AN13321

TPMS FLASH Protection and Security

Rev. 1 — 22 July 2021 Application note

Document information

Information	Content
Keywords	FXTH87, FXTH87E, NTM88, FLASH block protection, security, TPMS
Abstract	This application note addresses the FLASH block protection and security for tire pressure monitoring systems.

Revision history

Rev	Date	Description
1	20210722	• Initial release

1 Introduction

The TPMS FLASH Memory Controller module offers users the ability to enable FLASH protection and security.

The block protection feature prevents the protected region of FLASH from program or erase changes.

The security feature prevents unauthorized access to the contents of FLASH and RAM memory.

2 Compatible part numbers

The information provided in this document is compatible with all devices of the FXTH family implementing library-based applications and all devices of the NTM88 family.

FXTH applications that are firmware-based do not provide a direct access to the non-volatile protection and security registers located in the embedded firmware section. For firmware-based applications, FLASH protection can be configured using the firmware function TPMS_FLASH_PROTECTION, and security is not supported.

For more information on firmware versus library model, refer to AN12523[1].

A complete description of the FLASH block protection and security is provided in the user manuals. See Section 5 "References".

3 FLASH block protection

3.1 Principle

When protection is enabled, no write or erase is allowed in the protected FLASH section shown in Figure 1, even from the program itself. The FLASH block protection feature protects boot loader code from being erased or overwritten during the Over The Air reprogramming procedure, ensuring that its integrity is always maintained even if the reprogramming procedure does not execute as expected.

The TPMS FLASH memory is divided into 32 pages of 512 bytes. When enabled, block protection begins at any 512-byte boundary below the last address of FLASH, 0xFFFF.

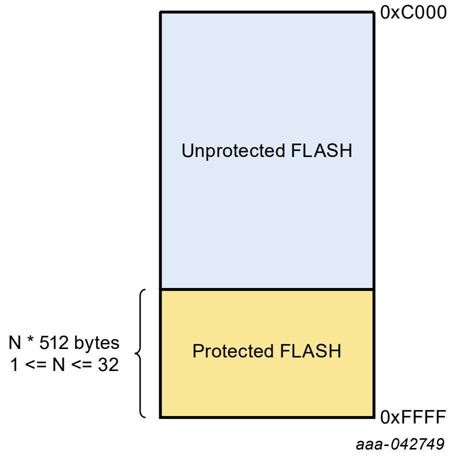


Figure 1. FLASH protected when protection is enabled

The protection settings are located in FPROT register, see Figure 2. After exit from reset, FPROT is loaded with the contents of the NVPROT location which is in the non-volatile register block of the FLASH memory. FPROT cannot be changed directly from application software so a runaway program cannot alter the block protection settings. Because NVPROT is within the last 512 bytes of FLASH, if any amount of memory is protected, NVPROT is itself protected and cannot be altered (intentionally or unintentionally) by the application software. FPROT can be written through BACKGROUND DEBUG commands which allow a way to erase and reprogram a protected FLASH memory.

