Automotive Security: CAN FD Secure Boot and Message Authentication With TA100-VAO for ADAS and IVI Systems

Presenter: Peter Kwak, Principal Embedded Solutions Engineer

Date: December 13, 2022

SMART | CONNECTED | SECURE

Agenda

- **Overview**
  - Market driven cybersecurity specifications
    - TrustAnchor100™ (TA100) ECU Security Upgrade
  - Secure Boot
  - Secure Communications

Influences on New Automotive Security Standards

ISO/IEC 9797-1 / ISO 26262
ISO 27001 / IEC 62443
J3061/ J3101

NHTSA

National Highway Traffic
Safety Administration

Automotive Security

CAN

CAN FD

OEM

Security
Specifications

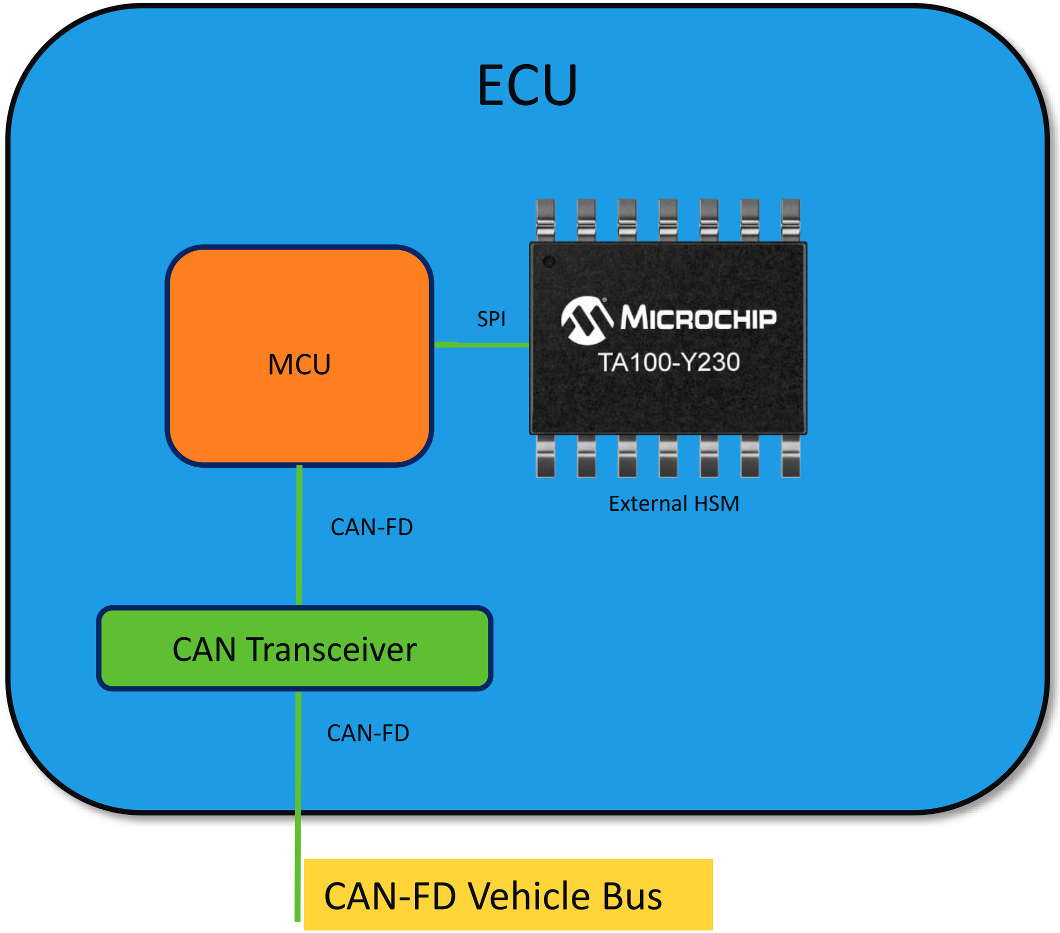
Cybersecurity Specification Review Services

- **In-Vehicle Network Security Trends**
  - Secure Boot and FW Update
  - Message Authentication
  - WPC / TLS / HDCP / USB-C Security / EV Battery Authentication

- **Microchip offers security specification review**
  - Microchip works directly with OEM's during specification development
  - Microchip performs line by line review for Tier-1's
    - Microchip can help respond to RFI/RFQ's
    - Microchip can help architect compliant security strategies

ECU Security Upgrade Architecture Options

- **Migrating to dual core secure MCU**
  - Expensive, risky and slow upgrade
- **Instead, keep your existing MCU**
  - Upgrade security with an external HSM



Agenda

- **Overview**
  - Market driven cybersecurity specifications
    - TrustAnchor100™ (TA100) ECU Security Upgrade
  - Secure Boot
  - Secure Communications

V71 Memory Example

(V71 micro datasheet section 11)

