ROM-based automotive bootcode compliant with functional safety and cybersecurity standards

Bipul Halder, Neha Sharma, Davide Silvio Fiorese, Anil Kumar Dwivedi

STMicroelectronics

62 THE CHIPS TO SYSTEMS CONFERENCE

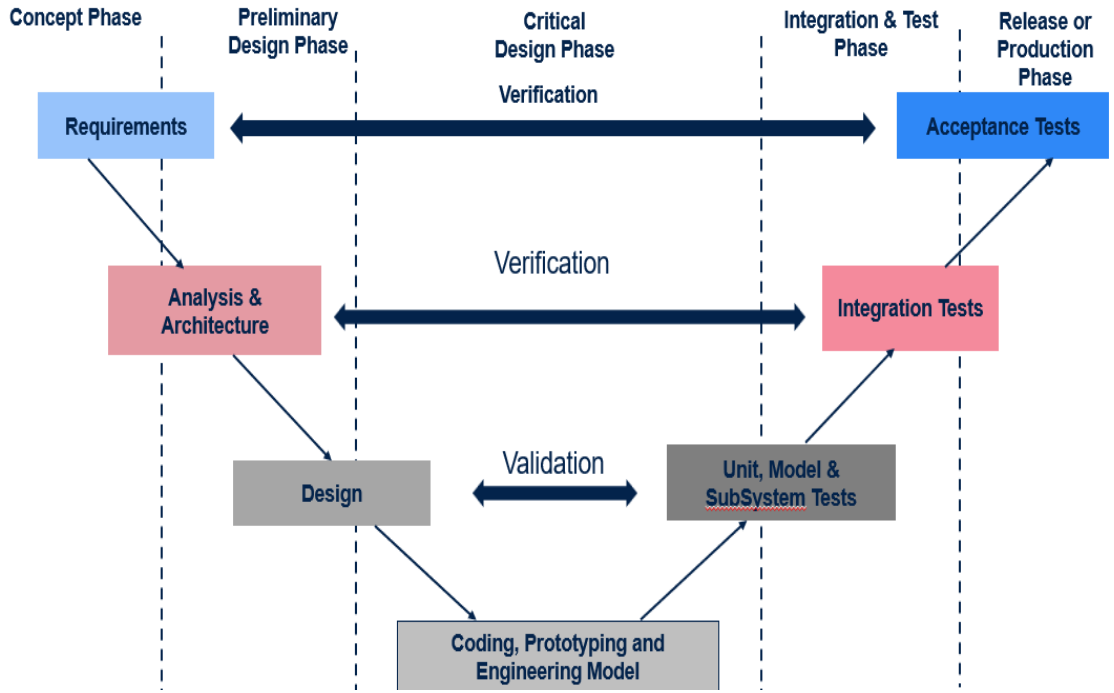
SPONSORED BY CEDA IEEE Council on Electronic Design Automation sigda acm

Motivation

Ensuring functional safety	Critical systems initialization Compliance with standards
Enhancing cybersecurity	Secure boot process Threat mitigation
Facilitating future updates	Over-the-air(OTA) updates Scalability
Reducing development costs	Early issue detection Efficient development