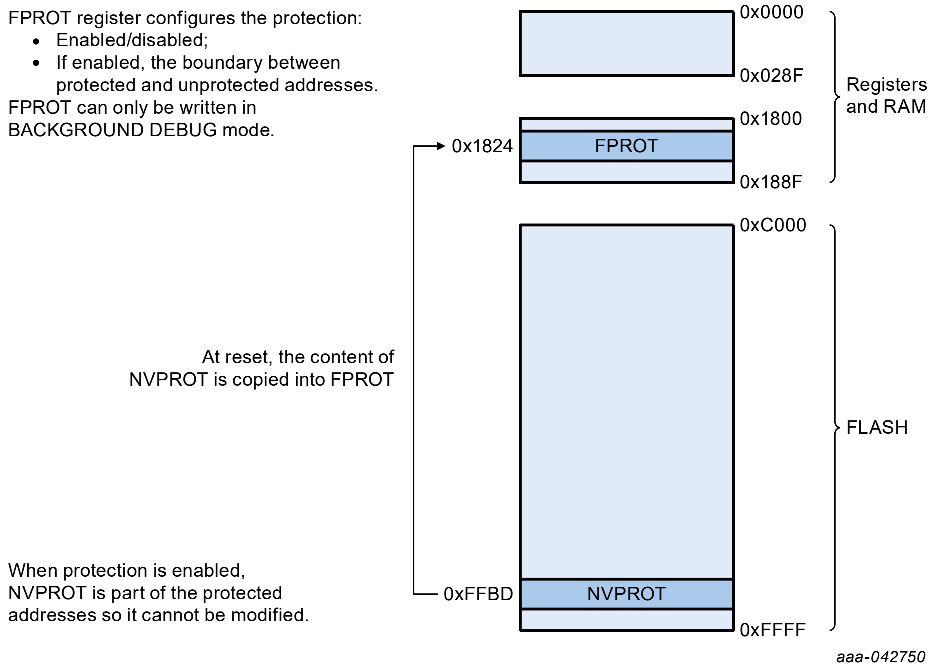

FPROT register configures the protection:
• Enabled/disabled;
• If enabled, the boundary between protected and unprotected addresses.
FPROT can only be written in BACKGROUND DEBUG mode.
0x0000
0x028F
Registers and RAM
0x1800
0x1824
FPROT
0x188F
0xC000
At reset, the content of NVPROT is copied into FPROT
FLASH
When protection is enabled, NVPROT is part of the protected addresses so it cannot be modified.
0xFFBD
NVPROT
0xFFFF
aaa-042750

Figure 2. Protection registers

3.2 Register configuration

The FPROT/NVPROT register fields are shown in Table 1.

Table 1. FPROT/NVPROT register fields

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
FPS7	FPS6	FPS5	FPS4	FPS3	FPS2	FPS1	FPDIS

When FPDIS bit is 1, protection is disabled. When FPDIS bit is 0, protection is enabled and the range of addresses is configured with the FPS7:1 bits, which correspond to the seven upper bits of the last unprotected address in FLASH.

At production, NXP programs NVPROT to 0xFF, which disables the protection.

Example: we want to enable FLASH protection on the address range 0xE000 – 0xFFFF. The last unprotected address is 0xDFFF = 0b**1101111**111111111. The seven upper bits highlighted in bold must be written in FPS7:1, and FPDIS must be set to 0. This gives NVPROT = 0b**1101111**0 = 0xDE.

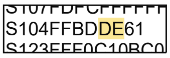
3.3 Enabling protection

NVPROT register can be written in FLASH:

- By the user during programming. The following declaration can be written in the source code:

```
volatile const UINT8 FLASH_NVPROT @0xFFBD = 0xDE;
```

As a result, the value to be written at address 0xFFBD appears in the s19 file:



```
S107FDFCFFFFFF
S104FFBDDE61
S123FFF0C10BC0
```

aaa-042751

Figure 3. s19 extract showing the value to be written at 0xFFBD

- At runtime by the application using the library function TPMS_FLASH_WRITE. The program can execute the following instructions:

```
u8NewNVPROT = 0xDE;
TPMS_FLASH_WRITE(0xFFBD, (const UINT8*)&u8NewNVPROT, 1u);
```

After NVPROT has been written, protection is enabled after a reset occurs. STOP1 exit is considered a reset source.

3.4 Disabling protection

Protection can be disabled in BACKGROUND DEBUG MODE only by writing 1 to FPDIS in FPROT register. Once protection is disabled, NVOPT register can be erased in order to disable protection permanently.

4 Security

4.1 Principle

When security is engaged, the MCU resources are separated between secure and unsecure resources as shown in Figure 4. A program executing within secure memory has normal access to both secure and unsecure memory resources. A program executing from an unsecure memory location, via the BACKGROUND DEBUG controller or via the NTM88 hardware SPI, cannot access secure locations: writes are ignored and reads return all 0s.

