Protecting your FPGA Intellectual Property

Mark Frost

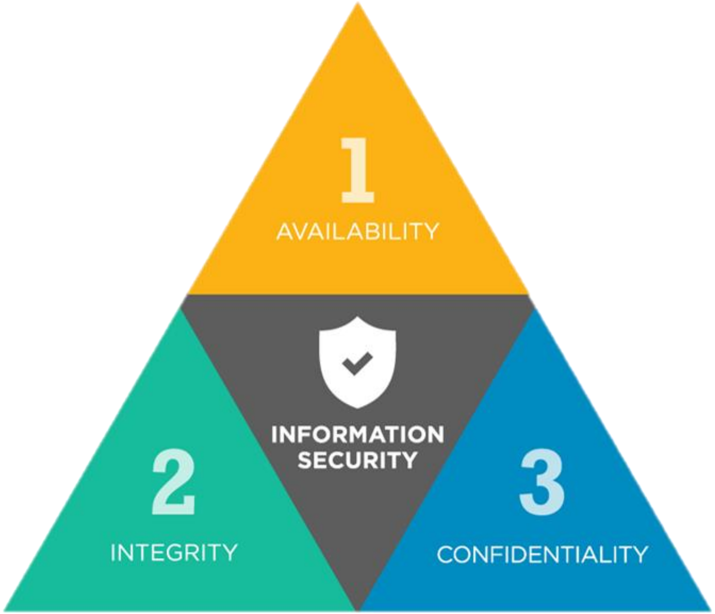
Key Takeaways

FPGA IP Theft is a real concern

You can easily do something about it

Why do you need Security in your FPGA?

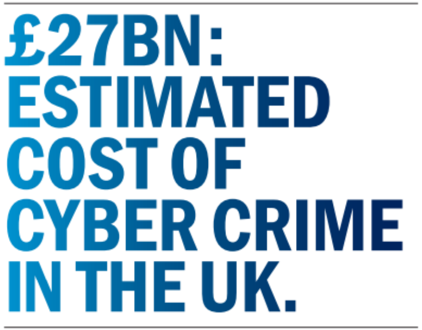
- Increased reliance on third party manufacturers [OEMs/ODMs] leads to concerns over IP theft and other security vulnerabilities
- Increased remote access in the interconnected world is a double-edged sword: increased cyber threats and attacks
- More and more Security researchers and hackers are trying to expose device vulnerabilities
- Other unscrupulous parties are also trying to clone/modify or compromise products
- Corporate Mandates are increasingly focusing on device security



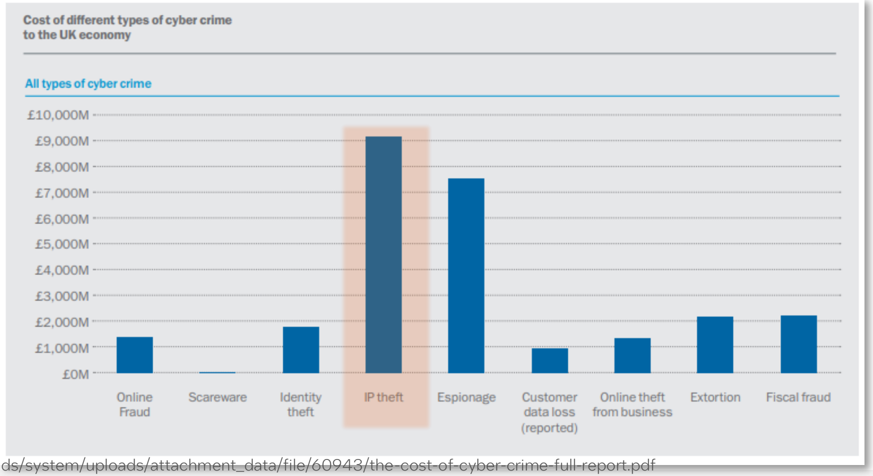
NIST FIPS140-3

A poor security policy costs time, money & reputation

- Cyber security issues on the rise
- Having a robust security policy is increasingly relevant.
- Many FPGA-based solutions still don’t implement any kind of security – these solutions are “protected and secure” until they suddenly are not.



