

S32K3_OTA_AB_SWAP_DEMOSISTRATION

Alvin Liu
2022-7-6



SECURE CONNECTIONS
FOR A SMARTER WORLD

CONFIDENTIAL & PROPRIETARY

NXP, THE NXP LOGO AND NXP SECURE CONNECTIONS FOR A SMARTER WORLD ARE TRADEMARKS OF NXP B.V.
ALL OTHER PRODUCT OR SERVICE NAMES ARE THE PROPERTY OF THEIR RESPECTIVE OWNERS. © 2020 NXP B.V.



S32K3XX: OTA Flash Features

- S32K3 extends the hardware capabilities for the OTA implementation enabling the concept of **active and passive blocks** and **read-while-write** capability between independent memory blocks.
- Each block size in all the S32K3 family is of 1MB and depending on the specific device you will have 1, 2, or 4 MB (S32K3x1, S32K3x2 or S32K3x4)
- The following table shows the main OTA features for some devices, we will use these concepts in the following sections.

Parameter	S32K311	S32K312/42	S32K344
pFlash size	1MB	2MB	4MB
pFlash block size	1MB	1MB	1MB
#pFlash blocks	2	2	4
RWW pFlash	Yes	Yes	Yes
A/B Swap support	Yes	Yes	Yes
A/B Swap resolution	512kB	1MB	2MB
HW Address translation	Yes	Yes	Yes

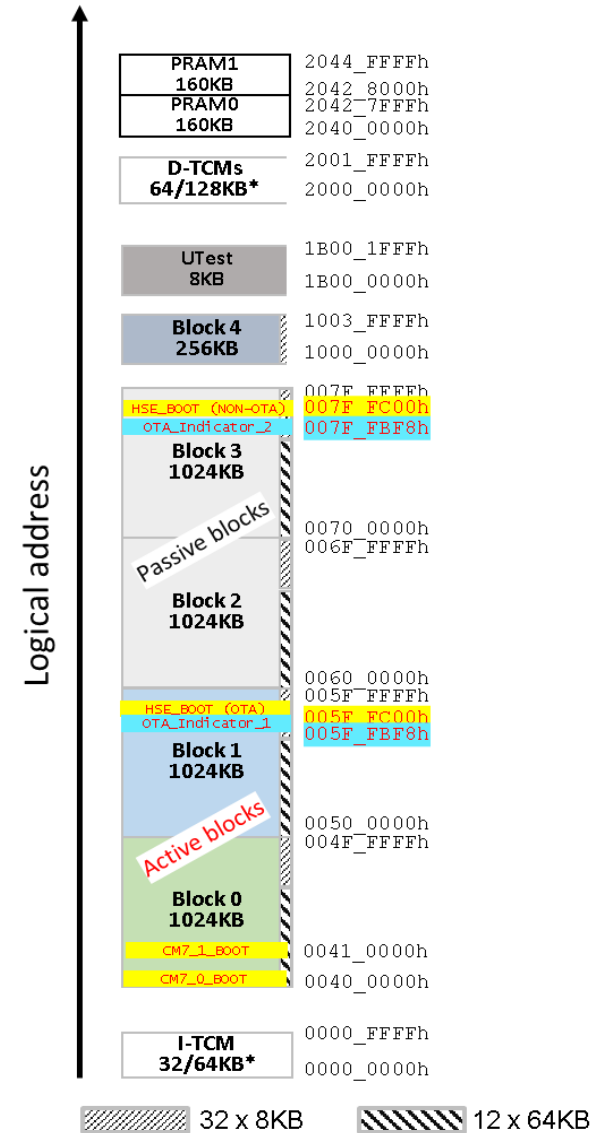
S32K3XX: OTA Flash Features

Active and passive blocks

- When **OTA is enable** in the part, device flash is divided in 2 X 2MB blocks (S32K344).
- **Active blocks** is the where the application code is located.
- **Passive blocks** is where the rollback image is located.

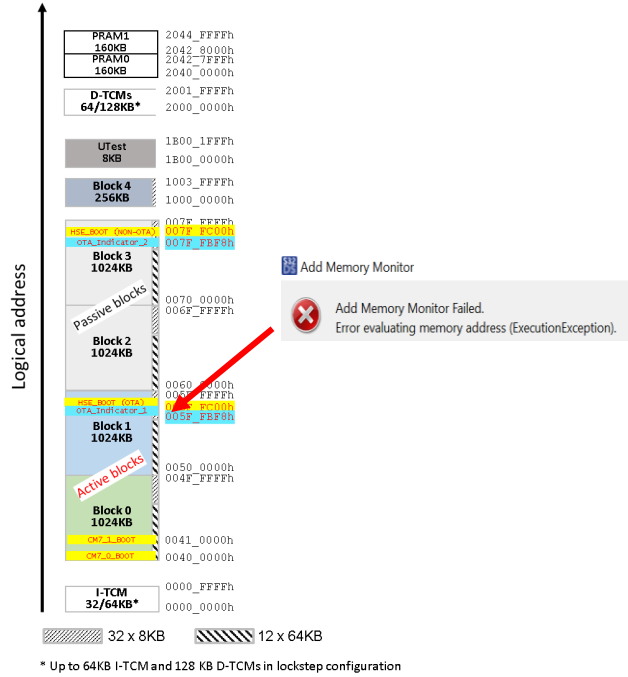
Flash Read-While-Write Functionality

- RWW, available between active and passive blocks, allows the flash to be updated while simultaneously executing code from it
- This feature allows for the firmware to be updated whilst the vehicle is in motion