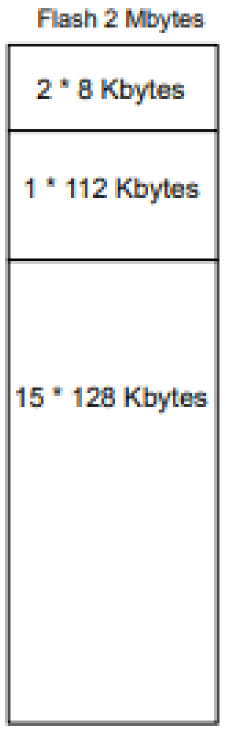
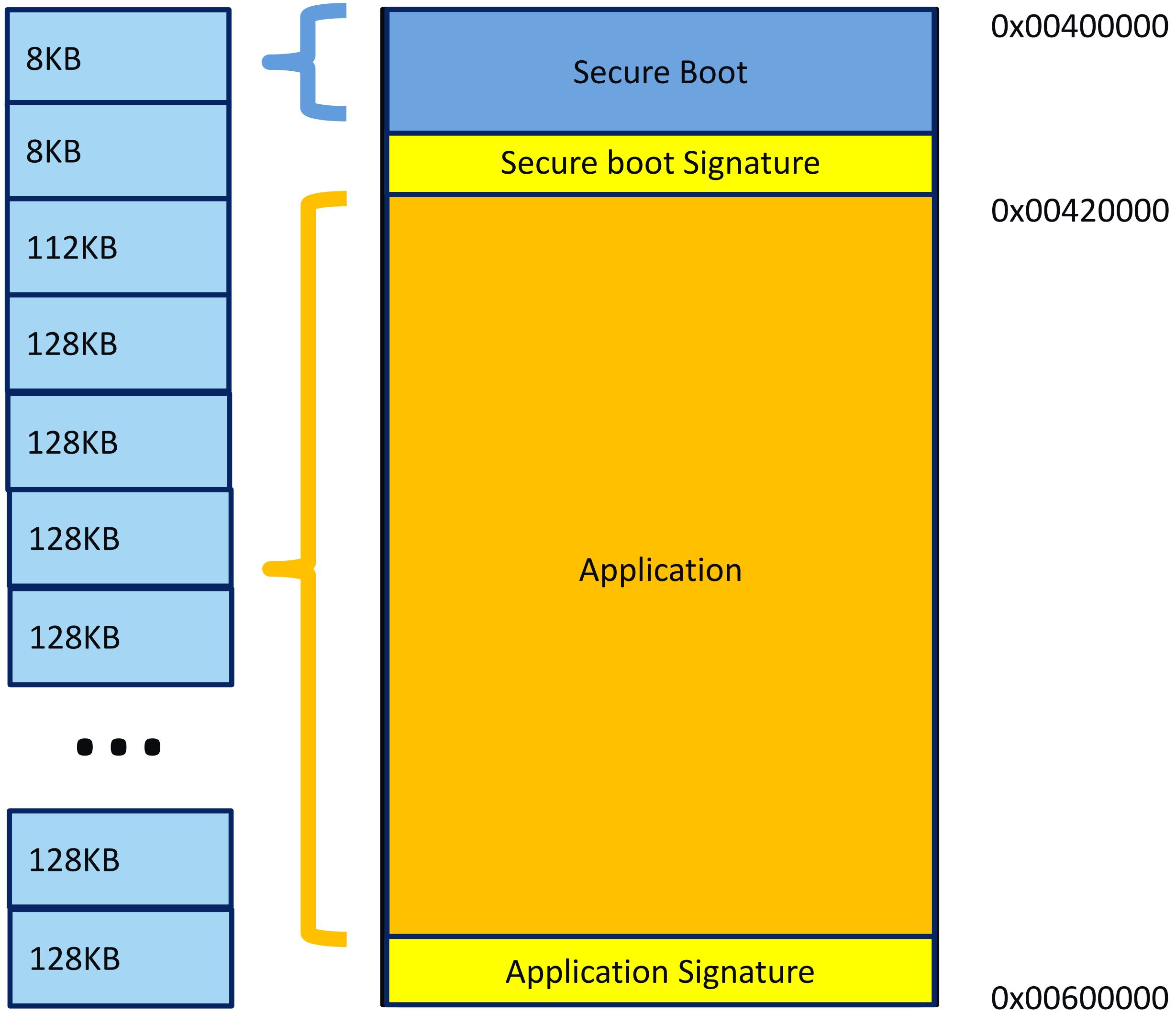


