TN1255
Technical note

How to disable the sealing in Test Mode

Introduction

Devices, starting from the 55nm families, have many security mechanisms to guarantee the information integrity and privacy of data saved within the device. The user configures these mechanisms depending on the security level of the application.

On the other hand, in case the users claim an unexpected behavior of a sample, they can send this sample to the silicon manufacturer to analyze the failure. One of the actions of the failure analysis is to move the flash blocks to "Test Mode" to perform multiple tests to verify their integrity.

Before sending the sample to ST Microelectronics®, the user moves the sample to the Failure Analysis lifecycle and configures what Flash blocks ST can access.

In the current implementation, most of the Flash memory protections (for example PASS and Censorship) – typically used while the FLASH works in user mode – don’t work in Test Mode.

The risk is that the silicon manufacturer can access the entire content of the Flash once the Flash is in Test Mode. The Flash, however, can contain sensitive data that the user doesn’t want to disclose to the silicon manufacturer. For this reason, ST introduced a new security feature, for example the Sealing.

The sealing avoids that ST can move some Flash blocks to Test Mode. The user specifies which these Flash blocks are.

This TN gives some hints about how the user can configure the sealing for each block of the Flash memory.

The example reported in the document is focused on SPC58ECxx, but all concepts are valid also for other 55nm and 40nm devices (it makes exception the memory map different from device to device).

Contents

List of tables

List of figures

1 Overview

Security and Testability have two different goals that can be in contrast:

- Security:
  - limit or block the access to some resources that may disclose sensitive details
- Testability:
  - access all resources to verify their integrity

These 2 properties don't match when the ST needs to analyze the MCU due to a failure. Typically, in this case, ST gains the full access[a] of the flash by moving it into Test mode. On the other hand, the user wants to protect the content of one or more blocks of the flash.

There are different protection layers that allow the user to configure what ST can do while the life cycle of the device is in Failure Analysis:

- MCU life cycle
- Sealing
- eFuse
- Data protection during FA
- MCU disabling after FA

Sealing prevents ST to move Flash blocks to Test Mode. In this way, the content of these blocks is not disclosed to ST. On the other hand, under these circumstances, the silicon provider cannot perform Failure Analysis on these blocks.

By default, all Flash blocks are sealed but the user has the possibility to disable this mechanism in some blocks of the Flash.

1.1 Device Life Cycle vs. Flash Test Mode

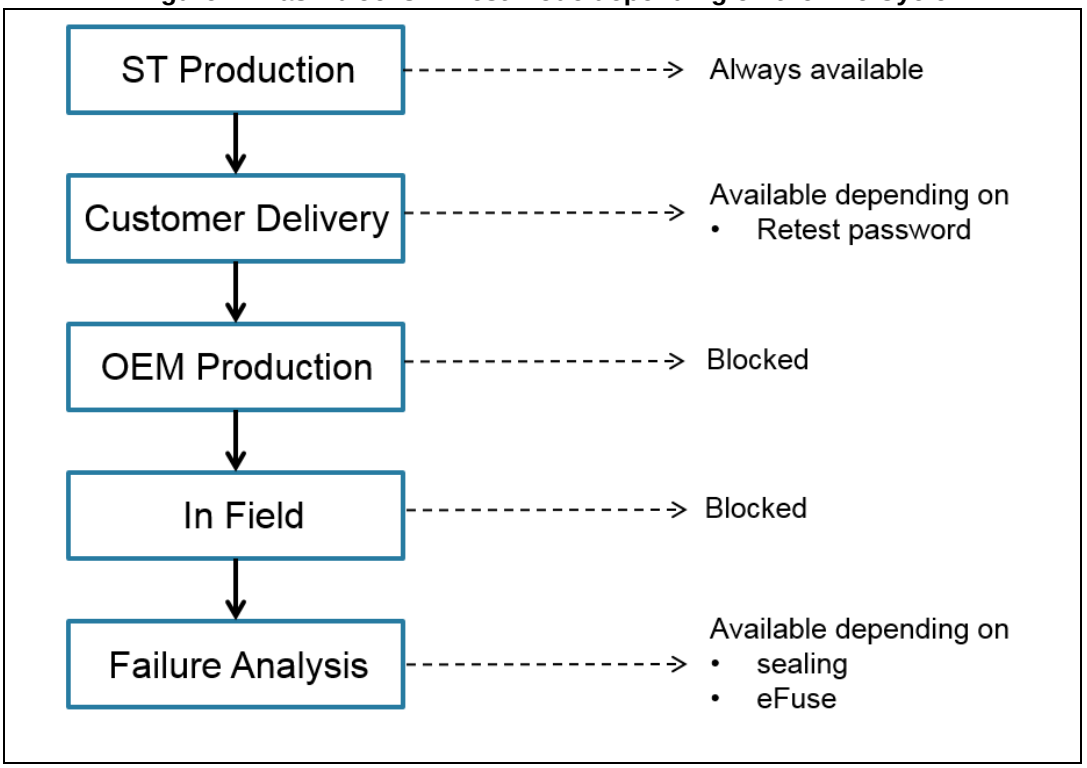
The silicon provider can move a Flash block to Test Mode according to different parameters, for example the life cycle, the sealing and eFuse [b]configurations. Flash blocks in Test Mode depending on the Life Cycle shows the dependency between life cycle and Test Mode activation.

