Firmware Updates for Renesas MCU

Introduction for "FW UP Module"

〜 RX Family, RL78 Family 〜

2025 JUL

EMBEDDED PROCESSOR & CONTROLLER SOLUTION
MARKETING DEPT
EMBEDDED PROCESSING MARKETING DIVISION
EMBEDDED PROCESSING PRODUCT GROUP
(EP/EPMD/EPMSM)

RENESAS ELECTRONICS CORPORATION

Rev2.00

EPMSM-AA-25-0017-2

RENESAS

CONTENTS

RENESAS

Market Needs

Firmware Update feature be necessary for a rapidly growing various IoT applications

IoT x Segments

Common Technology for various application
(IA, FA, HA, BA, HC, Power Equipment, Social Infra…etc.)

Continuous Service

Maintain latest firmware by OTA (Over-the-Air) via Cloud

Vulnerability Threats

Security Integration and Legislation such as CRA, U.S. Cyber Trust Mark, Other IoT Guideline

RENESAS

Applications that require Firmware Updates

As shown the previous page, MCU (🔲) Firmware Update function must be implemented in all system blocks of IoT products.
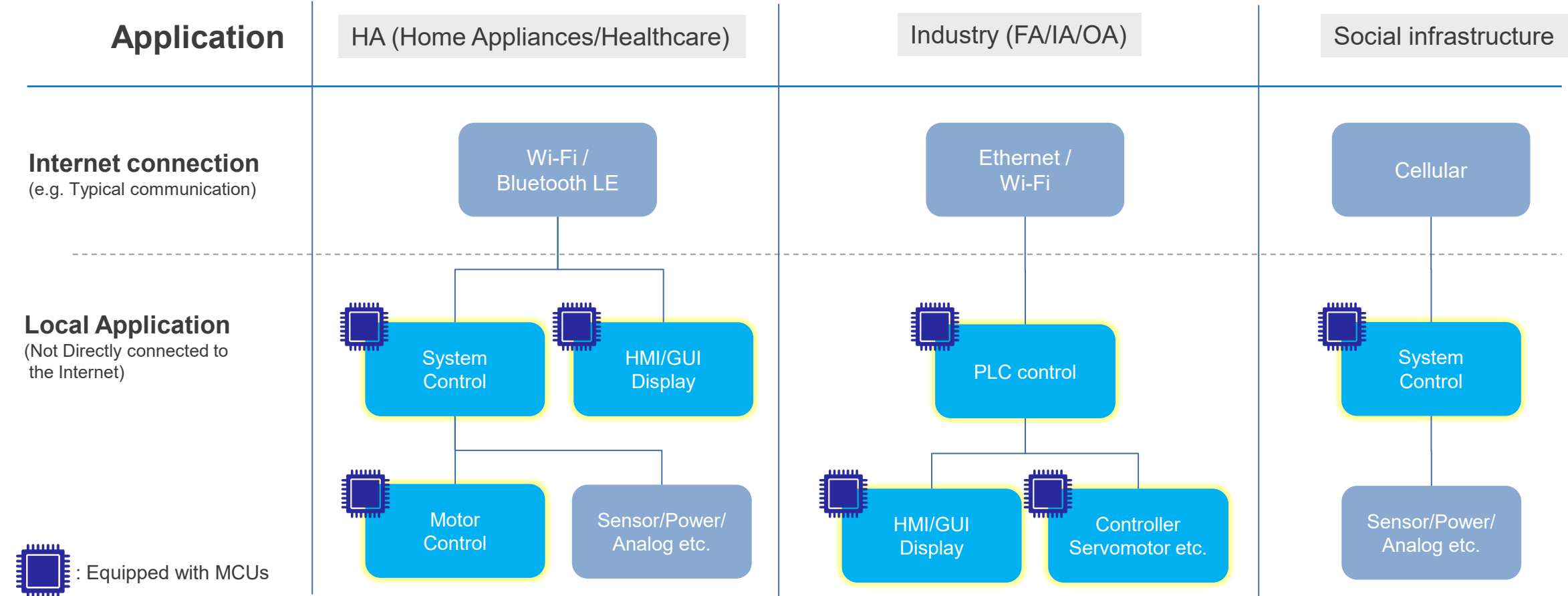
Application	HA (Home Appliances/Healthcare)	Industry (FA/IA/OA)	Social infrastructure
Internet connection (e.g. Typical communication)	Wi-Fi / Bluetooth LE	Ethernet / Wi-Fi	Cellular
Local Application (Not Directly connected to the Internet)	System Control / HMI/GUI Display / Motor Control / Sensor/Power/Analog etc.	PLC control / HMI/GUI Display / Controller Servomotor etc.	System Control / Sensor/Power/Analog etc.

🔲 : Equipped with MCUs

RENESAS

Renesas MCUs

Firmware Update Solution Introduction as "FW UP Module"

RENESAS

Features of Renesas "FW UP Module"

Your firmware update function can be easily incorporated using the FW UP Module!

Renesas Provide

Image Generator for Updating FW
Renesas Image Generator

+

Boot loader
Sample

+

FW UP Module

+

User apps

Pre-preparation : of the program
Creating mot Files and Key Pairs

Generates signed Update FW Image*
by the Renesas Image Generator

* OpenSSL generates
Key Pairs and
Code Signature

Input Update FW Image
via various Methods

Easy FW update with various system-configuration

The user application can implements how to acquire updated Images according to meet customer requirements.

Supports three FW Update Methods by Flash ROM type

1. Dual Mode : **Dual-bank Method** *1
2. Linear Mode : **Partial Update Method**
   ➤ Easy recovery even if FW updating fails
3. Linear Mode : **Full Update Method** *2
   ➤ FW can be updated in small memory footprint MCU

*1 Only some available for RX-Family MCU
*2 RL78 Family supports a method of using also an external flash memory for the buffer area.