Figure 11-3. Flash Size

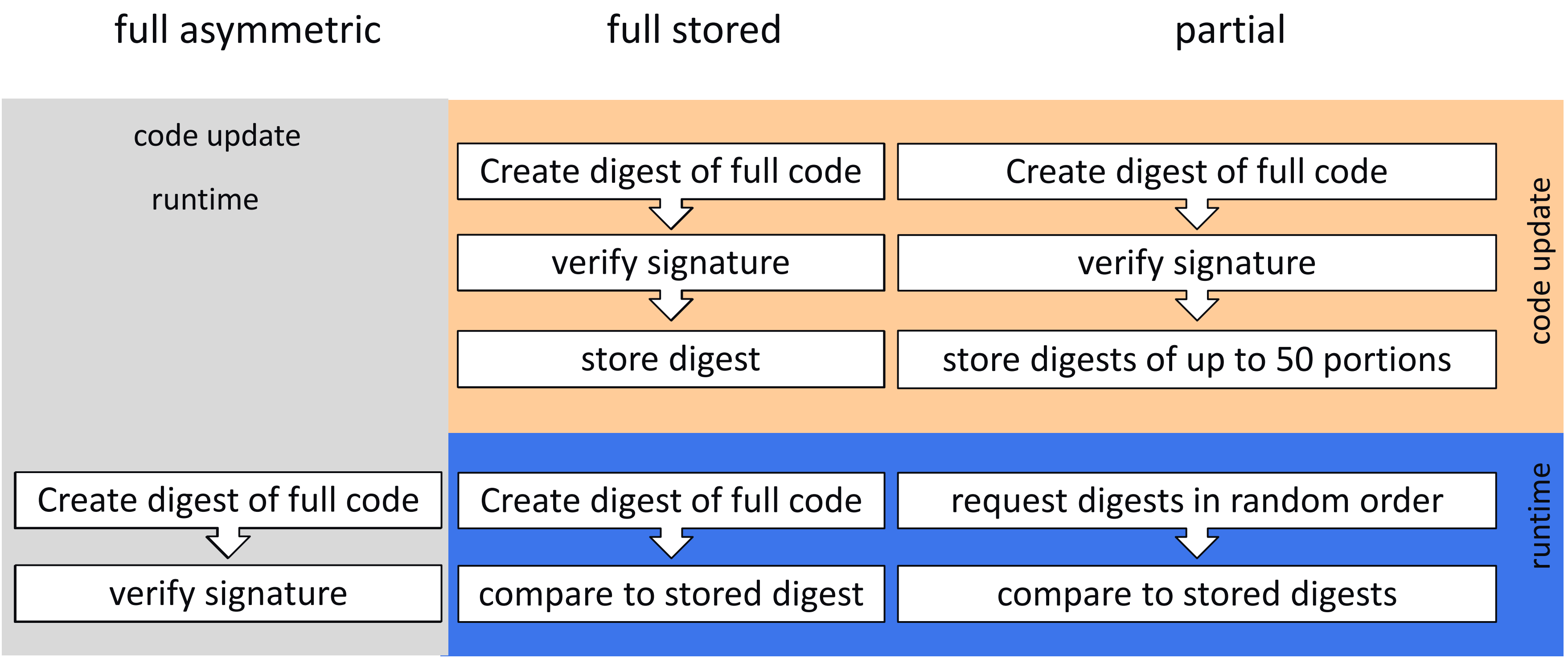


8KB
8KB
112KB
128KB
128KB
128KB
128KB
• • •
128KB
128KB

Secure Boot
Secure boot Signature
Application
Application Signature

0x00400000
0x00420000
0x00600000

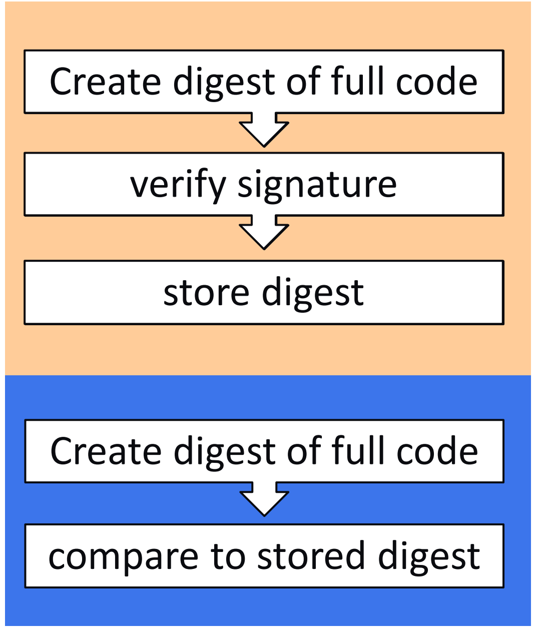
Secure Boot:: Modes for Application Memory



	full asymmetric	full stored	partial
code update	code update runtime	Create digest of full code verify signature store digest	Create digest of full code verify signature store digests of up to 50 portions
runtime	Create digest of full code verify signature	Create digest of full code compare to stored digest	request digests in random order compare to stored digests

Secure Boot:: Modes for Application Memory

full stored



Create digest of full code

verify signature

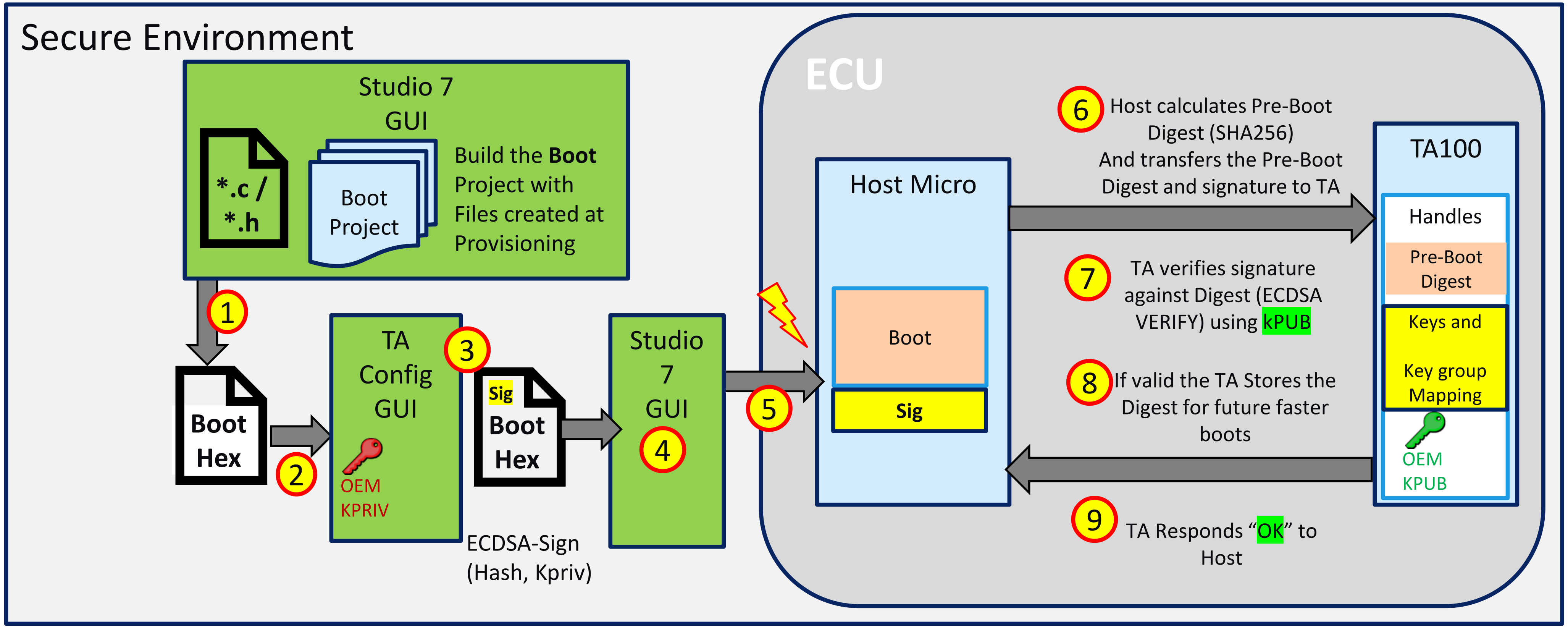
store digest

Create digest of full code

compare to stored digest

Pre-Boot

(First Reset)



Secure Environment
Studio 7 GUI
*.c / *.h
Boot Project
Build the Boot Project with Files created at Provisioning
1
Boot Hex
2
TA Config GUI
3
OEM KPRIV
Sig
Boot Hex
ECDSA-Sign (Hash, Kpriv)
Studio 7 GUI
4
5
ECU
Host Micro
Boot
Sig
6
Host calculates Pre-Boot Digest (SHA256) And transfers the Pre-Boot Digest and signature to TA
7
TA verifies signature against Digest (ECDSA VERIFY) using kPUB
8
If valid the TA Stores the Digest for future faster boots
9
TA Responds "OK" to Host
TA100
Handles
Pre-Boot Digest
Keys and
Key group Mapping
OEM KPUB