* Up to 64KB I-TCM and 128 KB D-TCMs in lockstep configuration

S32K3XX: OTA Flash Features

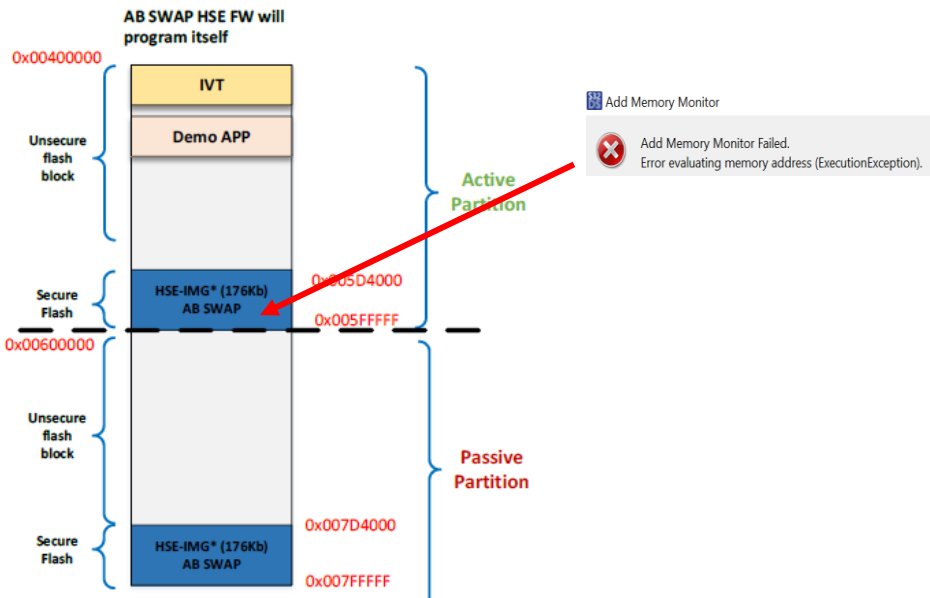


- **OTA enable parameter (OTA_ENABLE)**

- Activates flash swap hardware support (OTA_ENABLE = 0x6f74616374697665)
- Stored in UTest One-time programming (OTP) NVM sector
- Programmed by HSE core

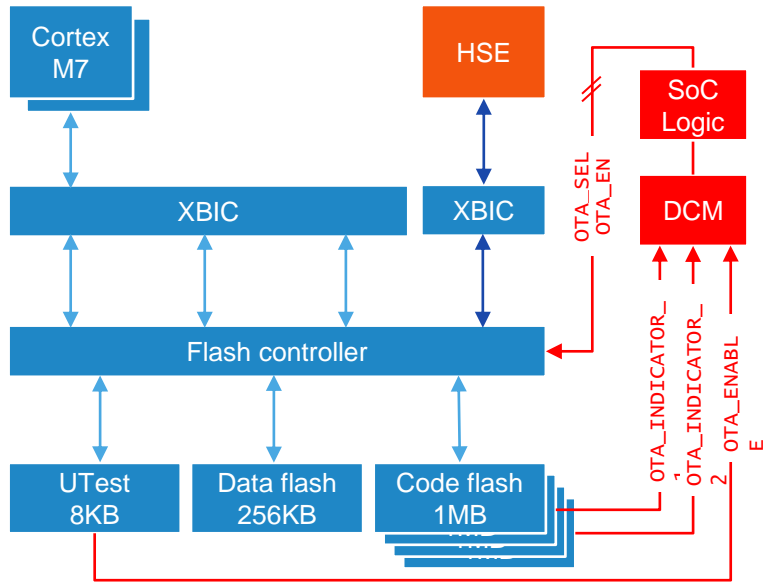
- **OTA indicators (OTA_INDICATOR_1 and OTA_INDICATOR_2)**

- Control mapping of flash blocks into logical address space
- Stored in active and passive code flash blocks (secure memory areas)
- Contains incremental and bitwise-complement sections to indicate firmware revision and firmware validity
- Can be programmed by HSE core; usually updated after new firmware is received and programmed into passive code flash blocks
- Flash blocks mapping defined by set of rules



S32K3XX: OTA Flash Features

- **Device Configuration Module (DCM)** reads NVM parameters after device reset
- **SoC Logic** decodes NVM parameters and generates signals to the flash controller (OTA_EN, OTA_SEL)
- **Flash blocks are mapped into logical address space by the flash controller prior any cores can start to execute!**



OTA_EN	OTA_SEL	Flash Block Select	Block Status
0	X	Block 0 (1MB)	Active
		Block 1 (1MB)	Active
		Block 2 (1MB)	Active
		Block 3 (1MB)	Active
1	0	Block 0 (1MB)	Active
		Block 1 (1MB)	Active
		Block 2 (1MB)	Passive
		Block 3 (1MB)	Passive
1	1	Block 2 (1MB)	Active
		Block 3 (1MB)	Active
		Block 0 (1MB)	Passive
		Block 1 (1MB)	Passive

Table 139: DCM register DCMSTAT (0x402AC000)