£27BN: ESTIMATED COST OF CYBER CRIME IN THE UK.

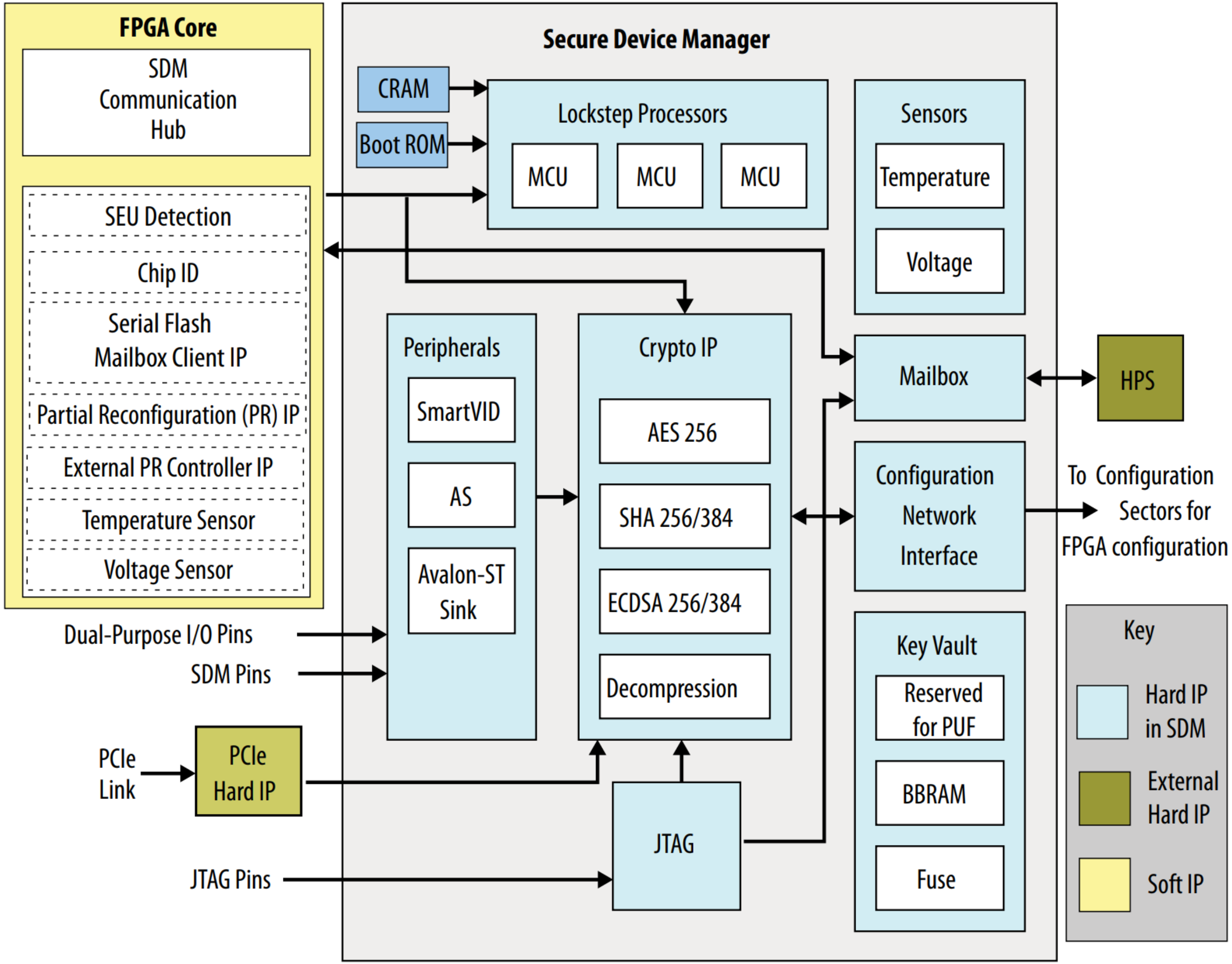


Cost of different types of cyber crime to the UK economy

https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/60943/the-cost-of-cyber-crime-full-report.pdf