ST can move Flash blocks in Test Mode only if the life cycle is either Customer Delivery or Failure Analysis.

a. In this context, full access means the capability to read, program and erase blocks of the Flash.

b. eFuse is a mechanism that gives evidence of the activation of Flash Test Mode that is obtained by a physical damage. This avoid the possibility of refurbished samples.

Figure 1. Flash blocks in Test Mode depending on the Life Cycle



1.2 UTEST and Sealing

When the device is in Failure Analysis, ST can move the Flash to Test Mode depending on the sealing configuration.

By default the sealing is disabled[(c)]. To enable this protection, the user has to program two locations of the UTEST:

1. Test Mode Disable Seal @ 0x00400040 (32 bit):
- by this password: 0x5A4B3C2D, and
2. Test Mode Disable Block
   a) Select Group A @ 0x00400050 (128 bit)
   b) Select Group B @ 0x00400060 (128 bit)

Group A and Group B are logically ANDed to determine the blocks to be protected:

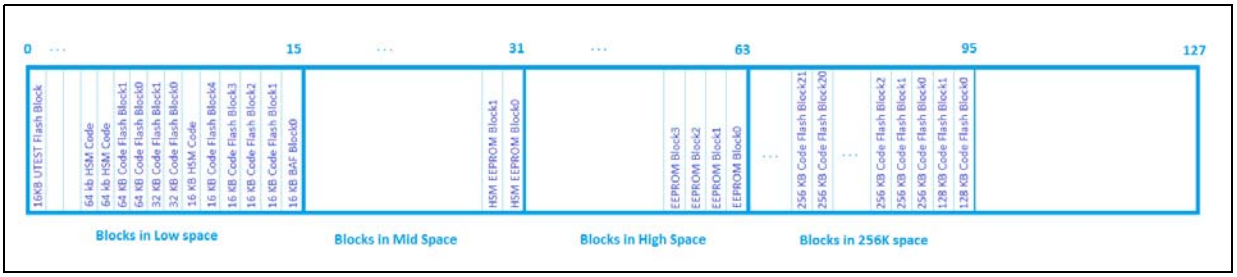
- Bit ‘1’ means that in the correspondent flash block, the sealing is disabled in Test Mode (block not protected)
- Bit ‘0’ means that in the correspondent flash block, the sealing is enabled in Test Mode (block protected)

The user can program Group A as first step. Afterwards, they can disable the sealing protection of other flash blocks by programming Group B.

c. In user mode, equivalent protection of the Flash Memory are provided by PASS and TDM module.

The flags of the Select Group A and B are aligned with the LOCK registers (see *Figure 2*). Since the UTEST is OTP, the user cannot modify these configuration.

Figure 2. Select block mapping



At least, there is another location in the Utest dedicated to the sealing. This is:

- Test Mode Disable Override Passcode @ 0x0040030 (32 bit)

Customer can provide this password to the manufacturer in order to temporally bypass this protection. The manufacturer pass it in the Test Mode Disable Password register (TMD) of the Flash Memory interface before entering in Test Mode. After that the sealing protection is temporally removed until next power on reset.

1.3 Example of Test Mode Disable sealing

Table 1 shows an example of configuration of the sealing. Some Flash blocks are sealed and others are not.

Table 1. List of the Flash Block sealed

Flash				Sealed
Item	Start	End	Blocks	
1	0x00FF0000	0x00FFFFFF	64 KB Low Flash block1	y
2	0x01000000	0x0101FFFF	128 KB Flash block0	n
3	0x01020000	0x0103FFFF	128 KB Flash block1	n
4	0x01040000	0x0107FFFF	256 KB Flash block0	n
5	0x01080000	0x010BFFFF	256 KB Flash block1	n
6	0x010C0000	0x010FFFFF	256 KB Flash block2	n
7	0x01100000	0x0113FFFF	256 KB Flash block3	n
8	0x01140000	0x0117FFFF	256 KB Flash block4	n
9	0x01180000	0x011BFFFF	256 KB Flash block5	n

Table 1. List of the Flash Block sealed (continued)