Realize secure FW Update by validating Code signature

Supports firmware validation (ECDSA NIST P-256 and SHA256) with Secure Boot feature

Sample system configuration using the FW UP Module

The FW UP Module is ideal for FW update over various Primary Devices

Get an Update FW via serial communication

Primary Device (e.g.)
- **Renesas MCU**
- USB Memory Stick
- Communication device
- Other MCUs
- MPU, SoC etc....

Serial Communication I/F (*1)

FW UP Module For RX/RL78 Family

FW UP Module

| UART Driver | Cryptography Library | Flash Driver |

Boot loader
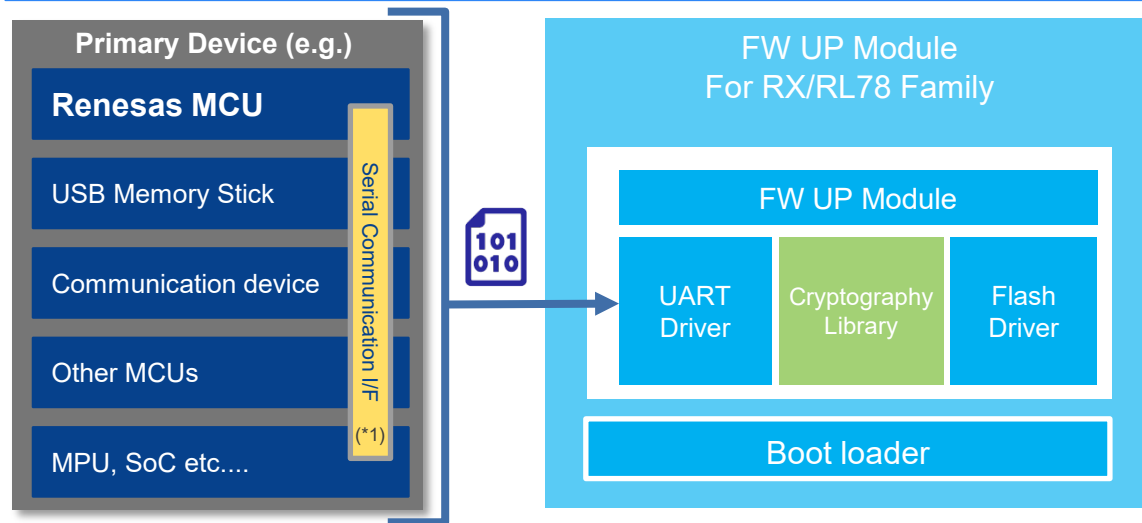
*1 : Firmware Updating Communications Module : (**RX**, **RL78**)

Application Notes

FW UP Module APN (Link) for RX Family

FW UP Module APN (Link) for RL78 Family

Compatible with various use cases

Stand-alone
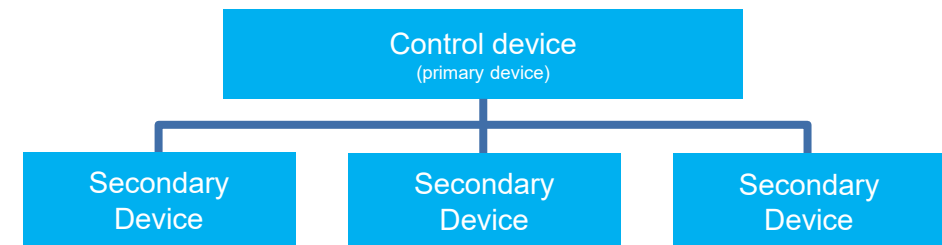(Products that run on a single device)

Air conditioners, etc. such as Paired systems

Outdoor unit

Indoor unit

Industrial system
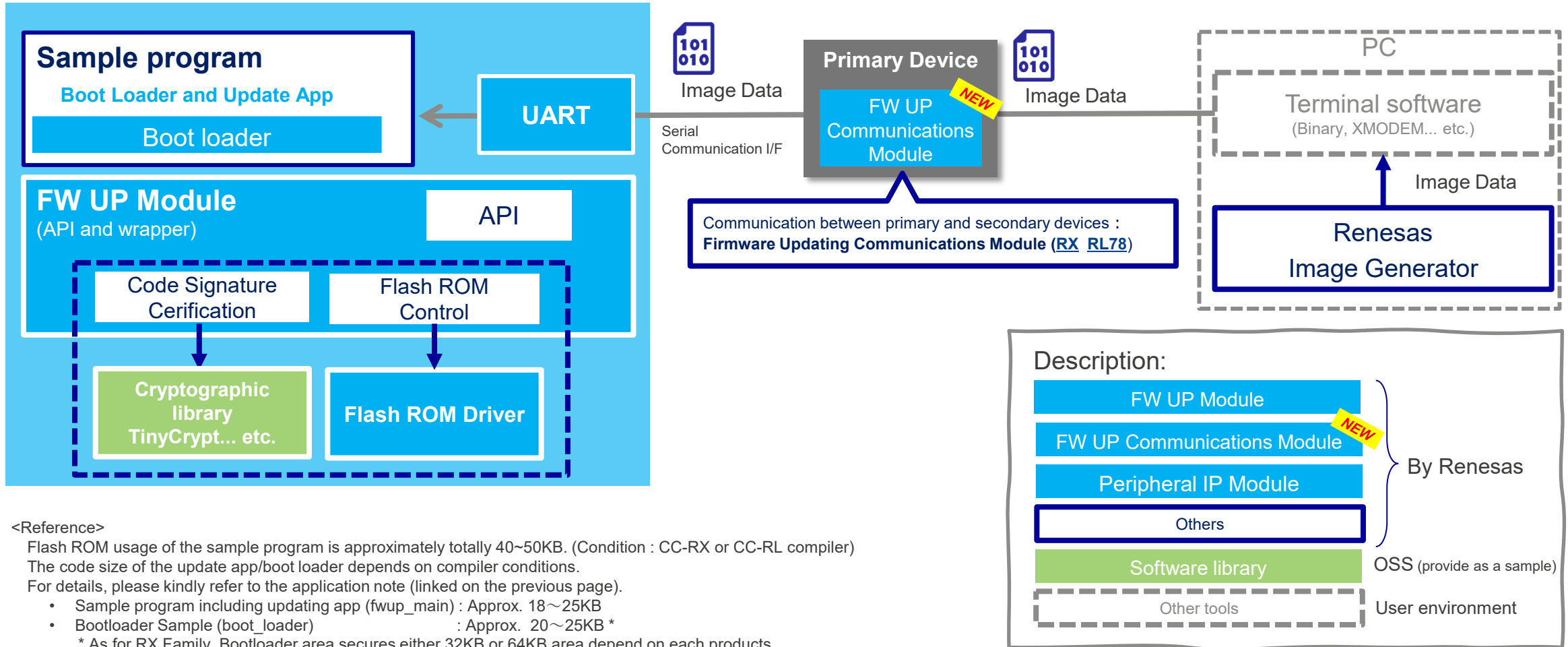(System consisting of control devices and connected various secondary devices)