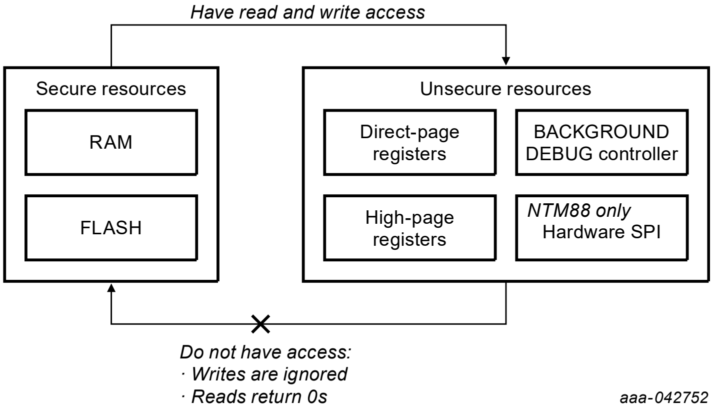


Figure 4. Secure and unsecure MCU resources when security is engaged

Security is engaged or disengaged based on the value of FOPT register. During reset, the contents of the non-volatile location NVOPT are copied from FLASH into the working FOPT register in high-page register space.

The user has the option to allow or disallow a security unlocking mechanism through an 8-byte backdoor security key written by the user in NVBACKKEY registers. When unlocking security via the backdoor key, security is temporarily disengaged until the next reset occurs. If the security unlocking mechanism is not allowed or if the user wants to disable security permanently, the only option is to proceed to a mass erase of the chip in order then to write to the NVOPT register with a value that maintains security disengaged.

Note: *The trim coefficients are erased during the mass erase operation. The trim coefficients are unique to each device so they cannot be copied from one device to another. If the trim coefficients are not present in the device then the sensors are not functional.*

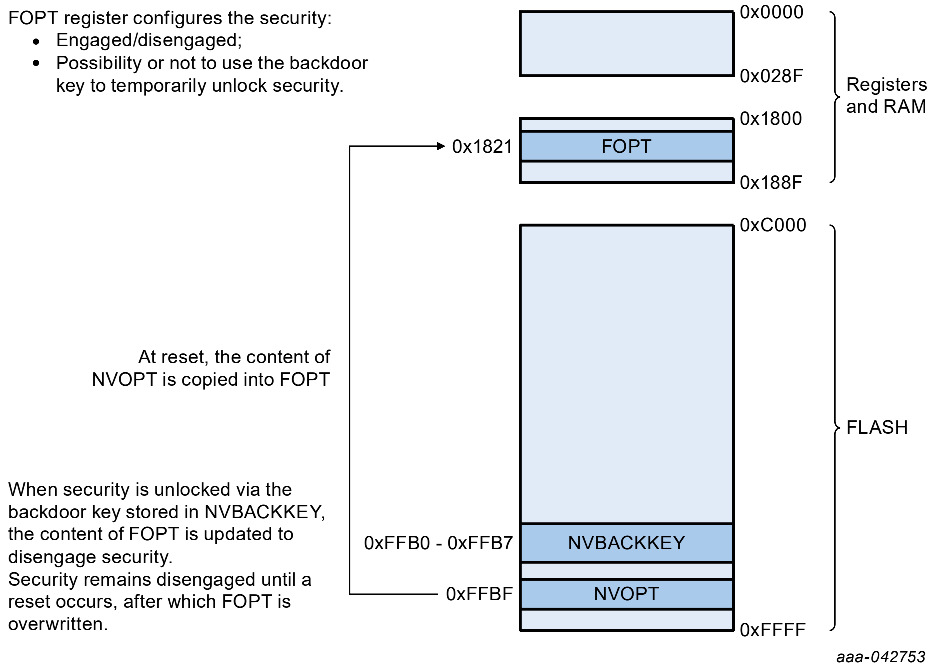

FOPT register configures the security:
• Engaged/disengaged;
• Possibility or not to use the backdoor key to temporarily unlock security.
0x0000
0x028F
Registers and RAM
0x1800
0x1821
FOPT
0x188F
0xC000
At reset, the content of NVOPT is copied into FOPT
FLASH
When security is unlocked via the backdoor key stored in NVBACKKEY, the content of FOPT is updated to disengage security. Security remains disengaged until a reset occurs, after which FOPT is overwritten.
0xFFB0 - 0xFFB7
NVBACKKEY
0xFFBF
NVOPT
0xFFFF
aaa-042753

Figure 5. Security registers

Enabling the security feature disables the ability for NXP to perform failure analysis without first completely erasing all flash memory contents. If the security feature is implemented, the customer is responsible for providing NXP with unsecured parts before any failure analysis can begin. As an alternative, customers may also supply the entire contents of the device flash memory data as part of the return process to allow NXP to erase and subsequently restore the device to its original condition.