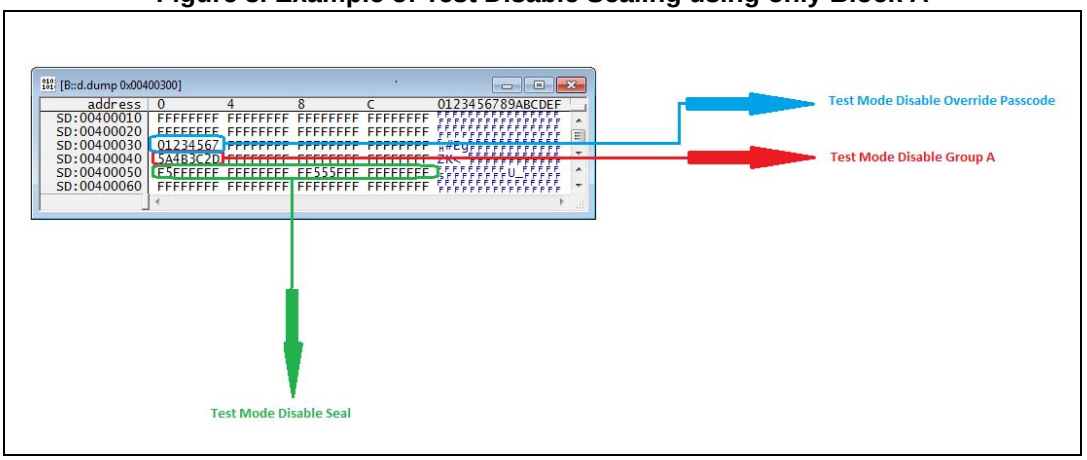
Flash				Sealed
Item	Start	End	Blocks	
10	0x011C0000	0x011FFFFF	256 KB Flash block6	n
11	0x01200000	0x0123FFFF	256 KB Flash block7	n
12	0x01240000	0x0127FFFF	256 KB Flash block8	n
13	0x01280000	0x012BFFFF	256 KB Flash block9	n
14	0x012C0000	0x012FFFFF	256 KB Flash block10	n
15	0x01300000	0x0133FFFF	256 KB Flash block11	y
16	0x01340000	0x0137FFFF	256 KB Flash block13	n
17	0x01380000	0x013BFFFF	256 KB Flash block14	y
18	0x013C0000	0x0133FFFF	256 KB Flash block15	n
19	0x01400000	0x0143FFFF	256 KB Flash block16	y
20	0x01440000	0x0147FFFF	256 KB Flash block17	n
21	0x01480000	0x018BFFFF	256 KB Flash block18	y
22	0x014C0000	0x014FFFFF	256 KB Flash block19	n
23	0x01500000	0x0153FFFF	256 KB Flash block20	y
24	0x01540000	0x0157FFFF	256 KB Flash block21	n
25	0x01580000	0x015BFFFF	256 KB Flash block22	y
26	0x00620000	0x0062FFFF	64 KB HSM block	y

The steps to disable the sealing protection are:

1. To program the Test Mode Disable Seal
2. Program the Test Mode Disable for Block A[d]
3. Move the life cycle versus FA

Figure 3 shows the portion of UTEST memory related to the sealing programming.

Figure 3. Example of Test Disable Sealing using only Block A



According to the table reported in *Figure 3* and considering the Select Block Mapping of Select block mapping, if the users want to seal other locations, for example EEPROM Block 0, they have to program the Group B[e].

```
@ 0x000400030 -> 0xFFFFFFFF_FFFFFFFE_FFFFFFFF_FFFFFFFF
```

Moreover, at location 0x40000030 there is the passcode to pass to TMD register to disable the sealing until the next POR.

d. Since Test Mode Disable Block Group A and Group B are ANDed, it's enough to program the Test Mode Disable Block Group A to disable the sealing of some flash blocks.

e. This is possible because Group A and Group B are logically ANDed.

2 Summary

This document describes how the user can disable the Sealing protection that is enabled by default. This protection deactivates the possibility of moving the Flash in Test Mode while the life cycle of the sample is in Failure Analysis.

Using this protection, the users can secure sensitive data that are saved within Flash blocks. It is worth noticing that, under these circumstances, the silicon vendor cannot test the integrity of the sealed Flash blocks in case Failure Analysis.