Bit #	Field Name	Description
18-34	-	Refer [REF02] for more details.
17	DCMOTAR	AB_SWAP (OTA) Active region (valid only if DCMDONE bit is set) 0b - Low address 1b - High address
16	DCMOTAA	AB_SWAP (OTA) Active State (valid only if DCMDONE bit is set) 0b - Inactive 1b - Active
10-15	-	Refer [REF02] for more details.
9	DCMOTAS	DCM AB_SWAP (OTA) Scanning Status (valid only when the value of the DCMDONE field is 1) 0b - Completed with errors 1b - Completed successfully
1-8	-	Refer [REF02] for more details.
0	DCMDONE	DCM Scanning Status Indicates whether the DCM scanning is in progress or complete. 0b - Running 1b - Completed

S32K3XX: OTA A/B SWAP

Logical Address:

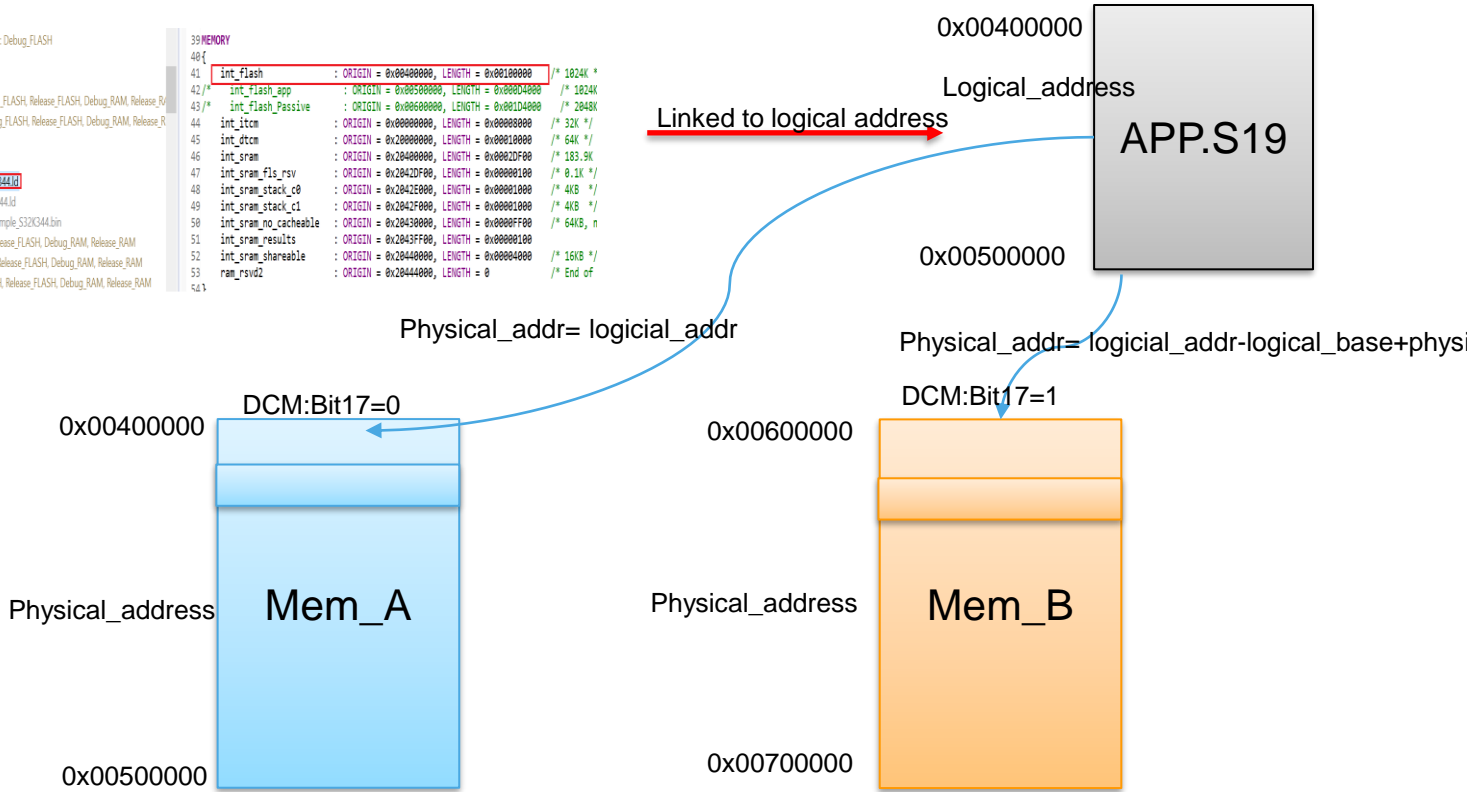
Memory area of code running in

Physical Address:

