Taking eFPGA Security to the Next Level

By Ralph Grundler, Flex Logix Technologies, Inc. and Vincent van der Leest, Intrinsic ID B.V.

Many markets – 5G, networking, cloud storage, defense, smart home, automotive, and others – are looking to embedded FPGAs (eFPGA) to save power and reduce cost. By removing the high-speed SERDES and other unnecessary I/Os or unused peripherals, the customer can save power and reduce latency. With FPGA embedded, the end product can take advantage of these benefits and still be reconfigurable in the field, saving time and money.

Reconfigurability has many uses, but in the past, this meant greater area and came at an additional cost that was difficult to justify except where it was a requirement. Smaller technology nodes and the increasing cost of taping out (or retaping out), have made FPGA technology from Flex Logix, EFLX® eFPGA, both simple to integrate using less area than ever before and easy to justify from a cost perspective.

Flex Logix has become the number one eFPGA vendor because of strong adoption based on several patented technologies that reduce the size of the eFPGA. One of these technologies is the Boundless Radix Interconnect technology, which can reduce the interconnect area by forty-five percent (Fig 1). This results in an area efficient solution, similar in density as custom designed FPGA chips while offering higher utilization and using just the lower layers in a metal stack making it compatible with most metal stacks. As more and more SOCs are integrating FPGAs, the question of security is being raised, and it's an important one.

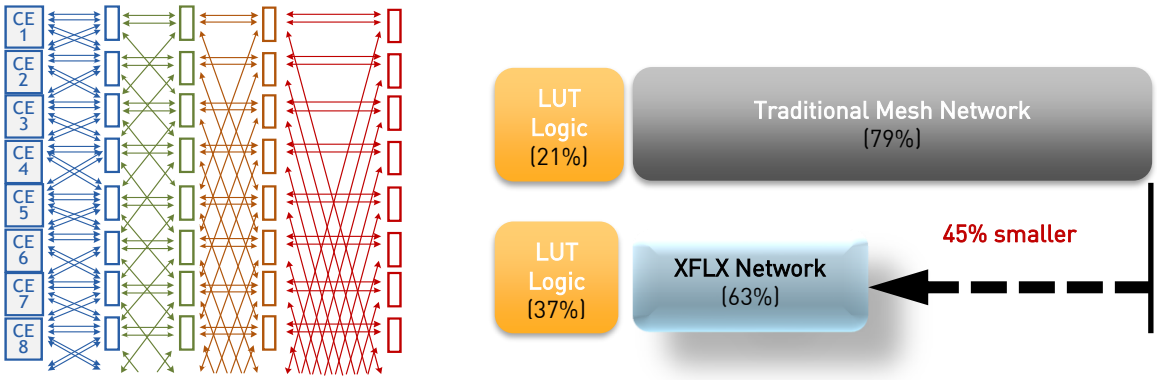

Fig 1. Although LUT logic is slightly larger the Boundless Radix Interconnect (XFLX® Network) is 45% smaller and results in higher utilization

Security is an important topic for every SOC, but it's especially salient in the context of high-risk assets included in the eFPGA for obfuscation. Whether the device is used in defense systems or in cars driving around town, encryption is important so the device remains secure and can't be modified maliciously, whether through physical attacks or remote hacking. There are several different established ways to secure eFPGA content, each with its own tradeoffs. Now there is a new and better way to take encryption of the eFPGA content to the next level.

What if you could encrypt your eFPGA configuration data with a device-unique key that is never stored on the device, that cannot be copied from one device to the next, and that is not known to anyone (not even you)? Now you can, by using the secure and patented SRAM PUF (or Physical Unclonable Function) technology from Intrinsic ID. The SRAM PUF creates a device-unique fingerprint (Fig 2) from which a cryptographic root key is extracted. A key derived from this device-unique root key is used to encrypt and authenticate the bitstream of the eFPGA. If the device is attacked or found in the field, the bitstream of the eFPGA cannot be altered, read, or copied to another device, because it is protected by a key that is never stored and therefore is invisible and unclonable for the attacker.