Appendix A Further information

In this section is provided the Lauterbach script related to the example reported in Chapter.

```
AREA.RESET
AREA.CREATE log
AREA.SELECT log
AREA.VIEW log

DIALOG.YESNO "Flash programming prepared. Program new DCF now?"
ENTRY &progflash

&UTEST_TEST_MODE_DISABLE_SEAL=0x00400040
&current_address=&UTEST_TEST_MODE_DISABLE_SEAL

IF &progflash
	(
		;Lock0 ->TSLock enable UTEST memory
		PER.S ANC:0xF7FE0010 %LONG 0x3FFFFFFF
		print "UTest Unlocked"
		;================================================
		print "Test mode disable seal"
		;1 - TEST mode disable password 0x5A4B_3C2D. ;32 bit
		; this the PASSCODE protected: customer create a password to
enable manufacturer access into Flash test mode
		GOSUB program_word &current_address 0x5A4B3C2D
		&current_address=&current_address+0x10 ;0x0400_0050
		;================================================
		print "Test mode disable block select group A"
		;2 - Test mode disable block select group A 0x F5FF_FFFF
0xFFFF_FFFF 0xFF55_5FFF 0xFFFF_FFFF
		GOSUB program_word_128 &current_address 0xF5FFFFFFFFFFFFFF
0xFF555FFFFFFFFFFF
	)
	ELSE
	(
		FLASH.List
	)
	SYStem.BdmClock 4MHz
ENDDO

program_word:
	entry &address &data
	;MCR->PGM =1 enable program memory
	PER.S ANC:0xF7FE0000 %LONG 0x610
	D.S EA:(&address) %BE %LONG (&data)
	print "written = 0x" &address
	PER.S ANC:0xF7FE0000 %LONG 0x611 ;MCR->EHV=1 program memory
	WHILE ((Data.Long(ea:0xF7FE0000)&0x0200)==0);     ;while
(FLASH.MCR.B.DONE == 0);
```

```
        PER.S ANC:0xF7FE0000 %LONG 0x610   ;MCR->EHV=0 program memory
        PER.S ANC:0xF7FE0000 %LONG 0x600;MCR->PGM =0
        RETURN

program_word_128:
        entry &address &data &data1
;MCR->PGM =1 enable program memory
PER.S ANC:0xF7FE0000 %LONG 0x610
        D.S EA:(&address) %BE %QUAD (&data)
        print "written 0x" &data " at 0x" &address
        D.S EA:(&address+0x8) %BE %QUAD (&data1)
        print "written 0x" &data " at 0x" &address
        PER.S ANC:0xF7FE0000 %LONG 0x611  ;MCR->EHV=1 program memory
        WHILE ((Data.Long(ea:0xF7FE0000)&0x0200)==0);     ;while
(FLASH.MCR.B.DONE == 0);
        PER.S ANC:0xF7FE0000 %LONG 0x610   ;MCR->EHV=0 program memory
        PER.S ANC:0xF7FE0000 %LONG 0x600;MCR->PGM =0
RETURN
```

Here another script to Override PassCode.

```
AREA.RESET
AREA.CREATE log
AREA.SELECT log
AREA.VIEW log

DIALOG.YESNO "Flash programming prepared. Program new DCF now?"
ENTRY &progflash

&UTEST_TEST_OVERRIDE_PW=0x00400030
&current_address=&UTEST_TEST_OVERRIDE_PW

IF &progflash
        (
                ;Lock0 ->TSLock enable UTEST memory
                PER.S ANC:0xF7FE0010 %LONG 0x3FFFFFFF
                print "UTest Unlocked”
;================================================
print "Test mode disable seal"
                ;1 - TEST mode disable password 0x0123_4567. ;32 bit
                ; this the PASSCODE protected: customer create a password to
enable manufacturer access into Flash test mode
                GOSUB program_word &current_address 0x01234567


        )
        ELSE
        (
                FLASH.List
        )
        SYStem.BdmClock 4MHz
ENDDO
```

```
program_word:
       entry &address &data
       ;MCR->PGM =1 enable program memory
       PER.S ANC:0xF7FE0000 %LONG 0x610
       D.S EA:(&address) %BE %LONG (&data)
       print "written = 0x" &address
       PER.S ANC:0xF7FE0000 %LONG 0x611  ;MCR->EHV=1 program memory
       WHILE ((Data.Long(ea:0xF7FE0000)&0x0200)==0);     ;while
(FLASH.MCR.B.DONE == 0);
       PER.S ANC:0xF7FE0000 %LONG 0x610   ;MCR->EHV=0 program memory
       PER.S ANC:0xF7FE0000 %LONG 0x600;MCR->PGM =0
RETURN
```

A.1 Reference documents

SPC584Cx/SPC58ECx 32-bit MCU family built on the Power Architecture® for automotive body electronics applications (RM0707, DocID028117).

Table 2. Acronyms

Acronym	Name
FA	Failure Analysis
TM	Test Mode
OTP	One Time Programming
POR	Power On Reset

Revision history

Table 3. Document revision history

Date	Revision	Changes
26-Mar-2019	1	Initial release.