Protecting Your IP : Agilex FPGAs

Feature	Functional Description
Authentication	Only authorised bitstreams can run on silicon
Bitstream Encryption / Side Channel Mitigation	Protects IP from cloning or reverse-engineering, prevent leak of configuration data during configuration process
Black Key Provisioning	Program AES key securely in a non-secure manufacturing environment
Physically Unclonable Function (PUF)	Device-unique, undiscoverable key, for key wrapping and remote authentication
HPS Debug Certificate	Secures HPS JTAG debug by requiring a debug certificate
Physical Anti-Tamper	Prevent asset loss under intrusive/non-intrusive attack
Vendor Authorized Boot	Provide vendors control over what software is authorized for use from ROM to OS kernel launch
Bit Stream and Platform Attestation	The device provides measurement of configured bitstream for external verification
Crypto Services	Use of SDM cryptographic engines via Mailbox



Secure Device Manager

Protecting Your IP : At least do this....

Enable Authentication

Enable Bitstream Encryption

Authentication

INTEGRITY

- Has the asset been altered since creation?
  - Configuration bitstream
  - SDM commands
  - HPS or Nios Software
  - Customer Data

ORIGIN

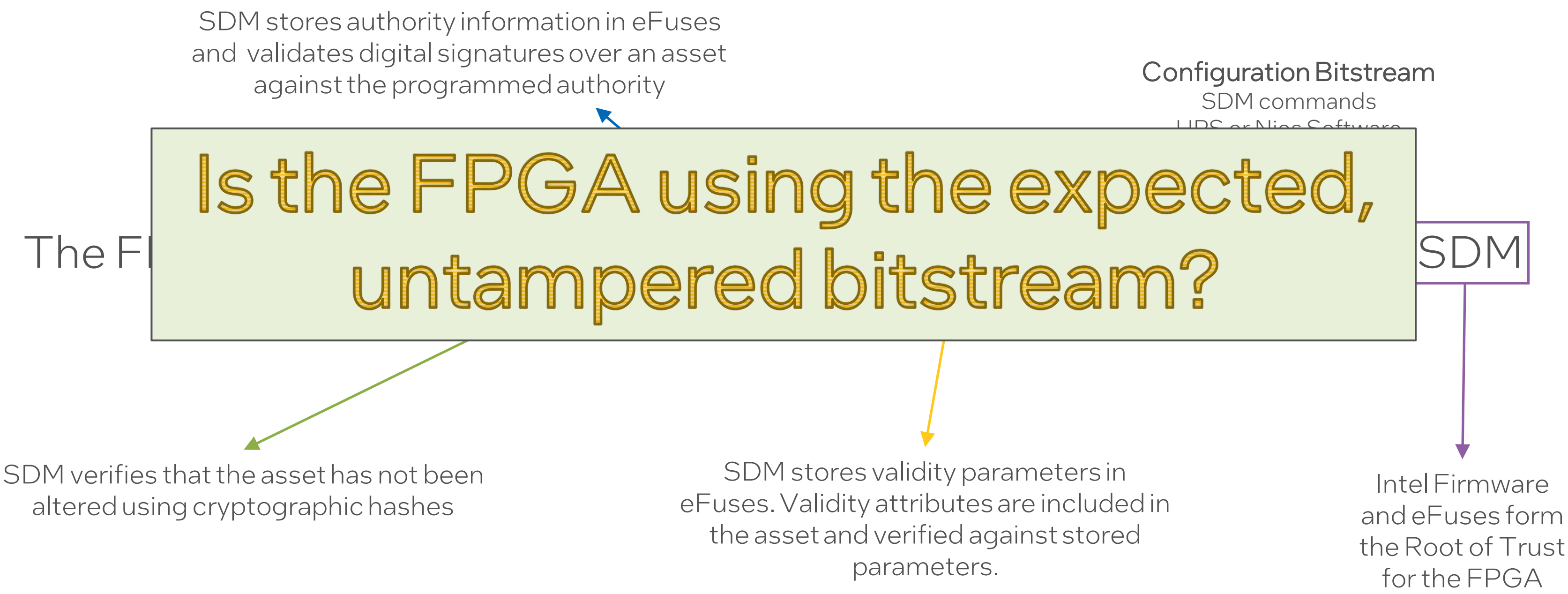
- What authority does the device trust?
- Can the asset be traced to that authority?