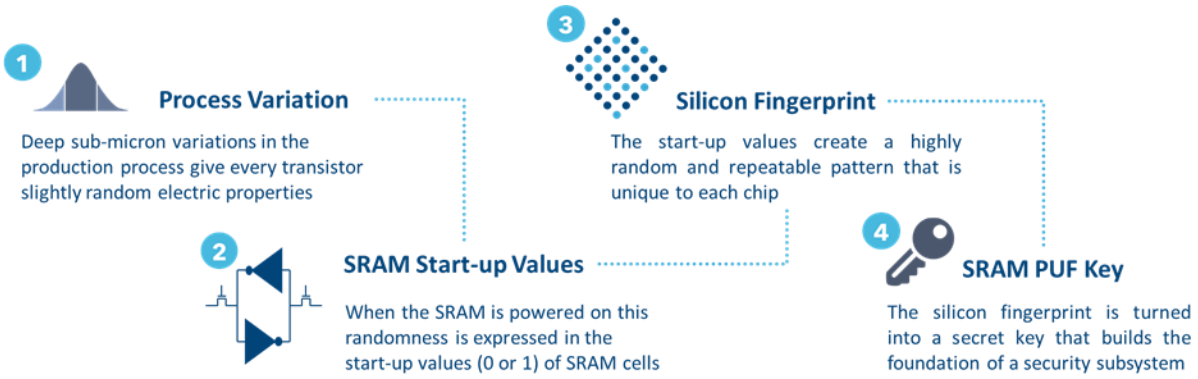

Fig 2. SRAM fingerprint turned into a strong cryptographic key

The main benefits of using the Intrinsic ID SRAM PUF technology over storing a key in non-volatile memory are:

- **High Security**: No key material programmed into device and no key present when it is not in use.
- **High Flexibility**: Key generation at any time and place in the supply chain without external provisioning.
- **Low Cost**: No dedicated security hardware required to protect the key, as it is never stored.
- **Highly Scalable**: It employs only standard logic, scaling effortlessly with shrinking technology nodes.

From the SOC hardware engineering point of view, the process for encryption with SRAM PUF-based keys is no different than with any other type of encryption. It only requires instantiating the SRAM PUF through the Intrinsic ID hardware IP called QuiddiKey®. The standard implementation looks like the diagram below (Fig 3).

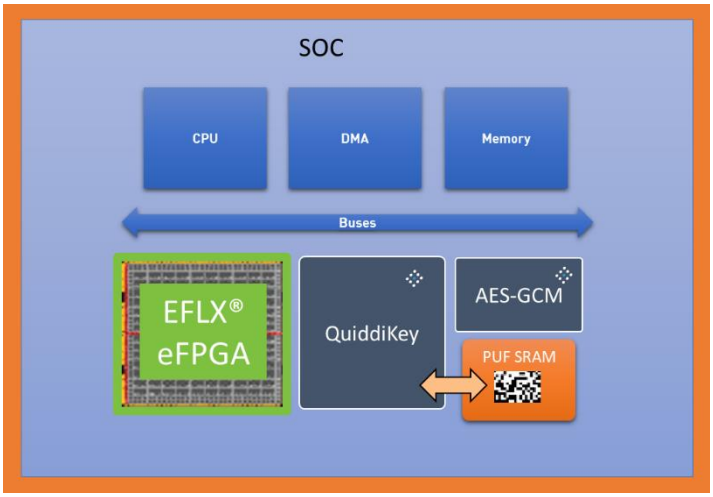

Fig 3. Diagram of QuiddiKey with PUF SRAM implemented in a system with EFLX eFPGA

The system would power up just like any other eFPGA, but the AES-GCM will fetch the key from QuiddiKey and decrypt the eFPGA configuration data before programming the eFPGA in the SOC. The configuration data can be stored in any non-volatile memory since it is protected by the key from the SRAM PUF. Not only will your supply chain be secure by using obfuscation, but the bitstream of the eFPGA can also no longer be altered, read, or copied to another device.

Contact us now to help you take the security of your configuration data to the next level by combining eFPGA and SRAM PUF technology.

Learn more about the Flex Logix eFPGA solutions at: https://flex-logix.com/efpga/
Learn more about the Intrinsic ID PUF technology at: https://www.intrinsic-id.com/physical-unclonable-function/