Note: boot is signed with the **private** key

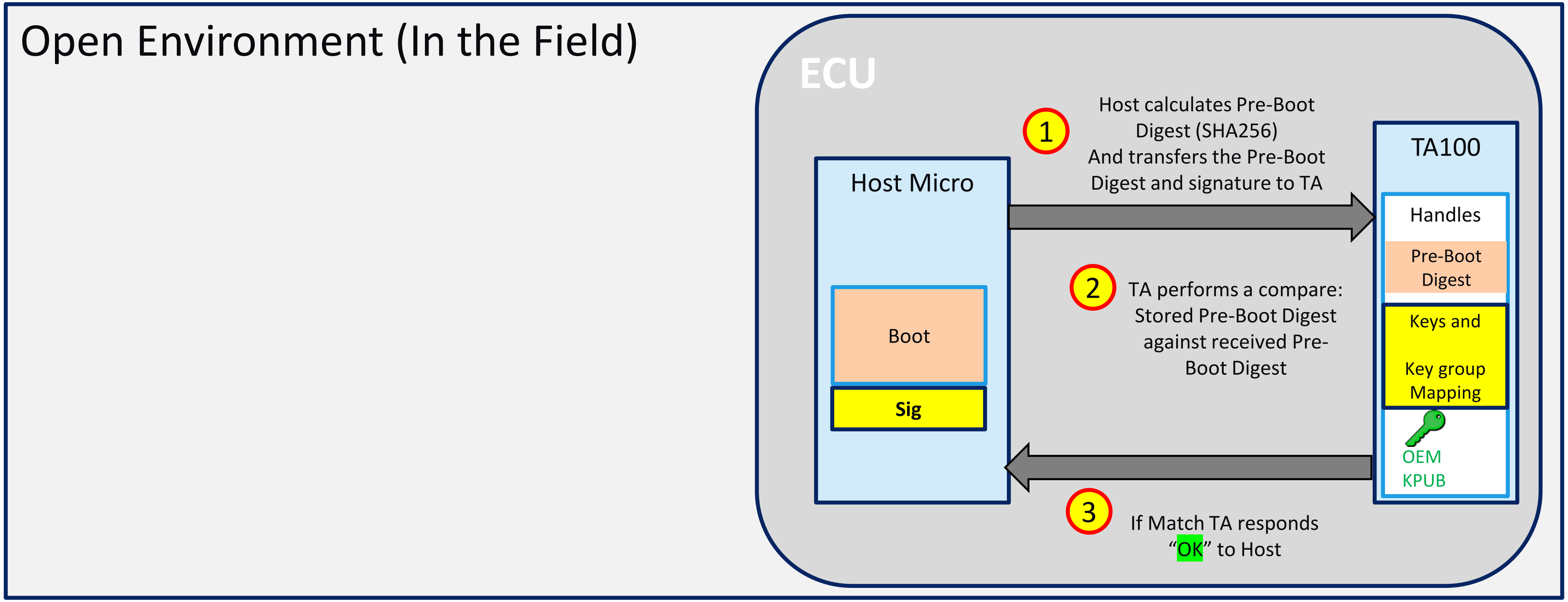
Note: Signature is 64 bytes

Note: Pre-boot must pass before secure boot command can be executed

Note: Pre-boot is ONLY full store

Pre-Boot

(Every Reset After Successful Digest Store)



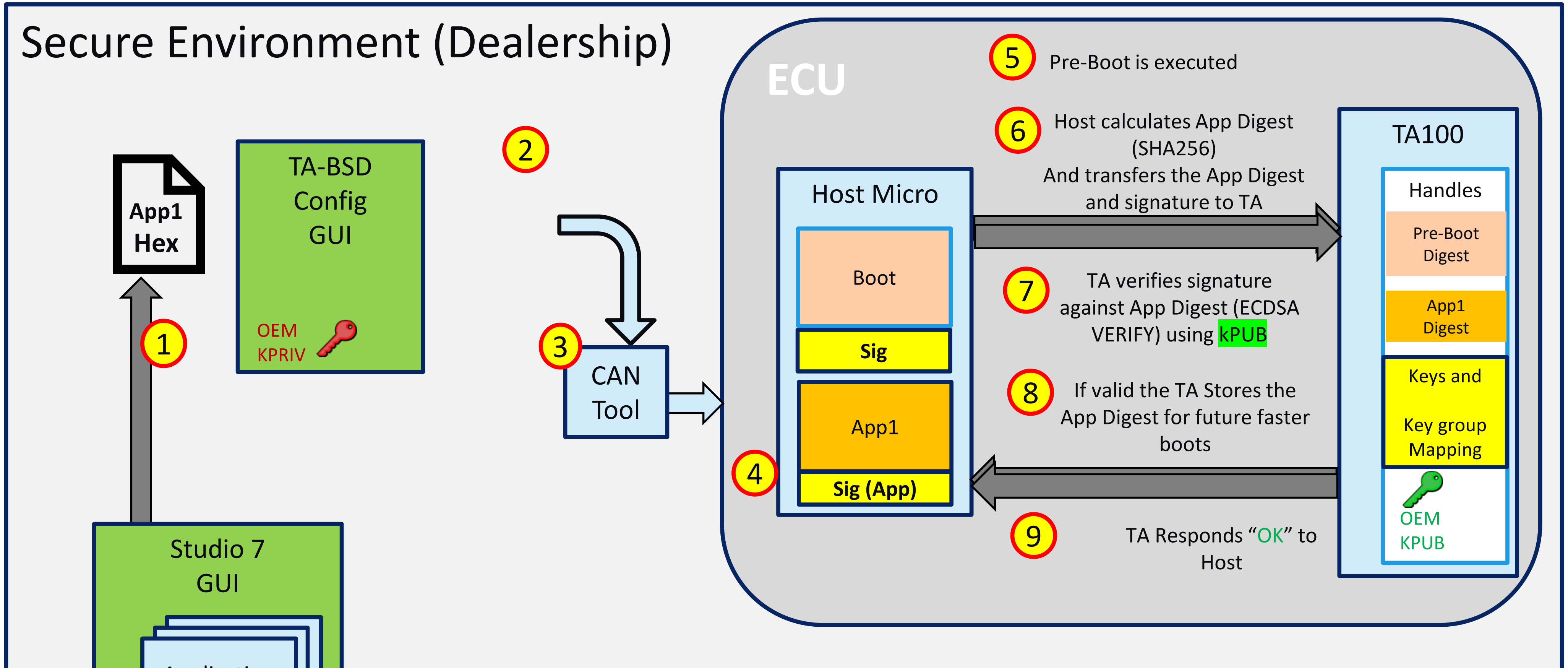
Open Environment (In the Field)
ECU
Host Micro
Boot
Sig
1
Host calculates Pre-Boot Digest (SHA256) And transfers the Pre-Boot Digest and signature to TA
2
TA performs a compare: Stored Pre-Boot Digest against received Pre-Boot Digest
3
If Match TA responds "OK" to Host
TA100
Handles
Pre-Boot Digest
Keys and
Key group Mapping
OEM KPUB