Control device
(primary device)

| Secondary Device | Secondary Device | Secondary Device |

FW UP Module supports all RX Family, and RL78/G22, RL78/G23, RL78/G24 of RL78 Family.

System Configuration for Secondary Device Update with FW UP Module

> ## Secondary device firmware update

Sample program
Boot Loader and Update App
Boot loader

UART

FW UP Module
(API and wrapper)

API

Code Signature Cerification

Flash ROM Control

Cryptographic library TinyCrypt... etc.

Flash ROM Driver

Image Data
Serial Communication I/F

Primary Device
FW UP Communications Module `NEW`

Communication between primary and secondary devices :
Firmware Updating Communications Module (RX RL78)

Image Data

PC
Terminal software
(Binary, XMODEM... etc.)

Image Data

Renesas
Image Generator

Description:

FW UP Module

FW UP Communications Module `NEW`

Peripheral IP Module

Others

Software library

Other tools

By Renesas

OSS (provide as a sample)

User environment

<Reference>
Flash ROM usage of the sample program is approximately totally 40~50KB. (Condition : CC-RX or CC-RL compiler)
The code size of the update app/boot loader depends on compiler conditions.
For details, please kindly refer to the application note (linked on the previous page).
- Sample program including updating app (fwup_main) : Approx. 18~25KB
- Bootloader Sample (boot_loader)   : Approx. 20~25KB *
   * As for RX Family, Bootloader area secures either 32KB or 64KB area depend on each products.

What is Boot loader and its role

- Bootloader is a kind of software that is responsible for verifying the firmware Update FW Image and writing the firmware into Flash Memory.

- The bootloader sample for this FW UP Module uses an open source TinyCrypt.

- The roles of this bootloader for each Method are listed in the table below.

Firmware Update Method	Boot loader	
	FW Image Validation	Write to Flash ROM
1. Dual Mode (RX Family only) Dual Bank Method	✓	- *
2. Linear Mode Partial Update Method	✓	- *
3. Linear Mode Full Update Method	✓	✓

* : This bootloader also includes a specification to write an initial firmware image to Flash ROM in MCU at a factory site.
By writing only this bootloader in advance at the factory, the initial image is programed via UART communication into the Flash ROM.
In case users wish to use other communication methods than UART, the communication method can be customized by the users.

RENESAS

Selectable Update Method according to your MCU Flash ROM product
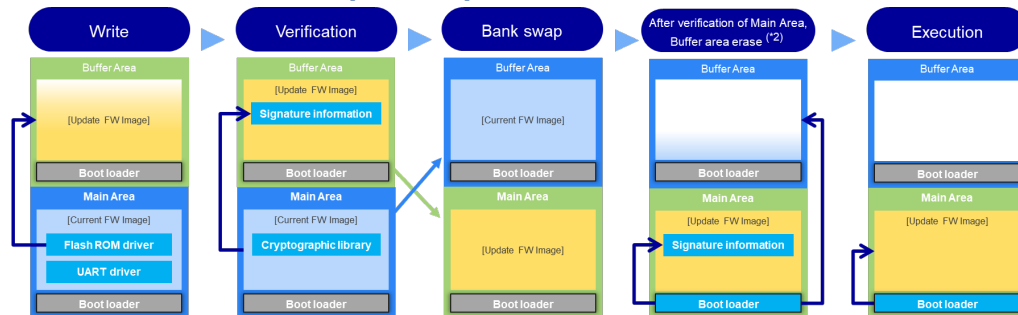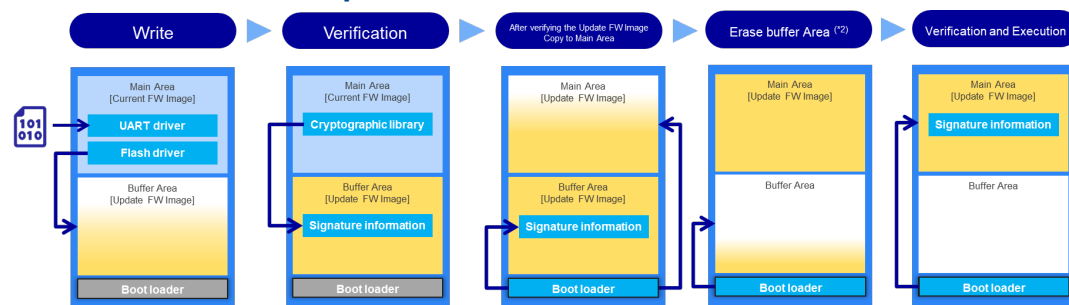## ~ Support for development with sample programs of each Method ~