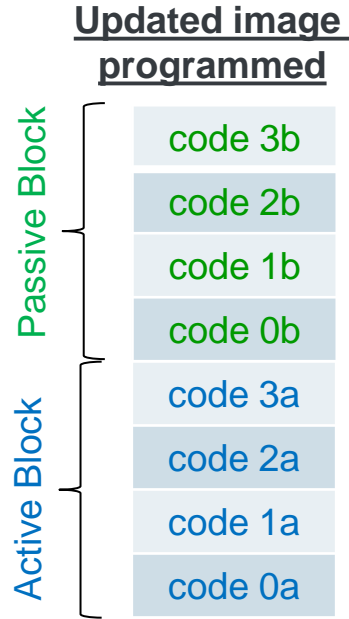
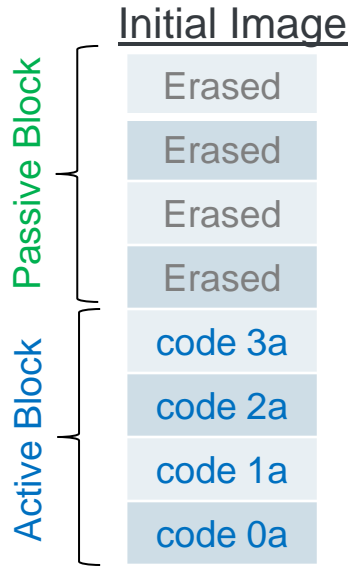
Memory area of code resident

```

39 MEMORY
40 {
41     int_flash : ORIGIN = 0x00400000, LENGTH = 0x00100000 /* 1024K */
42     int_flash_app : ORIGIN = 0x00500000, LENGTH = 0x00040000 /* 1624K
43     int_flash_passive : ORIGIN = 0x00600000, LENGTH = 0x001D4000 /* 2048K
44     int_itcm : ORIGIN = 0x00000000, LENGTH = 0x00000000 /* 32K */
45     int_dtcn : ORIGIN = 0x20000000, LENGTH = 0x00010000 /* 64K */
46     int_sram : ORIGIN = 0x20400000, LENGTH = 0x0002DF00 /* 183.9K
47     int_sram_fls_rsv : ORIGIN = 0x2042DF00, LENGTH = 0x00000100 /* 0.1K */
48     int_sram_stack_c0 : ORIGIN = 0x2042E000, LENGTH = 0x00001000 /* 4KB */
49     int_sram_stack_c1 : ORIGIN = 0x2042F000, LENGTH = 0x00001000 /* 4KB */
50     int_sram_no_cacheable : ORIGIN = 0x20430000, LENGTH = 0x0000FF00 /* 64KB, n
51     int_sram_results : ORIGIN = 0x2043FF00, LENGTH = 0x00000100
52     int_sram_shareable : ORIGIN = 0x20440000, LENGTH = 0x00004000 /* 16KB */
53     ram_rsvd2 : ORIGIN = 0x20444000, LENGTH = 0
54 }
    
```



S32K3XX: OTA A/B SWAP



Code logical addresses after reset and validation



Address Swap

Logical Address	Physical Address	Memory Content
0040_0000h-0040_FFFFh	0040_0000h-0040_FFFFh	Code0a
0041_0000h-0041_FFFFh	0041_0000h-0041_FFFFh	Code1a
0042_0000h-0042_FFFFh	0042_0000h-0042_FFFFh	Code2a
0043_0000h-0043_FFFFh	0043_0000h-0043_FFFFh	Code3a
0060_0000h-0060_FFFFh	0060_0000h-0060_FFFFh	Erased
0061_0000h-0061_FFFFh	0061_0000h-0061_FFFFh	Erased
0062_0000h-0062_FFFFh	0062_0000h-0062_FFFFh	Erased
0063_0000h-0063_FFFFh	0063_0000h-0063_FFFFh	Erased

Program passive block



Don't care current on A or B
Just operate passive in OTA

Logical Address	Physical Address	Memory Content
0040_0000h-0040_FFFFh	0040_0000h-0040_FFFFh	Code0a
0041_0000h-0041_FFFFh	0041_0000h-0041_FFFFh	Code1a
0042_0000h-0042_FFFFh	0042_0000h-0042_FFFFh	Code2a
0043_0000h-0043_FFFFh	0043_0000h-0043_FFFFh	Code3a
0060_0000h-0060_FFFFh	0060_0000h-0060_FFFFh	Code0b
0061_0000h-0061_FFFFh	0061_0000h-0061_FFFFh	Code1b
0062_0000h-0062_FFFFh	0062_0000h-0062_FFFFh	Code2b
0063_0000h-0063_FFFFh	0063_0000h-0063_FFFFh	Code3b