4.2 Register configuration

The FOPT/NVOPT register fields are shown in Table 2.

Table 2. Fields of the FOPT/NVOPT registers

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
KEYEN	FNORED	—	—	—	—	SEC1	SEC0

When KEYEN bit is 1, the backdoor key can be used to temporarily unlock security. When KEYEN bit is 0, security can only be disabled via a mass erase.

Vector redirection is enabled when FNORED is 0 and disabled when FNORED is 1. Refer to the manuals[2] for more information on vector redirection.

The SEC[1:0] bits determine the state of security: security is disengaged when SEC[1:0] = 0b10. All other values engage security. Note that after FLASH has been erased, the SEC[1:0] bits are set to 1, which secures the device after reset. In order to keep security disengaged, it is recommended to write SEC[1:0] = 0b10 immediately after erasing the FLASH, before a reset occurs.

At production, NXP programs NVOPT register with value 0x82, which maintains the device unsecure. NVBACKKEY registers are programmed with value 0xFF.

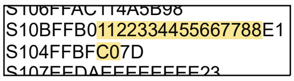
4.3 Engaging security

NVOPT and NVBACKKEY registers can be written in FLASH:

- By the user during programming. The following declarations can be written in the source code:

```
volatile const UINT8 FLASH_NVBACKKEY[8] @0xFFB0 =
 {0x11,0x22,0x33,0x44,0x55,0x66,0x77,0x88};
volatile const UINT8 FLASH_NVOPT @0xFFBF = 0xC0;
```

As a result, the values to be written at addresses 0xFFB0-0xFFB7 and 0xFFBF appear in the s19 file:



```
S106FFAC114A5B98
S10BFFB01122334455667788E1
S104FFBFC07D
S107FFDAFFFFFFFF23
```

Figure 6. s19 extract showing the value to be written at 0xFFB0-0xFFB7 and 0xFFBF

Important: *CodeWarrior debugger prevents NVOPT register from being programmed with a value securing the FLASH, as securing the FLASH prevents further access from the debugger. This implies that even if value 0xC0 is indicated in the s19 to be programmed at 0xFFBF, the CodeWarrior debugger changes this value during programming in order to maintain the FLASH unsecure and the debug session active once the programming is complete. When using CodeWarrior debugger for programming, securing the FLASH can only be done at runtime by the application, as described below. In other words, in order to be able to write NVOPT during programming with a value securing the FLASH, the user must use a tool different from CodeWarrior debugger, for example CodeWarrior Flash Programmer, the interfaces provided with the P&E Micro programming tools, or a custom programming tool or interface.*

- At runtime by the application using the library function TPMS_FLASH_WRITE. The program can execute the following instructions:

```
au8BackDoorKey[0] = 0x11;
au8BackDoorKey[1] = 0x22;
au8BackDoorKey[2] = 0x33;
au8BackDoorKey[3] = 0x44;
au8BackDoorKey[4] = 0x55;
au8BackDoorKey[5] = 0x66;
au8BackDoorKey[6] = 0x77;
au8BackDoorKey[7] = 0x88;
vfnFlashWrite(0xFFB0, (const UINT8*)au8BackDoorKey, 8);

u8NewNVOPT = (FOPT_KEYEN_MASK | FOPT_FNORED_MASK);
TPMS_FLASH_WRITE(0xFFBF, (const UINT8*)&u8NewNVOPT, 1u);
```

Note: *The examples above assume that NVBACKKEY has a fixed value, which may not be the case in a real application. The value could be unique to each device and communicated to the application via LF or serial communication.*

After NVPROT has been written, security is engaged after a reset occurs. STOP1 exit is considered a reset source.