Note: Time consuming "ECDSA VERIFY" operation is NOT needed

Note: When successful Bootloader is waiting for Application

Full Store Secure Boot

(Application Update)



Secure Environment (Dealership)
App1 Hex
TA-BSD Config GUI
OEM KPRIV
1
2
3
CAN Tool
Studio 7 GUI
Application Project
ECU
4
Host Micro
Boot
Sig
App1
Sig (App)
5
Pre-Boot is executed
6
Host calculates App Digest (SHA256) And transfers the App Digest and signature to TA
7
TA verifies signature against App Digest (ECDSA VERIFY) using kPUB
8
If valid the TA Stores the App Digest for future faster boots
9
TA Responds "OK" to Host
TA100
Handles
Pre-Boot Digest
App1 Digest
Keys and Key group Mapping
OEM KPUB

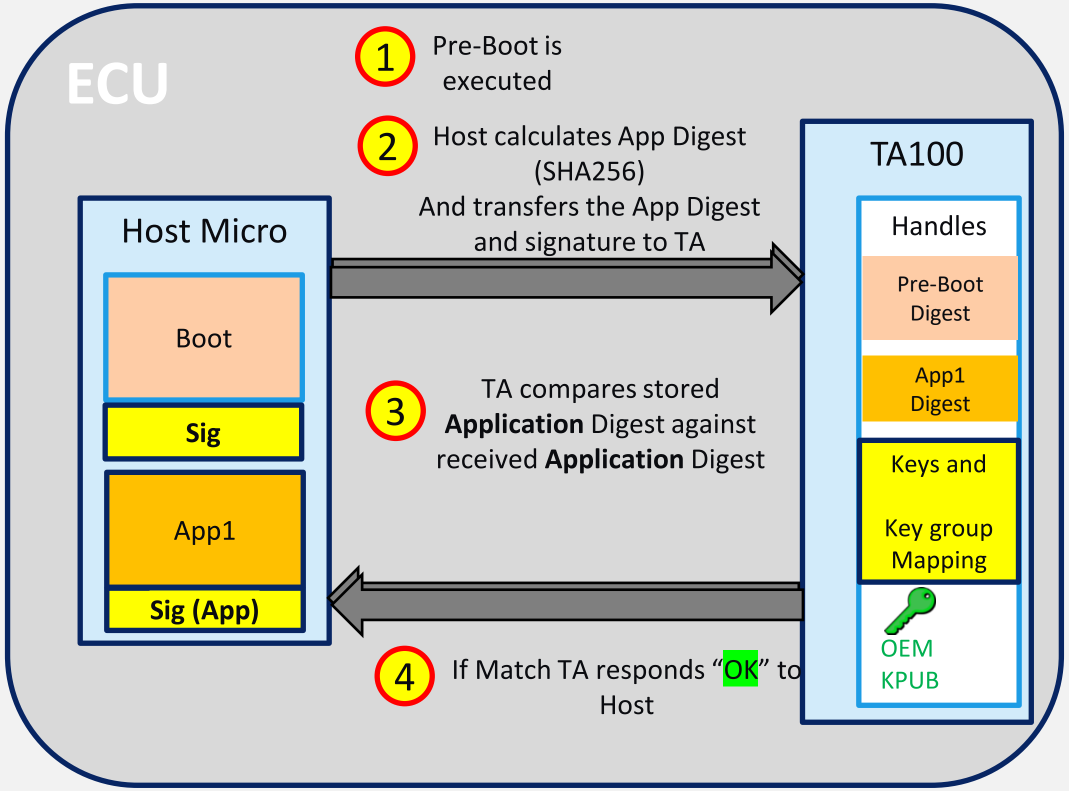
Note: For Asymmetric boot scheme there would be no need to store the digest since it transmits all bytes

Note: For Partial boot scheme the TA should store more then one digest

Full Store Secure Boot

(Every Reset After Successful Application Digest Store)

Open Environment (In the Field)