Active passive block



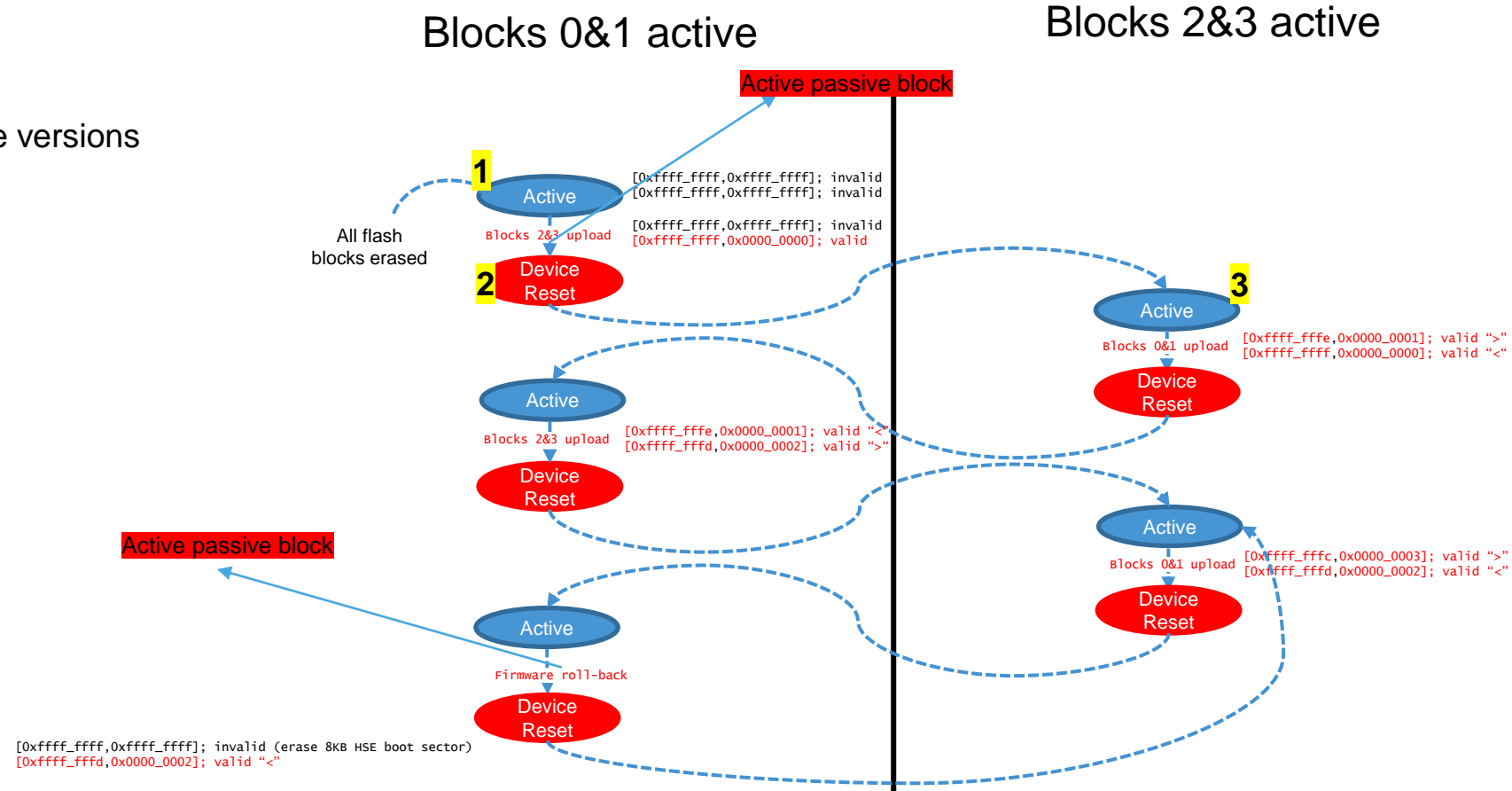
Call HSE service

Logical Address	Physical Address	Memory Content
0040_0000h-0040_FFFFh	0060_0000h-0060_FFFFh	Code0b
0041_0000h-0041_FFFFh	0061_0000h-0061_FFFFh	Code1b
0042_0000h-0042_FFFFh	0062_0000h-0062_FFFFh	Code2b
0043_0000h-0043_FFFFh	0063_0000h-0063_FFFFh	Code3b
0060_0000h-0060_FFFFh	0040_0000h-0040_FFFFh	Code0a
0061_0000h-0061_FFFFh	0041_0000h-0041_FFFFh	Code1a
0062_0000h-0062_FFFFh	0042_0000h-0042_FFFFh	Code2a
0063_0000h-0063_FFFFh	0043_0000h-0043_FFFFh	Code3a

S32K3XX: OTA A/B SWAP

Flash Swap

- Allows for instant switching between firmware versions
- Automatic firmware translation
- Instant version swap after device reset.
- Rollback capability.



1. Cores executes firmware from flash blocks 0&1 (active) after all flash blocks are erased.
2. After new image is uploaded to passive flash blocks 2&3 (OTA indicator updated) a device reset can be triggered.
3. After device reset, passive flash blocks 2&3 will become active, mapped at low address space and new firmware image will execute.

S32K3XX: HSE_AB_SWAP FW INSTALL

- The way to install HSE FW

- Activates Using S32DS + PE Multi-Link

S32DS Project "S32K344_HSE_FW_INSTALL_Demo_V110_ABSwap".

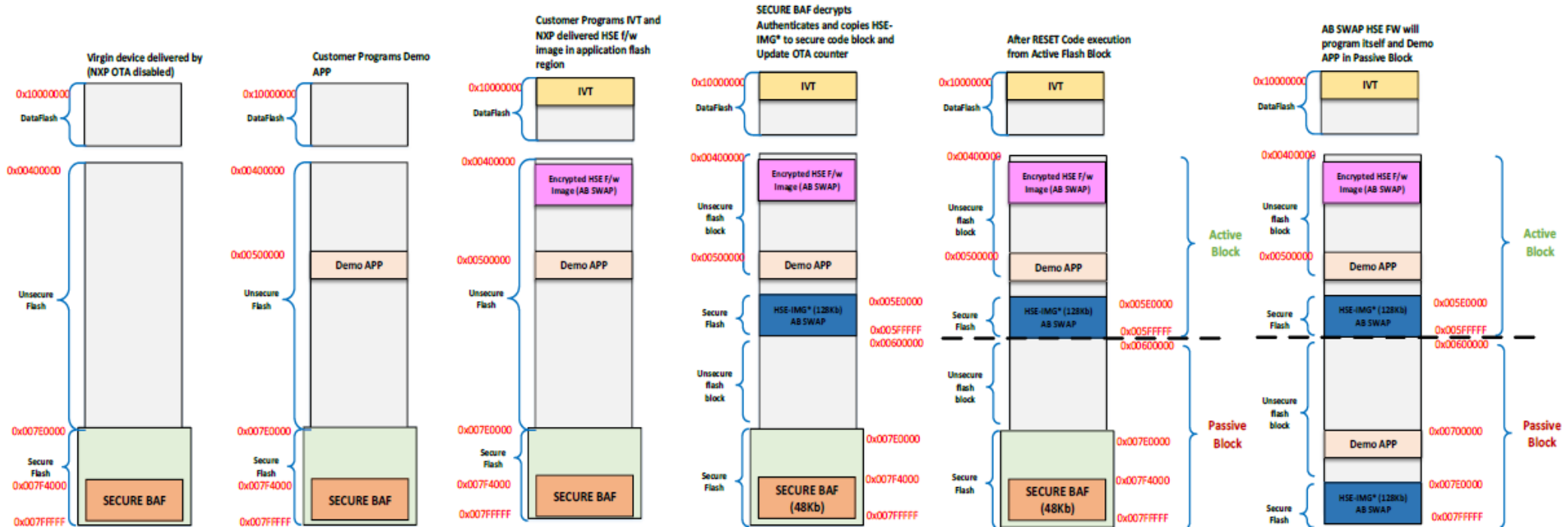
Guideline "Installing HSE Firmware FW to S32K344_readme.pdf"