<Sample program>
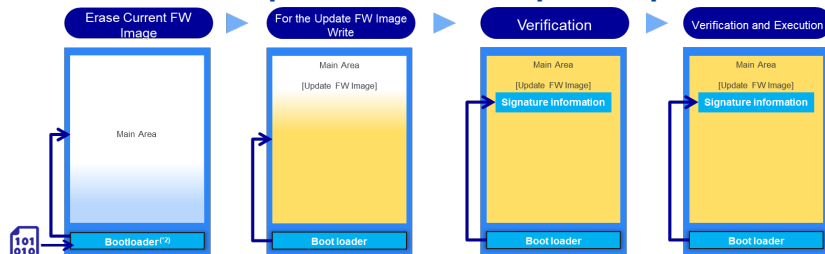RX : #R01AN6850 / Sample Zip
RL78 : #R01AN6374 / Sample Zip

1. Dual Mode: Dual-bank * update operation

2. Linear Mode: Partial Update Method *

3. Linear Mode: Full Update Method * update operation

Modes and Device Correspondence Table

Flash Product	4MB	2MB	1.5MB	1MB	768KB	512KB	384KB	256KB	128KB	96KB	64KB
RX130	-	-	-	-	-	L	L	L	L	-	-
RX140	-	-	-	-	-	-	-	L	L	-	-
RX231/230	-	-	-	-	-	L	L	L	-	-	-
RX23E-A	-	-	-	-	-	-	-	L	L	-	-
RX23E-B	-	-	-	-	-	-	-	L	L	-	-
RX24T	-	-	-	-	-	L	L	L	L	-	-
RX26T	-	-	-	-	-	DB/L	-	L	L	-	-
RX65N/651	-	DB/L	DB/L	L	L	L	-	-	-	-	-
RX66N	DB/L	L	-	-	-	-	-	-	-	-	-
RX66T	-	-	-	L	-	L	-	-	-	-	-
RX660	-	-	-	L	-	L	-	-	-	-	-
RX671	-	DB/L	DB/L	L	-	-	-	-	-	-	-
RX72M	DB/L	L	-	-	-	-	-	-	-	-	-
RX72N	DB/L	L	-	-	-	-	-	-	-	-	-
RL78/G22	-	-	-	-	-	-	-	-	-	L	-
RL78/G23	-	-	-	-	L	L	L	L	L	-	-
RL78/G24	-	-	-	-	-	-	-	L	-	-	L

DB : Products supporting Dual Mode (1. Dual Bank Method)
L : Products supporting Linear mode (2. Partial Update / 3. Full Update Method)
- : Not supported
red text: Products supporting the sample program(Sample provided in zip)

* RL78 family currently does **not** support the dual-bank Method. Also, the updating Method name is partially different from the application note for RL78. No.2 means "Partial Update Method" and No.3 means " Full Update Method (without buffer)."
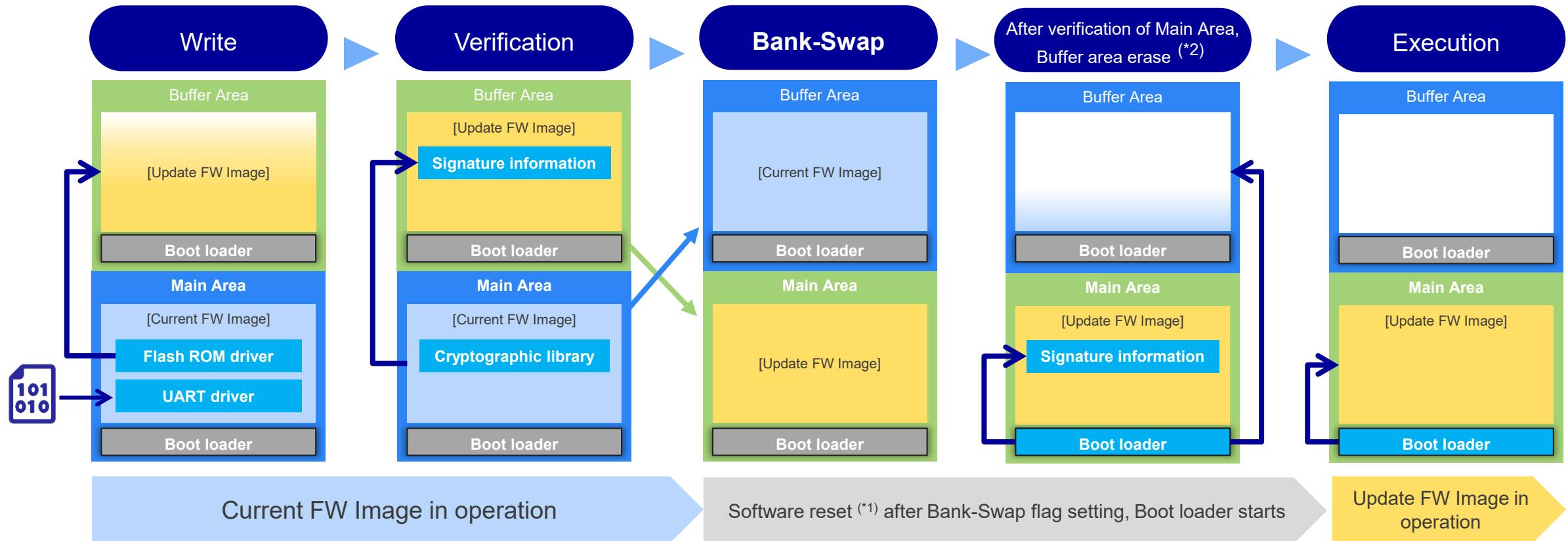
1. Dual Mode
Dual Bank Method: Normal operation

For use case that doesn't want to stop user applications, Dual-bank supported MCUs are recommended !

For **Dual Bank** supported MCUs, Update FW Image can be written while the current program on the main is executing!
Address placement management of programs is also _not necessary_ to utilize the **Bank-Swap** function.