1 Pre-Boot is executed

2 Host calculates App Digest (SHA256) And transfers the App Digest and signature to TA

3 TA compares stored **Application** Digest against received **Application** Digest

4 If Match TA responds “OK” to Host

Note: Time consuming “ECDSA VERIFY” operation is NOT needed

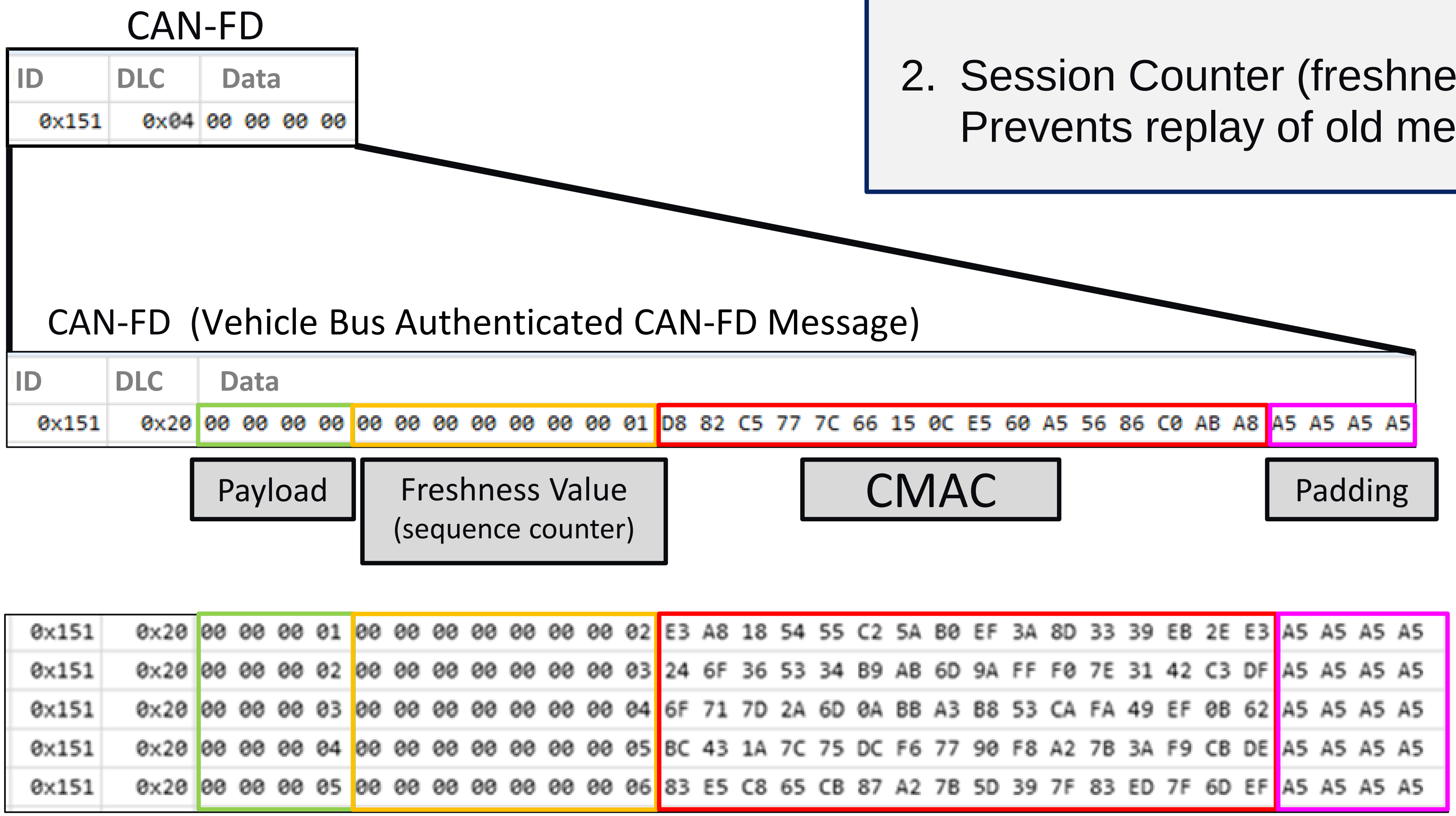
Note: When the Bootloader gets the “OK” from the TA100… the application will be executed

Agenda

- **Overview**
  - Market driven cybersecurity specifications
    - TrustAnchor100™ (TA100) ECU Security Upgrade
  - Secure Boot
  - Secure Communications

CAN Message Authentication Code (MAC)

1. MAC: Establishes authenticity
2. Session Counter (freshness value): Prevents replay of old messages



CAN-FD

ID	DLC	Data
0x151	0x04	00 00 00 00

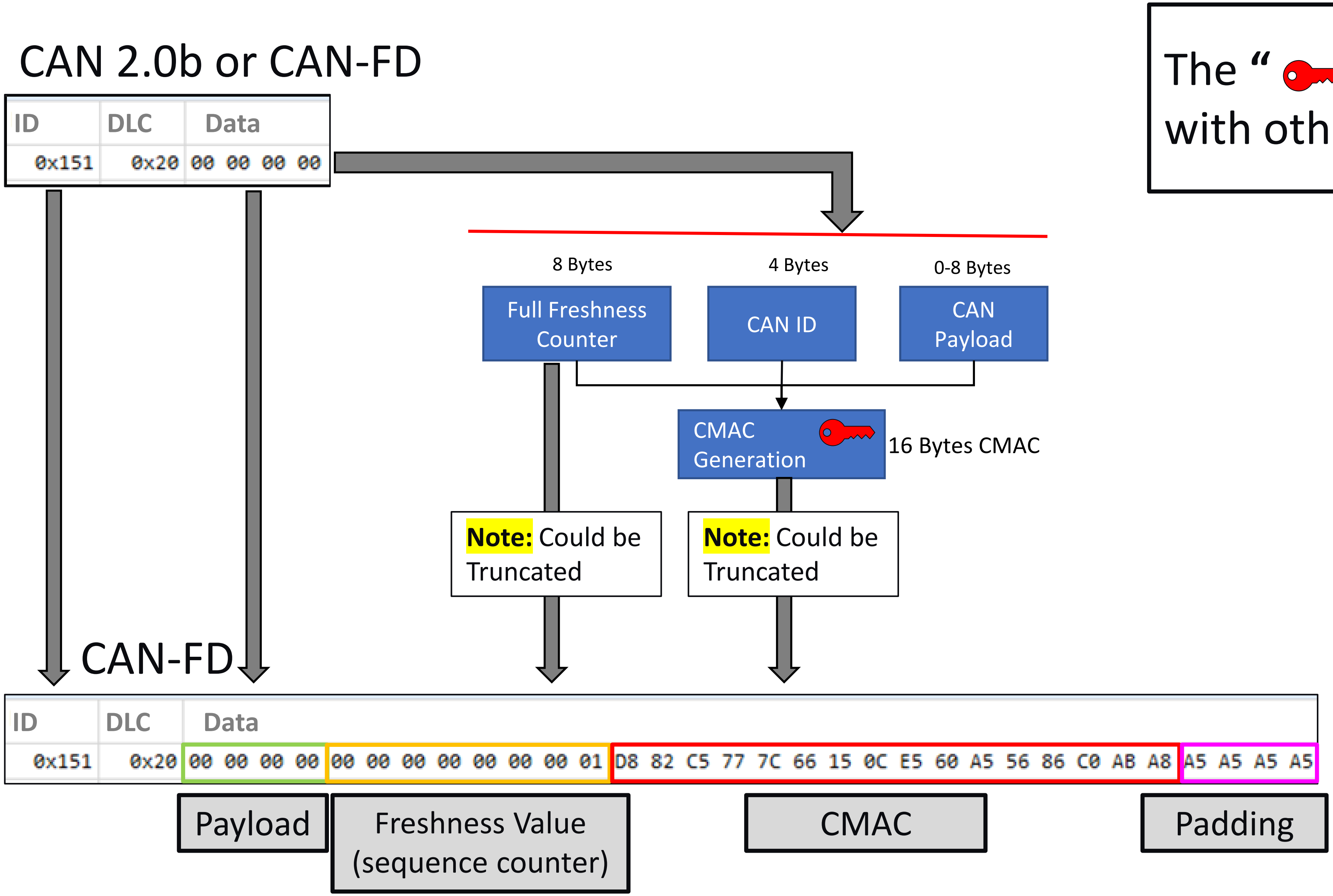
CAN-FD (Vehicle Bus Authenticated CAN-FD Message)

