply embedded charges in those flash or EEPROM memories to extract secret key information.

The challenge with flash, EEPROM, and ROM technology is that when power is removed from the system, the secret key contents remain stored in those memories, and there is no power available to erase those memories in the event an attack is detected. Battery-backed SRAM, especially when combined with tamper-detection sensors, presents a better approach. In systems with this kind of technology, very low-power sensors run off a small battery to detect various physical attacks, erasing the small battery-backed SRAM storing the secret keys if an attack is detected. If an attacker removes the battery from the system to disable

the sensors, this act also removes power from the SRAM and the secret key information is lost. While there are still some much more difficult attacks that can look at unpowered SRAM cells and try to determine a memory “imprint,” this is the preferred technology used in many government and financial applications today. However, in addition to being susceptible to the memory imprint inspection, there is one big drawback to this technology: the battery. It adds cost, size, and even environmental concerns to the design.

Why PUF Technology Is a Better Key Storage Method

A good PUF implementation addresses all of the drawbacks of conventional key storage:

- Under normal operating conditions, PUF circuitry is inherently non-volatile, requiring no battery or other permanent power source. While the number read from any IC’s PUF circuitry should have good random characteristics (in that each bit cannot be used to predict the value of any other bit in the PUF bit sequence), the PUF in the IC will reliably produce the same result every time.
- PUF circuitry should be resistant to physical inspection. By amplifying the minute imperfections in the physical silicon itself, PUF circuits are inherently highly sensitive. Attempts to physically probe the PUF implementation will dramatically change the characteristics of that PUF circuit, and result in a different number being produced.
- The key from PUF can be generated only when required for a cryptographic operation and can be instantaneously erased thereafter

This is a powerful combination: it provides the bill-of-materials (BOM) and environmental benefits of a non-volatile memory, with the security of a tamper-reactive SRAM. In other words, the secret is always there in the circuit, but you can never look at it. It has some similarities to Heisenberg’s uncertainty principle: you can know that the atom is there, but the mere action of observing it changes its behavior.

The implementation of a good PUF technology isn’t enough to assure key security: once that secret key is in use, the cryptographic implementation must make sure to be resistant to side-channel attacks. But PUF does help to make sure the embedded device is not the weak point in a system for attackers to focus their efforts.

Summary

The technology used in malicious attacks continues to advance while becoming less expensive and also more accessible. Today’s conventional security technology will likely be more susceptible to attack tomorrow. While each design and each situation will be different and call for different solutions, PUF technology is the best key storage technology available. As the most advanced key storage technology available today, PUF technology can provide your applications with a longer lifetime at a small cost—before your designs are subject to security threats.

For More Information

To learn more about PUF technology, visit the ChipDNA PUF Technology page, where you can find other white papers, an on-demand webinar, and additional resources. This white paper was adapted from an article which originally appeared on Embedded Computing Design on February 1, 2019.

Related Parts		
MAX32520	ChipDNA Secure Arm Cortex M4 Microcontroller	Free Sample
DS28E50	DeepCover Secure SHA-3 Authenticator with ChipDNA PUF Protection	Free Sample
DS28E38	DeepCover® Secure ECDSA Authenticator with ChipDNA PUF Protection	Free Sample
DS28E39	DeepCover Secure ECDSA Bidirectional Authenticator with ChipDNA PUF Protection	Free Sample
DS28C39	DeepCover Secure ECDSA Bidirectional Authenticator with ChipDNA PUF Protection	Free Sample
DS28C50	DeepCover I^2C Secure SHA-3 Authenticator with ChipDNA PUF Protection	Free Sample
DS28S60	DeepCover Cryptographic Coprocessor with ChipDNA	Free Sample
DS2477	DeepCover Secure SHA-3 Coprocessor with ChipDNA PUF Protection	Free Sample

Next Steps	
EE-Mail	Subscribe to EE-Mail and receive automatic notice of new documents in your areas of interest.

APP 7218: 22 Jul, 2020
WHITE PAPERS 7218, AN7218, AN 7218, APP7218, Appnote7218, Appnote 7218