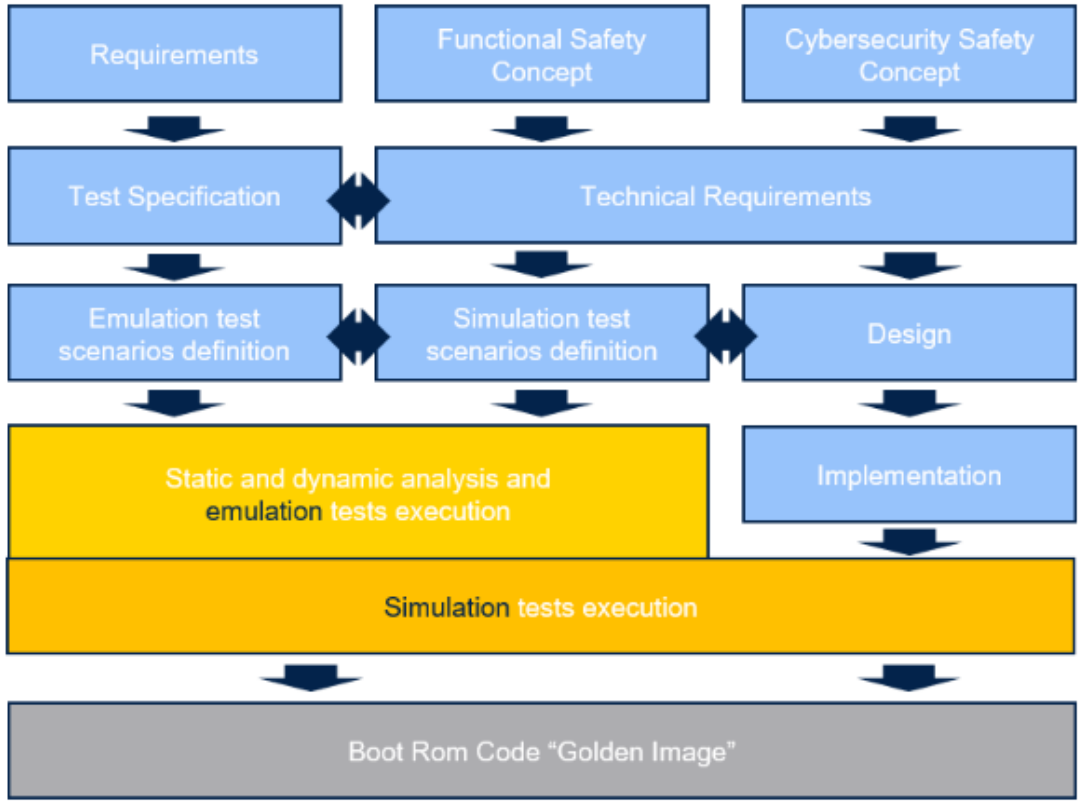
Learnings

- Basic understanding of booting sequence
- Importance of BootLoader
- Integration of safety and security
- Scope & challenges of BootLoader verification
- Multi-folded verification methodology in pre-silicon
  - Unit test
  - Emulation
  - Simulation



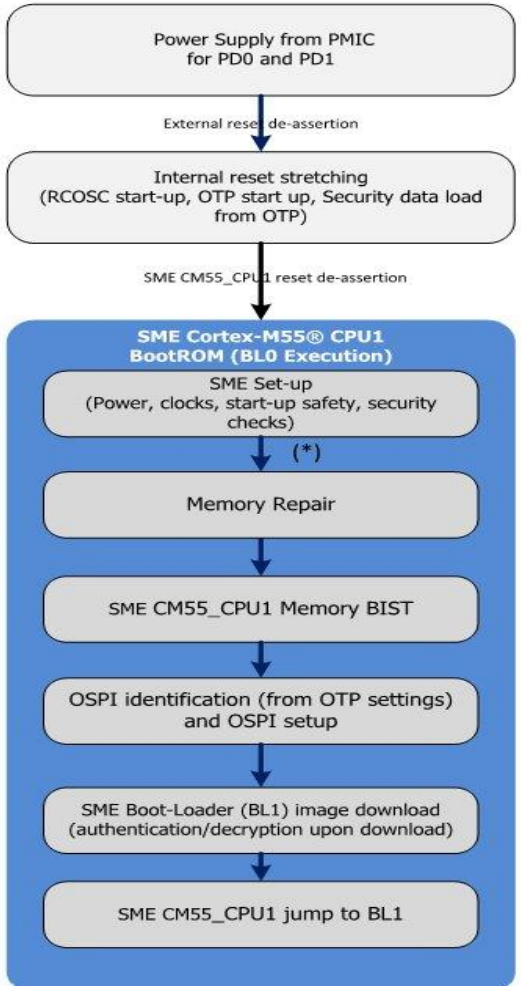
Importance of automotive BootROM

- **System initialization**
  - Hardware initialization
  - Boot sequence
- **Security**
  - Secure boot
  - Cryptographic verification
  - Root of trust
- **Reliability & safety**
  - Fault detection & recovery
  - Watchdog timers
- **Compliance with standards**
  - Functional safety – ISO26262
  - Cybersecurity – ISO21434