Guideline "如何在EAR版本的SBAF片子上安装10之后的AB SWAP固件.pdf"

- Using LAB CMM script


Guideline C:\NXP\HSE_DEMOAPP_S32K3X4_0_1_1_0\HSE_DEMOAPP_S32K3X4_0_1_1_0_ReadMe.pdf

address	0	4	8	C
SD:1B000280	74697665	6F746163	FFFFFFFF	FFFFFFFF
SD:1B000290	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF






S32K3XX:OTA DEMO

- Get OTADEMO Package from NXP website([SW32K3_OTADEMO_0.8.0_D2203](#))

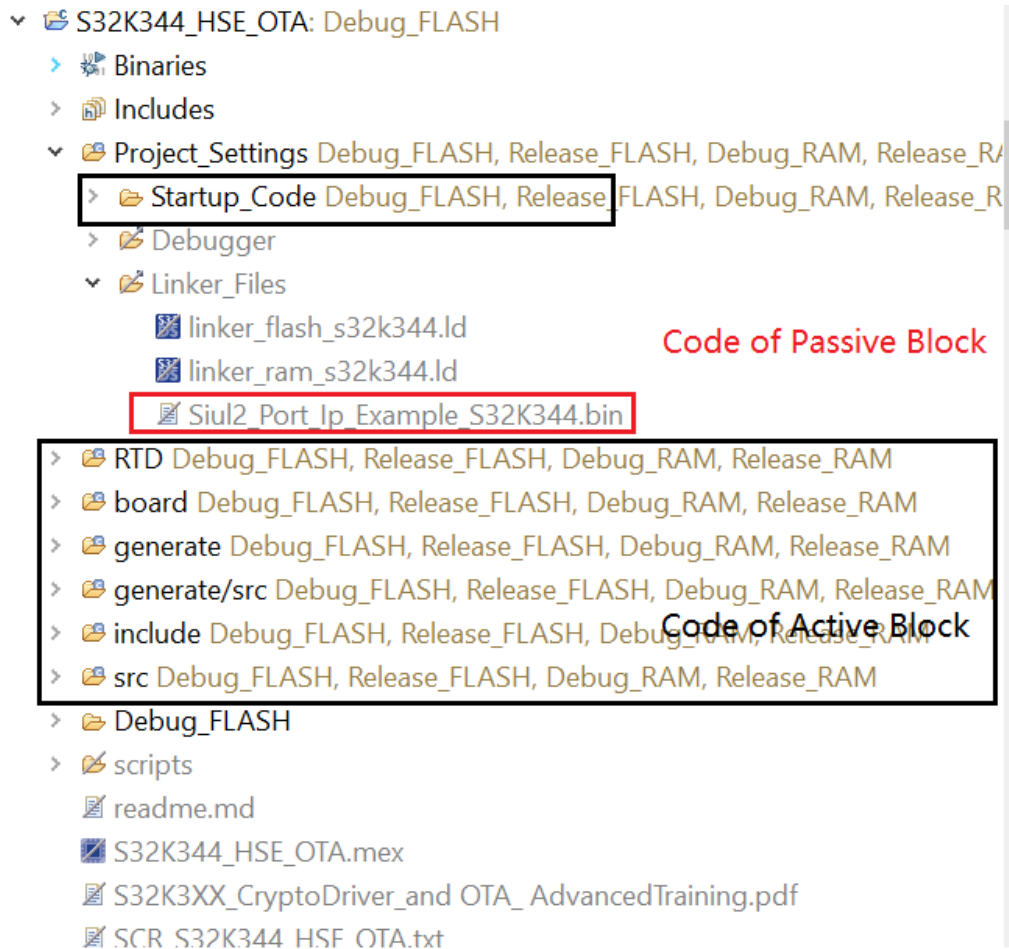
Show All Files 

3 Files

+	File Description	File Size	File Name
+	SCR_S32K344_HSE_OTA.txt	390 bytes	 SCR_S32K344_HSE_OTA.txt
+	SW32K3_OTADEMO_0.8.0_D2203.exe	1.7 MB	 SW32K3_OTADEMO_0.8.0_D2203.exe
+	SW32K3_OTADEMO_0.8.0_ReleaseNotes.pdf	162.1 KB	 SW32K3_OTADEMO_0.8.0_ReleaseNotes.pdf