Write	Verification	**Bank-Swap**	After verification of Main Area, Buffer area erase (*2)	Execution

Write
- Buffer Area
  - [Update FW Image]
  - Boot loader
- Main Area
  - [Current FW Image]
  - Flash ROM driver
  - UART driver
  - Boot loader

Verification
- Buffer Area
  - [Update FW Image]
  - Signature information
  - Boot loader
- Main Area
  - [Current FW Image]
  - Cryptographic library
  - Boot loader

Bank-Swap
- Buffer Area
  - [Current FW Image]
  - Boot loader
- Main Area
  - [Update FW Image]
  - Boot loader

After verification of Main Area, Buffer area erase (*2)
- Buffer Area
  - Boot loader
- Main Area
  - [Update FW Image]
  - Signature information
  - Boot loader

Execution
- Buffer Area
  - Boot loader
- Main Area
  - [Update FW Image]
  - Boot loader

Current FW Image in operation

Software reset (*1) after Bank-Swap flag setting, Boot loader starts

Update FW Image in operation

*1 : For initialization by a software reset, please see the "Reset Chapter" in the hardware manual of each MCU.
*2 : The demonstration program of the FW UP Module does not erase the buffer area. If it is necessary to erase the current(previous) Image before updating, users need to add the Image erase process of located the buffer side to prevent rollback.

RENESAS

1. Dual Mode
Dual Bank Method: Operation with error occurs

If the firmware update process fails due to a power-cut/signature verification failure as abnormal process, Enables to boot the Current FW Image and redo the firmware update,again.

|				
Buffer area	Disabled	Disabled	Enable	Enable
Main Area	Enable	Enable	Enable	Enable

Green text: Startable state / Black text: Unbootable state

Recovery to the Current FW Image is possible regardless of the timing of the error.

RENESAS
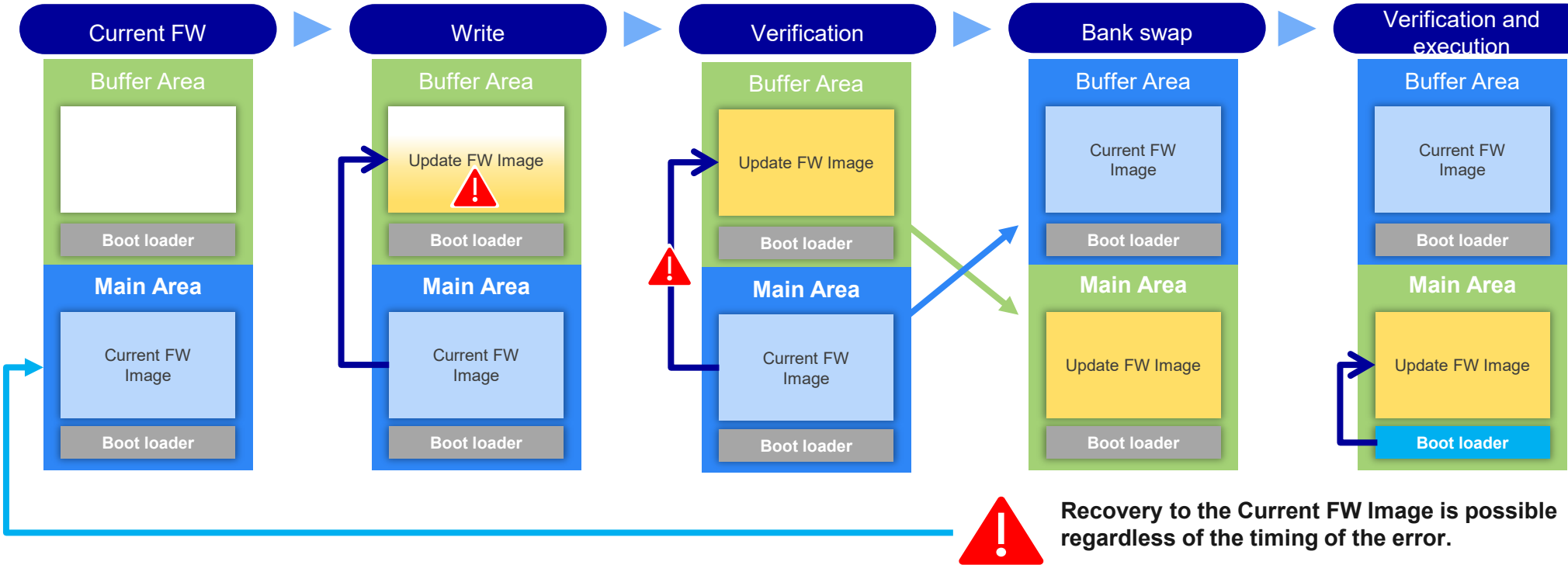
2. Linear Mode
Partial Update Method: Normal operation

Even if MCU does not support Dual Bank, a firmware can be renewed with the previous FW Image retained by using the Partial Update Method.

| Write | → | Verification | → | After verifying the Update FW Image Copy to Main Area | → | Erase buffer Area [*2] | → | Verification and Execution |

Write
- Main Area [Current FW Image]
  - UART driver
  - Flash driver
- Buffer Area [Update FW Image]
- Boot loader

Verification
- Main Area [Current FW Image]
  - Cryptographic library
- Buffer Area [Update FW Image]
  - Signature information
- Boot loader

After verifying the Update FW Image Copy to Main Area
- Main Area [Current FW Image]
- Buffer Area [Update FW Image]
  - Signature information
- Boot loader

Erase buffer Area [*2]
- Main Area [Current FW Image]
- Buffer Area
- Boot loader

Verification and Execution
- Main Area [Current FW Image]
  - Signature information
- Buffer Area
- Boot loader

Current FW Image in operation → After software reset [*1] Boot loader starts → After Software reset [*1], Boot loader starts → Update FW Image Operating

*1 : For initialization by a software reset, please see the "Reset Chapter" in the hardware manual of each MCU.
*2 : The demonstration program of the FW UP Module does not erase the buffer area. If it is necessary to erase the current(previous) Image before updating, users need to add the Image erase process of located the buffer side to prevent rollback.