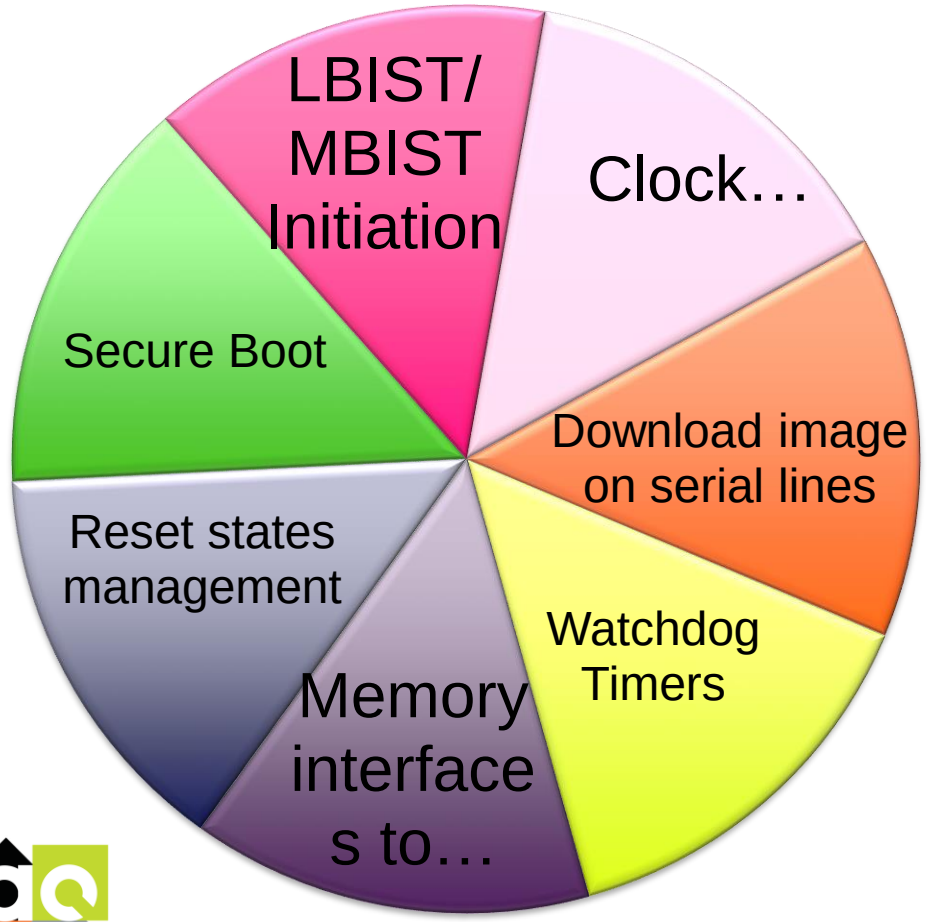
Boot code process flow

Booting sequence



- Power on reset
- RCOSC startup, OTP startup, security data load
- Safety, security checks
- BootLoader interface identification
- BootLoader image download
- Secure boot
- Jump to BootLoader and execution

Major portions of Bootcode



- The clocking section enables clock sources like PLL, XOSC, RC oscillators
- Programs it for communication parameters like baud rate, serial mode
- To tackle any hang situation, BootROM utilizes software watchdogs
- Application code is downloaded into various memory interfaces like flash, SDMMC, eMMC, etc.
- Devices working on the principle of RCT
- Run a built-in self-test (LBIST, MBIST) on startup