ID	DLC	Data
0x151	0x20	00 00 00 00 00 00 00 00 00 00 00 01 D8 82 C5 77 7C 66 15 0C E5 60 A5 56 86 C0 AB A8 A5 A5 A5 A5

0x151	0x20	00 00 00 01	00 00 00 00 00 00 00 02	E3 A8 18 54 55 C2 5A B0 EF 3A 8D 33 39 EB 2E E3	A5 A5 A5 A5
0x151	0x20	00 00 00 02	00 00 00 00 00 00 00 03	24 6F 36 53 34 B9 AB 6D 9A FF F0 7E 31 42 C3 DF	A5 A5 A5 A5
0x151	0x20	00 00 00 03	00 00 00 00 00 00 00 04	6F 71 7D 2A 6D 0A BB A3 B8 53 CA FA 49 EF 0B 62	A5 A5 A5 A5
0x151	0x20	00 00 00 04	00 00 00 00 00 00 00 05	BC 43 1A 7C 75 DC F6 77 90 F8 A2 7B 3A F9 CB DE	A5 A5 A5 A5
0x151	0x20	00 00 00 05	00 00 00 00 00 00 00 06	83 E5 C8 65 CB 87 A2 7B 5D 39 7F 83 ED 7F 6D EF	A5 A5 A5 A5

CAN Message Authentication Code (MAC)

The “ 🔑 “ is an AES key shared with other nodes on the network



CAN 2.0b or CAN-FD

ID	DLC	Data
0x151	0x20	00 00 00 00

8 Bytes

Full Freshness Counter

4 Bytes

CAN ID

0-8 Bytes

CAN Payload

CMAC Generation 🔑

16 Bytes CMAC

Note: Could be Truncated

Note: Could be Truncated

CAN-FD

ID	DLC	Data			
0x151	0x20	00 00 00 00	00 00 00 00 00 00 00 01	D8 82 C5 77 7C 66 15 0C E5 60 A5 56 86 C0 AB A8	A5 A5 A5 A5
		Payload	Freshness Value (sequence counter)	CMAC	Padding

Application

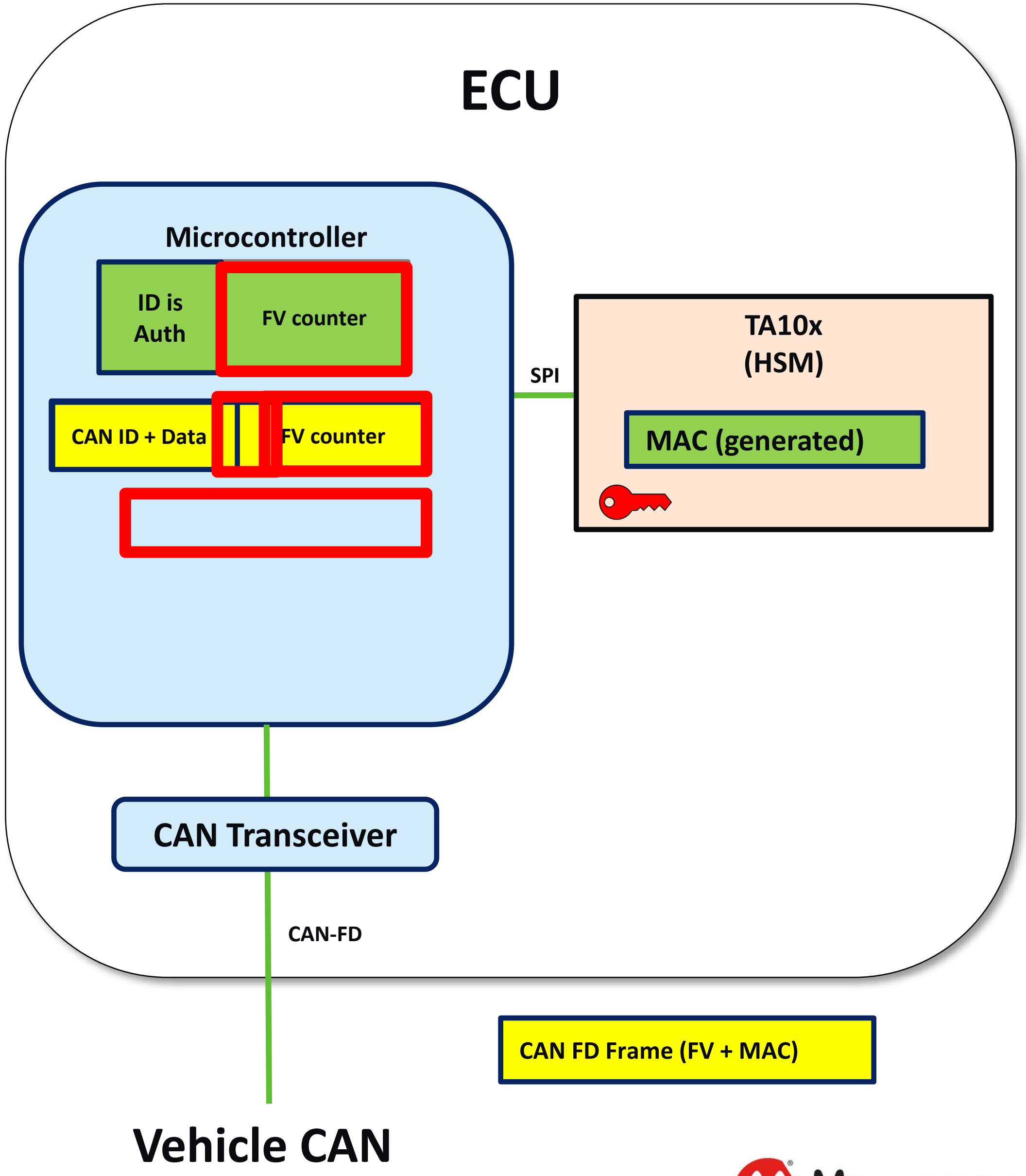
“Lookup” Table (in host)