4.4 Unlocking security

If KEYEN bit is 1, security can be unlocked temporarily using the backdoor key. An example of function unlocking security via the backdoor key is shown.

```
UINT8 u8DisengageSecurity (void)
{UINT8 *p;
UINT8 u8Sec_disengaged;
DisableInterrupts;
/* Do unsecure */
FCNFG_KEYACC = 1;
p=(UINT8*)0xFFB0;
*p = 0x11;
p++;
*p = 0x22;
p++;
*p = 0x33;
p++;
*p = 0x44;
p++;
*p = 0x55;
p++;
*p = 0x66;
p++;
*p = 0x77;
p++;
*p = 0x88;
FCNFG_KEYACC = 0;
/* Wait a little bit for FOPT to be updated */
__asm nop;
__asm nop;
__asm nop;
__asm nop;
__asm nop;
__asm nop;
/* Here we can check FOPT value to see that security is
 disengaged */
if (FOPT_SEC == 0x02)
{u8Sec_disengaged = 0;}
else
{u8Sec_disengaged = 1;}
EnableInterrupts;
return u8Sec_disengaged;}
```

After this function has been executed, security is disengaged until a reset occurs. STOP1 exit is considered a reset source.

4.5 Security and protection

When protection is enabled and security is engaged, the user first needs to temporarily unlock the security by writing the backdoor key, then the user can enter BACKGROUND DEBUG mode to disable the protection.

5 References

[1] **AN12523** — *Firmware versus Library model applications*, application note
https://www.nxp.com/docs/en/application-note/AN12523.pdf

[2] **FXTH87ERM** — *FXTH87E, Family of Tire Pressure Monitor Sensors*, reference manual
https://www.nxp.com/docs/en/reference-manual/FXTH87ERM.pdf

[3] **UM11227** — *NTM88 family of tire pressure monitor sensors*, user manual
https://www.nxp.com/docs/en/user-guide/UM11227.pdf

6 Legal information

6.1 Definitions

Draft — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

6.2 Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors. In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory. Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification. Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products. NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at http://www.nxp.com/profile/terms, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Evaluation products — This product is provided on an "as is" and "with all faults" basis for evaluation purposes only. NXP Semiconductors, its affiliates and their suppliers expressly disclaim all warranties, whether express, implied or statutory, including but not limited to the implied warranties of non-infringement, merchantability and fitness for a particular purpose. The entire risk as to the quality, or arising out of the use or performance, of this product remains with customer. In no event shall NXP Semiconductors, its affiliates or their suppliers be liable to customer for any special, indirect, consequential, punitive or incidental damages (including without limitation damages for loss of business, business interruption, loss of use, loss of data or information, and the like) arising out the use of or inability to use the product, whether or not based on tort (including negligence), strict liability, breach of contract, breach of warranty or any other theory, even if advised of the possibility of such damages. Notwithstanding any damages that customer might incur for any reason whatsoever (including without limitation, all damages referenced above and all direct or general damages), the entire liability of NXP Semiconductors, its affiliates and their suppliers and customer's exclusive remedy for all of the foregoing shall be limited to actual damages incurred by customer based on reasonable reliance up to the greater of the amount actually paid by customer for the product or five dollars (US$5.00). The foregoing limitations, exclusions and disclaimers shall apply to the maximum extent permitted by applicable law, even if any remedy fails of its essential purpose.

Translations — A non-English (translated) version of a document is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Security — Customer understands that all NXP products may be subject to unidentified or documented vulnerabilities. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer's applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately. Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP. NXP has a Product Security Incident Response Team (PSIRT) (reachable at PSIRT@nxp.com) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

6.3 Trademarks

Notice: All referenced brands, product names, service names and trademarks are the property of their respective owners.

CodeWarrior — is a trademark of NXP B.V.

NXP — wordmark and logo are trademarks of NXP B.V.

Tables

Figures

Contents

Please be aware that important notices concerning this document and the product(s) described herein, have been included in section 'Legal information'.

For more information, please visit: http://www.nxp.com
For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: 22 July 2021
Document identifier: AN13321