RENESAS

2. Linear Mode
Partial Update Method: Operation when an error occurs

**If the firmware update fails due to a power-off or signature verification failure,
You can start the Image before the update and start the firmware update again.**

Main Area	Enable	Enable	Disabled	Enable	Enable
Buffer Area	Disabled	Disabled	Enable	Disabled	Disabled

Green text: Startable state / **Black text**: Unbootable state
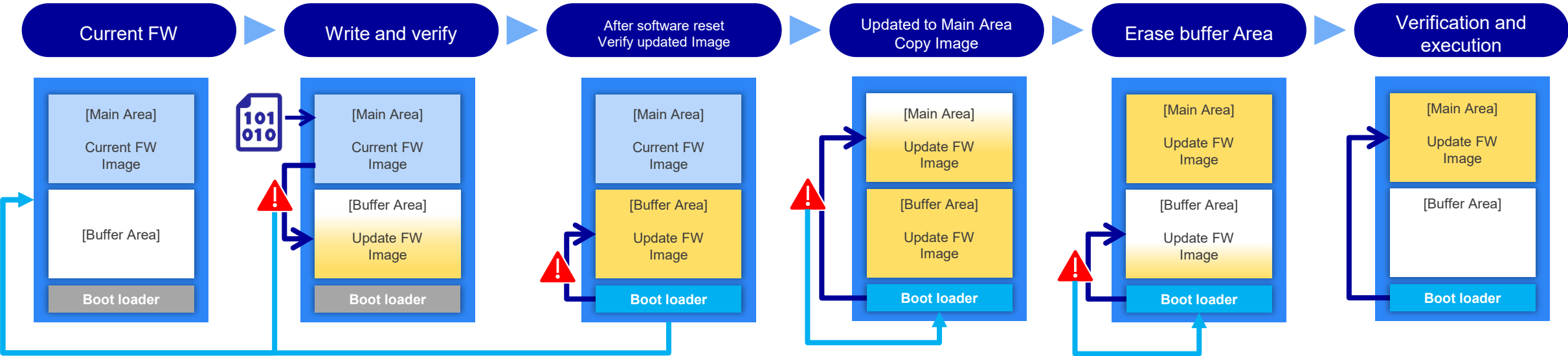
**There is always a valid Image on either side,
And recovery is possible in case of an error.**

RENESAS

3. Linear Mode
Full Update Method: Normal operation

If wants to secure the entire user program area of ROM at Low end MCU, Full Update Method is available.

Firmware updates can also be implemented for small ROM footprinted products using the Full Update Method.
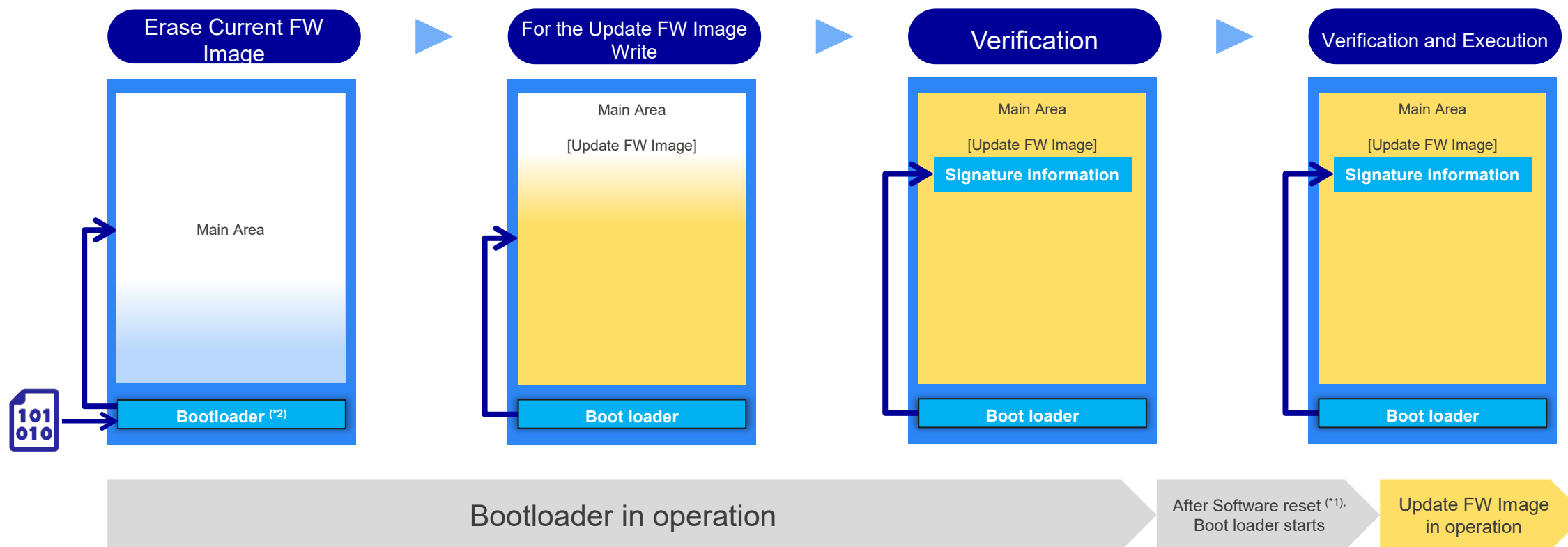
Erase Current FW Image

Main Area

Bootloader (*2)

For the Update FW Image Write

Main Area

[Update FW Image]

Boot loader

Verification

Main Area

[Update FW Image]

Signature information

Boot loader

Verification and Execution

Main Area

[Update FW Image]

Signature information

Boot loader

Bootloader in operation

After Software reset (*1), Boot loader starts

Update FW Image in operation

*1 : For initialization by a software reset, please see the "Reset Chapter" in the hardware manual of each MCU.
*2 : The demonstration program boot loader uses UART communication to obtain an updated Image. It is necessary to change it according to the communication Method customers want to use.

RENESAS

3. Linear Mode
Full Update Method: Operation when an error occurs

With the Full Update Method, if the firmware update fails, use the boot loader function to perform the firmware update again.