S32K3XX:OTA DEMO

- Import the project “S32K344_HSE_OTA” to S32DS

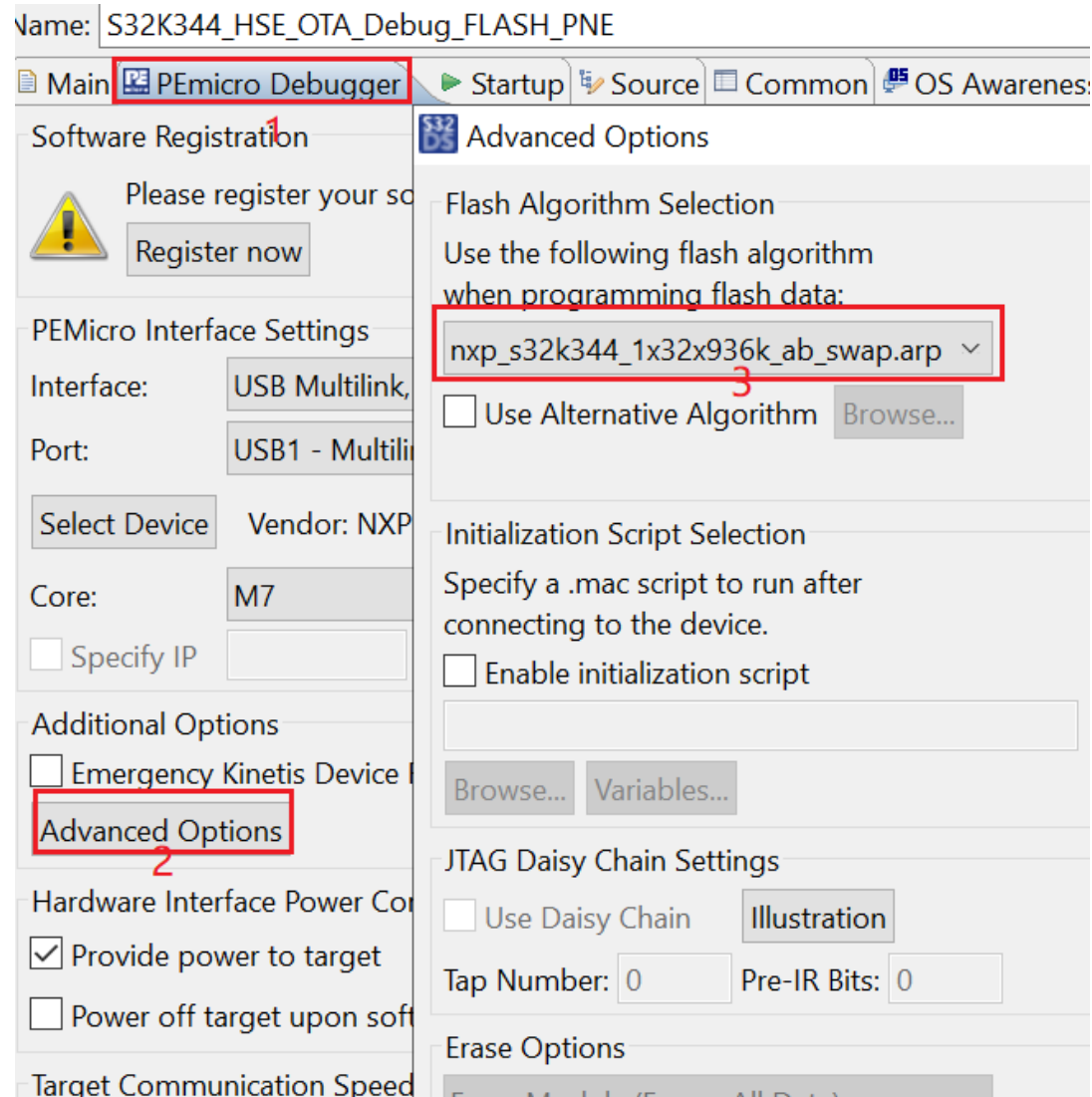


Project tree for S32K344_HSE_OTA: Debug_FLASH:

- Binaries
- Includes
- Project_Settings Debug_FLASH, Release_FLASH, Debug_RAM, Release_RAM
 - Startup_Code Debug_FLASH, Release_FLASH, Debug_RAM, Release_RAM
 - Debugger
 - Linker_Files
 - linker_flash_s32k344.ld
 - linker_ram_s32k344.ld
 - Siul2_Port_Ip_Example_S32K344.bin
- RTD Debug_FLASH, Release_FLASH, Debug_RAM, Release_RAM
- board Debug_FLASH, Release_FLASH, Debug_RAM, Release_RAM
- generate Debug_FLASH, Release_FLASH, Debug_RAM, Release_RAM
- generate/src Debug_FLASH, Release_FLASH, Debug_RAM, Release_RAM
- include Debug_FLASH, Release_FLASH, Debug_RAM, Release_RAM
- src Debug_FLASH, Release_FLASH, Debug_RAM, Release_RAM
- Debug_FLASH
- scripts
- readme.md
- S32K344_HSE_OTA.mex
- S32K3XX_CryptoDriver_and OTA_AdvancedTraining.pdf
- SCR_S32K344_HSF_OTA.txt