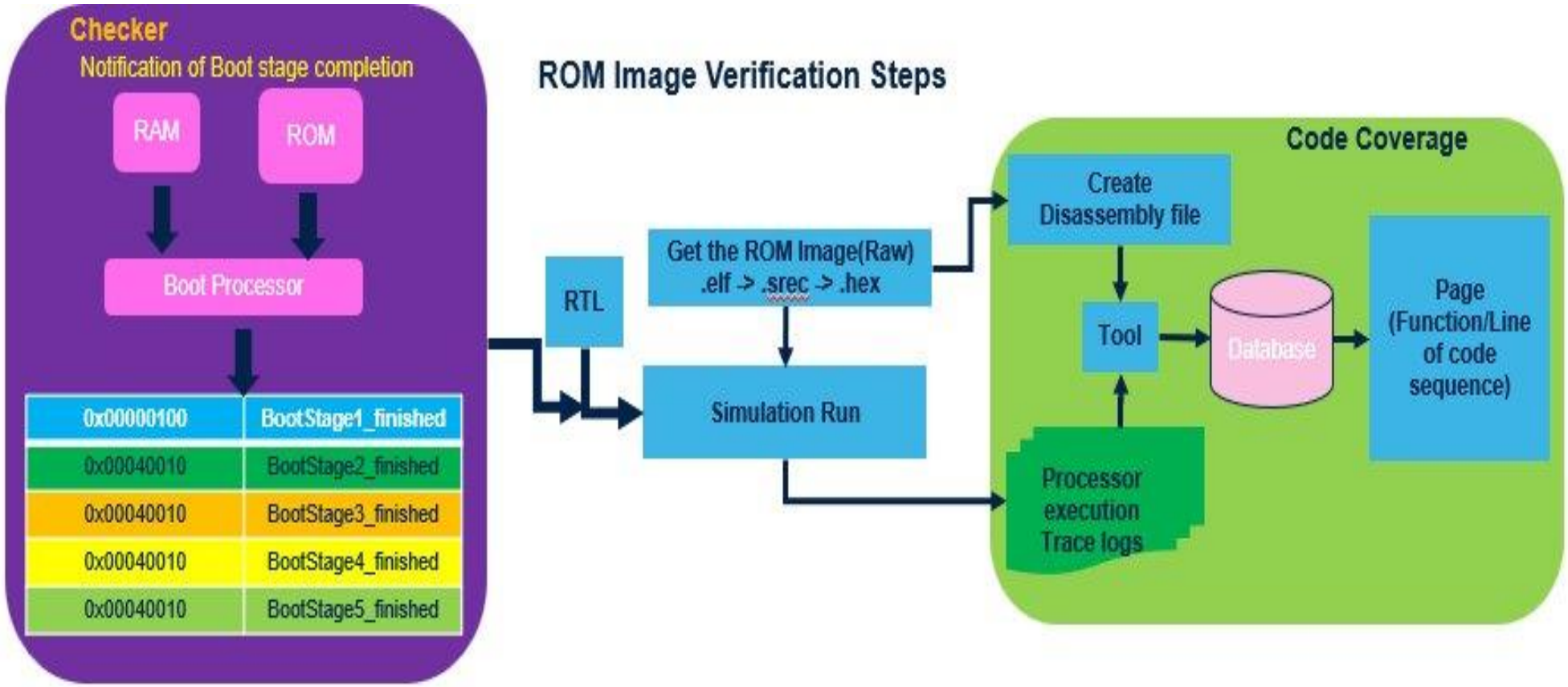
Verification challenge

- Complexity of systems
- A vast number of possible scenarios
- Through testing to ensure safety, security, and functionality
- Long runtime use cases
- Verification lacks standardization

Risk mitigation strategies

- Multifolded verification methodology in pre-silicon
- Automating ROM verification shifts Boot Code development left
- Code coverage & functional coverage
- Boot sequence checker