Main Area	Enable	Disabled	Disabled	Enable

Green text: Startable state / Black text: Unbootable state

Current FW	In the Current FW Image Erase	For the Update FW Image Write	Verification and execution

Current FW

[Main Area]

Current FW Image

Boot loader

In the Current FW Image Erase

[Main Area]

Current FW Image

Boot loader

For the Update FW Image Write

[Main Area]

Update FW Image

Boot loader

Verification and execution

[Main Area]

Update FW Image
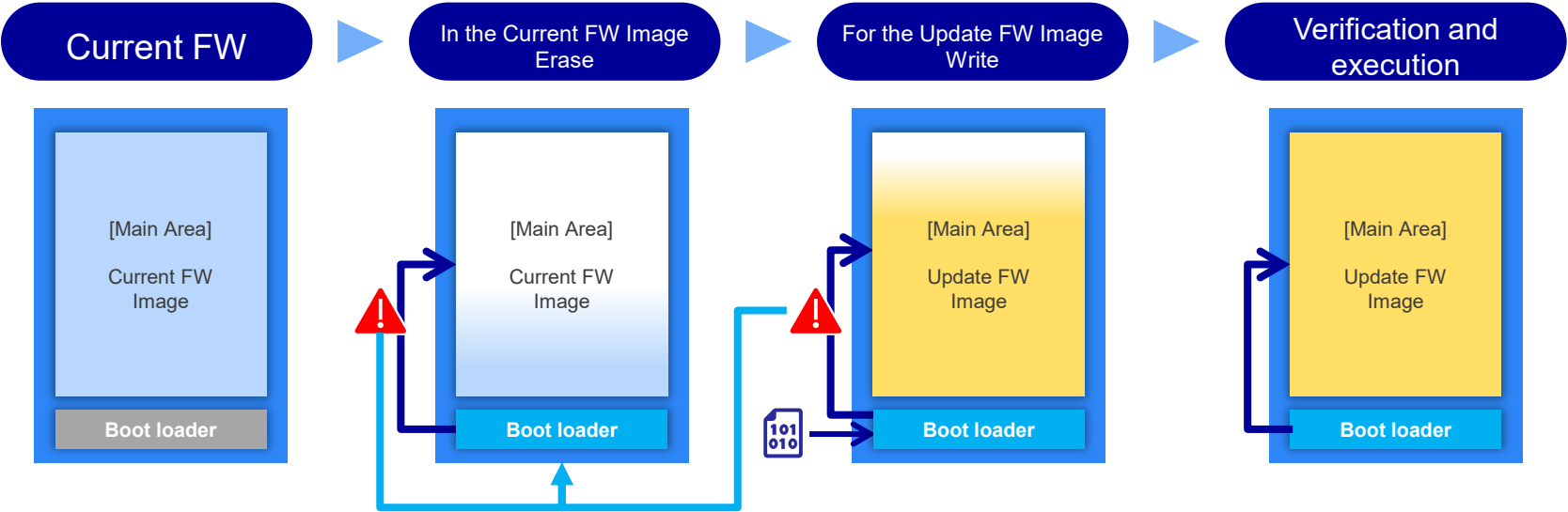
Boot loader

⚠ Repeat until the firmware update is successful.

RENESAS

Generate Signed Initial FW and Updated FW Image using with Renesas Image Generator

Renesas Image Generator can easily realize for both "signing to firmware" and "binding with bootloader and user firmware "

1. Pre-Preparation ▶ 2. Execute in the **Renesas Image Generator** ▶ 3. Complete of generating the initial Image

Generate Key and Signature Information

OpenSSL

Pub.
Public Key

Pvt.
Private Key

Generating Public Key and Private Key Key Pairs with the Key Generator

Create a firmware

e² studio

Application — Pub.

Boot loader — Pub.

Including a Public Key for Validating Signatures in **Application** and **Bootloader**.

Generate a mot Files

e² studio

Application.mot
Bootloader.mot
} Builde and Generate a mot file

Create RSU※ Header Addresses

Create RSU Header address information according to the device using parameter file

RSU Header-Address Information

※RSU=Renesas Secure Update

Grant RSU Header-Signing Information

Signing RSU Header Addresses and Applications with Private Key

RSU Header-Address Information
Application.mot
Pvt.

RSU Header-Signing Information

Generate initial Image with bootloader

Initial Image of the Dual Bank Method (.mot Files)

Boot loader (Code flash data) [Buffer Area]

RSU Header-Signing Information (0x200 turning tool)

RSU Header-Address Information (0x100 turning tool)

Application Program data

Boot loader (Code flash data) [Main Area]

Renesas.com

RENESAS

Appendix :

Introduction to Firmware Update Solutions

RENESAS

For optimal Firmware Updates on IoT products by Renesas MCU Family

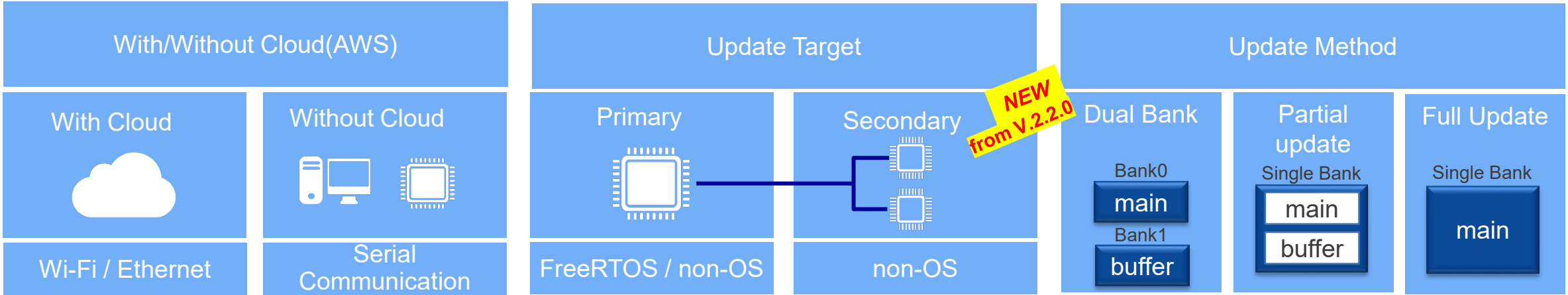**Support for firmware updates via OTA(Over the Air) through the cloud!**