Code of Passive Block

Code of Active Block

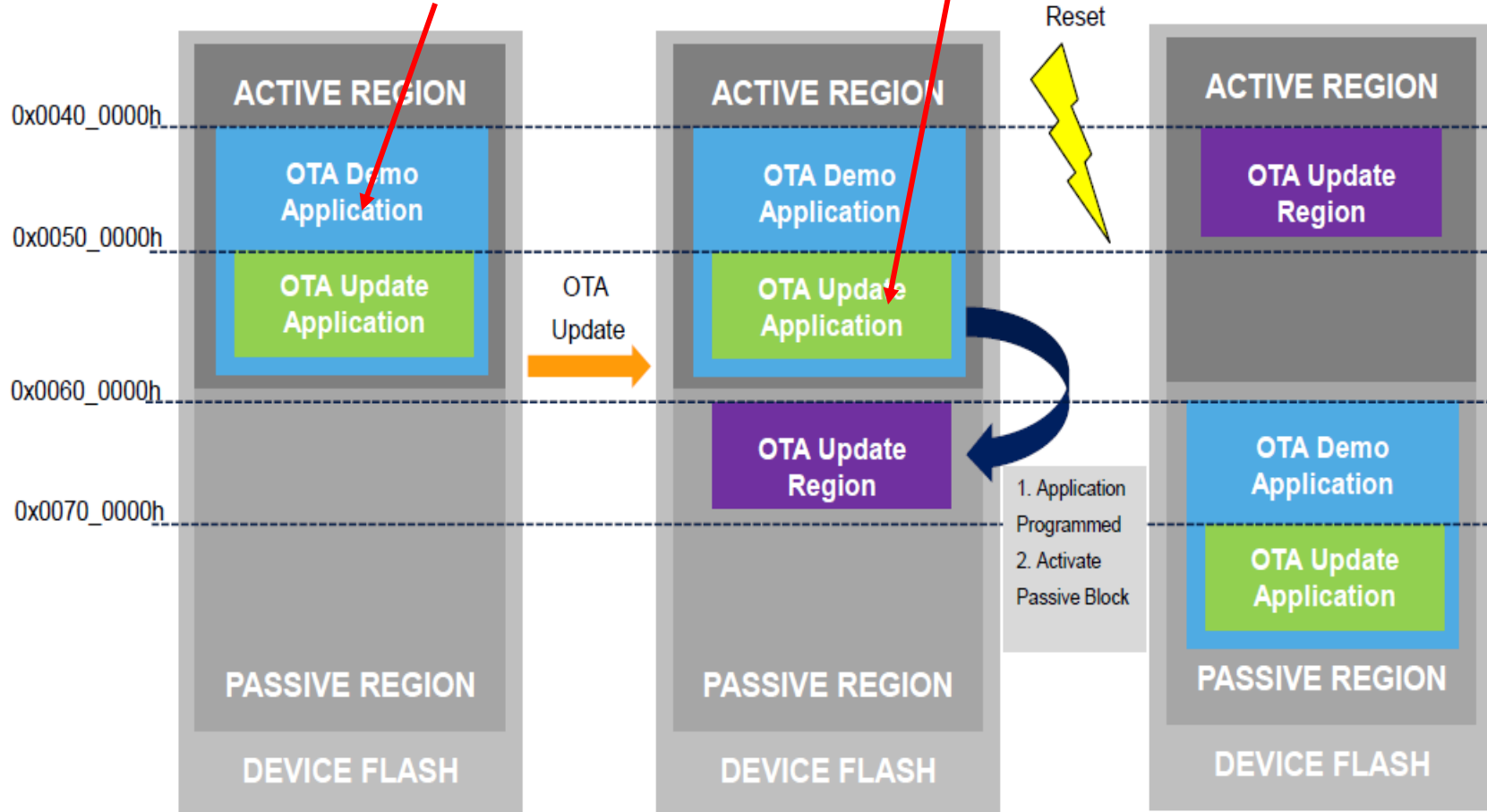


PEMicro Debugger Advanced Options dialog:

- Name: S32K344_HSE_OTA_Debug_FLASH_PNE
- PEMicro Interface Settings
 - Interface: USB Multilink
 - Port: USB1 - Multilink
 - Select Device
 - Vendor: NXP
 - Core: M7
 - Specify IP
- Additional Options
 - Emergency Kinetis Device
 - Advanced Options
- Hardware Interface Power Control
 - Provide power to target
 - Power off target upon soft reset
- Target Communication Speed
- Advanced Options
 - Flash Algorithm Selection
 - Use the following flash algorithm when programming flash data:
 - nxp_s32k344_1x32x936k_ab_swap.arp
 - Use Alternative Algorithm
 - Initialization Script Selection
 - Specify a .mac script to run after connecting to the device.
 - Enable initialization script
 - JTAG Daisy Chain Settings
 - Use Daisy Chain
 - Tap Number: 0
 - Pre-IR Bits: 0
 - Erase Options

S32K3XX:OTA DEMO

- OTA Update **S32K344_HSE_OTA** **Siul2_Port_Ip_Example_S32K344**



```

void Demo_OtaUpdate(void)
{
    hseSrvResponse_t    HseResponse;

    /* =====
    /* Update passive blocks
    /* =====
    Update_Passive();

    /* =====
    /* Request A/B Swap to passive blocks
    /* =====
    HseResponse = HSE_ActivatePassiveBlock(MU0_INSTANCE_U8);
    App_SetSuccessStatus(HSE_SRV_RSP_OK == HseResponse);
}
    
```



SECURE CONNECTIONS
FOR A SMARTER WORLD