VALIDITY

- Does the asset meet the specified parameters of validity?

Authentication: The FPGA checks the integrity, origin and validity of the configuration bitstream

Device Security : Authentication Main Concepts



SDM stores authority information in eFuses and validates digital signatures over an asset against the programmed authority

Configuration Bitstream
SDM commands
HPS or Nios Software

The FI

SDM

Is the FPGA using the expected, untampered bitstream?

SDM verifies that the asset has not been altered using cryptographic hashes

SDM stores validity parameters in eFuses. Validity attributes are included in the asset and verified against stored parameters.

Intel Firmware and eFuses form the Root of Trust for the FPGA

Device Security : Bitstream Encryption

The security objective of configuration bitstream encryption is to protect the confidentiality of owner data in a configuration bitstream by scrambling its contents in a way that makes it infeasible for an adversary to extract any protected data from the encrypted configuration bitstream or the device, using standardized cryptographic encryption algorithms.

Protect confidentiality of design data and IP contained in bitstream

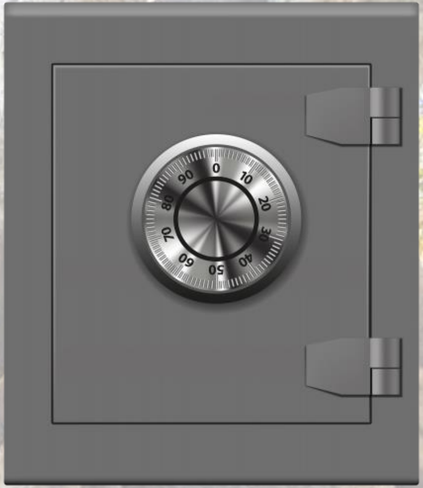
Device Security : Bitstream Encryption Usage

- Symmetric encrypt/decrypt used : AES256 (CTR/GCM)
- Device authentication must be enabled first! [Root key hash]
- Provision encryption key to FPGA
  - Often done during manufacturing process – plaintext or black key
  - Persistent storage in eFuses or BBRAM. Virtual flow also available
- Encrypt the configuration bitstream using Quartus
- SDM uses stored encryption key to decrypt the bitstream during the FPGA configuration process
- Bitstream structure and key usage is complex .Entire process limits any single key's exposure to any side channel leakage of encryption keys
  - AES Update Mode and Scrambling. Additional side channel countermeasures

Device Security : Attestation

TRUST

- Secure Device Manager
  - Updateable Firmware
- Bitstream Authentication
  - Vendor Authorized Boot
  - Secure Debug Authorization
- Bitstream Encryption
- Cryptographic Services



VERIFY

- Did the FPGA load the expected bitstream?
- Are the expected security settings programmed?
- Is this the expected FPGA?
- Is this an authentic FPGA?

Attestation: The FPGA provides signed measurements of configuration to an external verifier

Device Security : Crypto Services

- SDM has cryptographic engines to provide configuration security
- After device is configured, engines may be accessed via the SDM mailbox
- Services provided:
  - SHA2- HMAC digest request & verify
  - ECDSA signature request & verify
  - ECDSA get public key
  - ECDH
  - AES encrypt/decrypt
  - RNG

Next Steps?

Implementation Guides

- Security Methodology Guide (RDC link)
- Agilex 7/5 Device Security User Guides
- How To Tutorials documentation / video

FPGA Security

Authentication and bitstream encryption tutorial

Joe Finnivan

Authentication & Encryption Tutorial

Security : Why You Should Care
Security : Authentication/Encryption