Design Time Data					Run Time Data
CAN Message ID	Index of Key Used	Size of Message	Size FV on the bus	Size MAC on the bus	Full 8 byte FV counter
CAN Message ID	Index of Key Used	Size of Message	Size FV on the bus	Size MAC on the bus	Full 8 byte FV counter
CAN Message ID	Index of Key Used	Size of Message	Size FV on the bus	Size MAC on the bus	Full 8 byte FV counter
CAN Message ID	Index of Key Used	Size of Message	Size FV on the bus	Size MAC on the bus	Full 8 byte FV counter
CAN Message ID	Index of Key Used	Size of Message	Size FV on the bus	Size MAC on the bus	Full 8 byte FV counter
CAN Message ID	Index of Key Used	Size of Message	Size FV on the bus	Size MAC on the bus	Full 8 byte FV counter

CAN MAC Flow

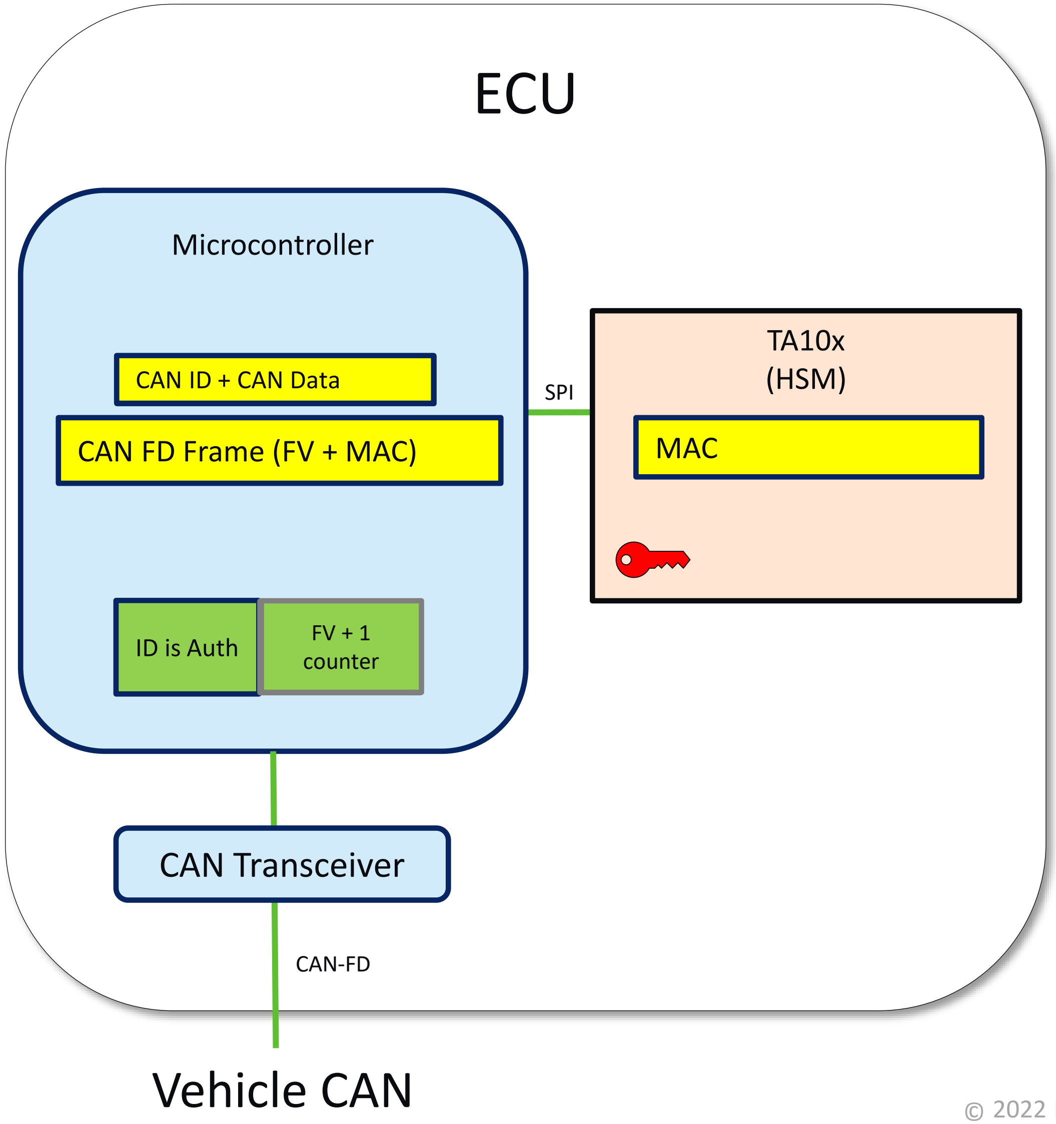
(Reception)

- The “ 🔑 “ is a symmetric private AES128 key
- The same key is shared with other nodes



CAN MAC Flow

(Transmit)



- The “ 🔑 “ is a symmetric private AES128 key
- The same key is shared with other nodes

Thank You

Microchip provides material in this webinar strictly “as is” for informational purposes only and without any warranties.
This material is deemed “Content” under Microchip’s Website Terms and Conditions (“Terms of Use”) and
governed by such Terms of Use available at www.microchip.com.

MICROCHIP