MCU Family	Secondary device Firmware Update	IoT cloud OTA
RENESAS RA	RA0, RA2, RA4 Series Non-RTOS	RA6, RA8 Series MQTT / OTA / TLS / TCP/IP[1], Fleet Provisioning FreeRTOS *(Coming soon)*
RENESAS RX	RX100/200 Series Non-RTOS	RX65N MQTT / OTA / TLS / TCP/IP[1] / Fleet Provisioning FreeRTOS [1] : Ethernet(MCU) Only
RENESAS RL78	RL78/G22, RL78/G23, RL78/G24 Non-RTOS **Introduced in this material**	RL78/G23 MQTT / OTA / FreeRTOS [2] [2] : When using with these components with RL78/G23 (max ROM 768KB, RAM 48KB), RAM available for your app may be less than 10KB.

IoT cloud OTA

Ethernet, Wi-Fi, Cellular

1st OTA Via Cloud

IoT cloud OTA

UART, SPI, USB etc...

2nd OTA Via Cloud

Secondary device Firmware Update

RENESAS

Firmware Update Solutions List

		RENESAS RX	**RENESAS RL78**
Secondary Device Firmware Update	FW UP Module Sample code	✓ RX Family Firmware Update Module Using Firmware Integration Technology Application Notes - Sample Code	✓ RL78/G22, RL78/G23, RL78/G24 Firmware Update Module - Sample Code
	Firmware Updating Communications Module	✓ RX Family Firmware Updating Communications Module Using Firmware Integration Technology	✓ RL78/G23 Firmware Updating Communications Module
IoT Cloud OTA	Secondary device OTA Firmware Update Sample code	✓ RX65N Group Sample Code for OTA Update of a Secondary Device by Amazon Web Services with the Use of FreeRTOS	✓ RL78/G23 Sample Code for OTA Update of a Secondary Device by Amazon Web Services with the Use of FreeRTOS
	OTA firmware Update sample code	✓ RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)	✓ Getting Started Guide for Connecting Amazon Web Services in Wi-Fi Communication: RL78/G23-128p Fast Prototyping Board + FreeRTOS
	Development product	✓ QE for OTA: Development Assistance Tool for Firmware Update	

RENESAS

QE for OTA: Development Assistance Tool for Firmware Update

From Over the Air (OTA) to Local application
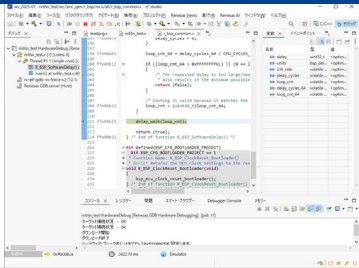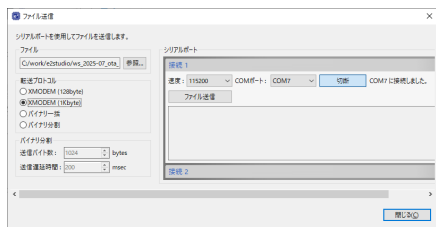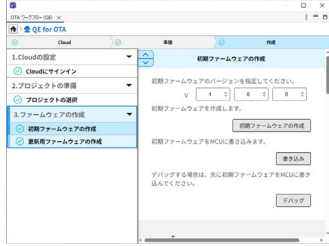**Firmware update with various system structure can be executed with simple GUI**
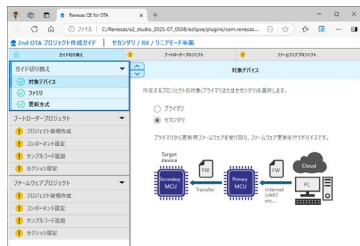
With/Without Cloud(AWS)

With Cloud
Wi-Fi / Ethernet

Without Cloud
Serial Communication

Update Target

Primary
FreeRTOS / non-OS

Secondary
non-OS

NEW from V.2.2.0

Update Method

Dual Bank
Bank0 — main
Bank1 — buffer

Partial update
Single Bank
main
buffer

Full Update
Single Bank
main

Reduce time required for 1 IoT device OTA by 86%! Support smooth PoC development!

Quick and Effective tool solution

QE for OTA

Create Firmware, Bootloader → Generate & implement Signature verification info → Execute Firmware Update → Debug

Firmware Update Structures supported by QE FOR OTA V.2.2.0

NEW
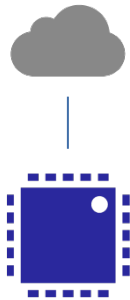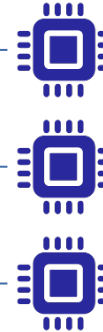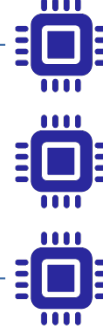
QE for OTA

Diagram labels: IoT Device | Primary — Secondary | Primary — Secondary

	IoT Device	Primary	Secondary NEW	Primary	Secondary NEW
Over the Air (OTA)	✓	✓		-	
Cloud Services	AWS	AWS		-	
Update target	IoT Device	Primary	Secondary	Primary	Secondary
Device	RX65N, RA6M5, RL78/G23	RX65N	RX23E-B, RX66T, RX660, RX261, RX140, RL78/G23	RX65N	RX23E-B, RX66T, RX660, RX261, RX140, RL78/G23
RTOS	FreeRTOS	FreeRTOS	non-OS	FreeRTOS, non-OS	non-OS
Update Method	RA, RX : Dual Bank Method RL78 : Partial Update Method		Dual Bank method (※), Partial Update Method, Full update method ※Supported by only dual bank supported device		
Communication method	Wi-Fi, Ethernet		Serial Communication Firmware Updating Communications Module (RX, RL78) NEW		