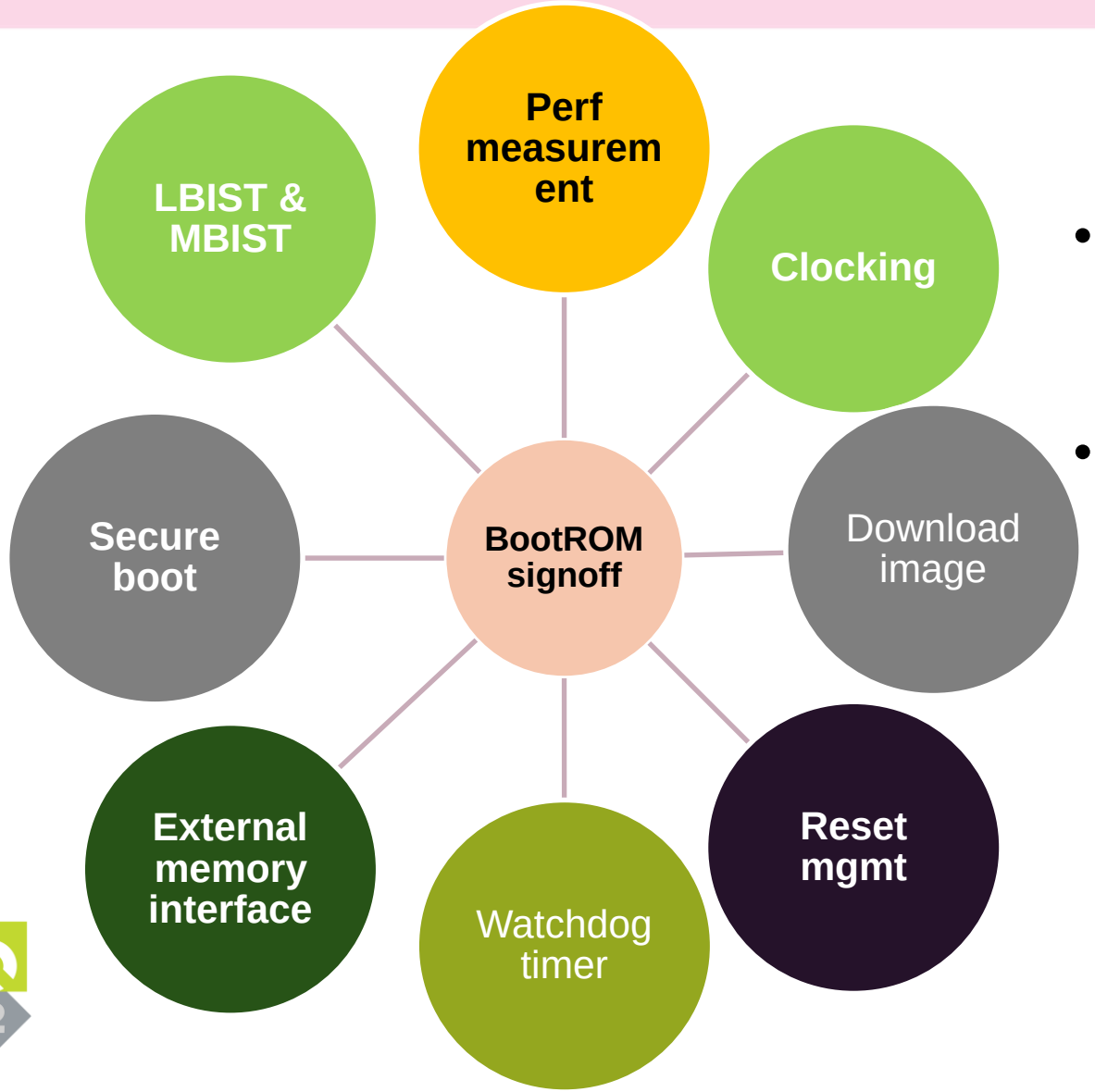
Verification environment



Comparative analysis of testing on various platforms

Description	Simulator	Emulator	Evaluation board
Clock programming	More suitable	Limited	Most suitable
Application download and execution	Less suitable	Good platform	Fastest way
Software watchdog	Difficult to check	Can be validated	Can be validated
Performance measurement	Difficult	Favorable	Most favorable
Code debugging ease	Difficult	Easy	Easy
Issues related to analog IPs behavioral model	Present	Present	Not present
RTL signal visibility	100%	Limited visibility	Almost none
Code coverage	Difficult to analyze	Doable	Doable
Gate Level verification	Applicable	Cannot be done	NA
Speed of execution	Slowest	Fast	Fastest
Reset states management	Can be validated	Difficult to validate	Most favorable
LBIST and MBIST initiation	Less suitable	Favorable	Most favorable
Cost of bug fixing	Very less	Very less	Huge

Best way to signoff BootROM



- Very important to do 100% testing of the code before base layer tape-out of the SoC
- By testing BootROM in simulation and emulation environments simultaneously, thorough coverage of BootROM code can be ensured

Conclusion

- The comparison table provides an at-a-glance guide for users to tap into the best functionalities of the three platforms for various target needs
- Lastly, the focus narrows to a practical, dependable table that suggests the best options for validating code before tape-out (simulation and emulation)
- This analysis can be leveraged for any scenario sharing similar content and DUT infrastructure

Acknowledgement

This is one of our initiatives to share our technology and innovations with various communities and foster microelectronics-related activities in the frame of the

Important Project of Common European Interest on Microelectronics and Communication Technologies

(**IPCEI ME/CT**)

IPCEI Microelectronics and Communication Technologies

Funded by the European Union
NextGenerationEU

Thank you

SPONSORED BY