



# NTA5332

## NTAG 5 boost - NFC Forum-compliant I<sup>2</sup>C bridge for tiny devices

Rev. 3.3 — 3 July 2020  
544733

Product data sheet  
COMPANY PUBLIC

### 1 General description

NTAG 5 boost uses active load modulation (ALM) to deliver robust and reliable communication with NFC phones, bringing a new level of convenience to tiny devices.

NXP's NTAG 5 boost shrinks the NFC footprint while adding AES security, so designers can deliver ultra-compact devices for use in IoT, consumer, and industrial applications. It offers an NFC Forum-compliant (customer development board is NFC Forum certified - Certification ID: 58625) contactless interface that delivers exceptional read range, giving tiny devices the ability to interact with the cloud and other NFC-enabled devices, including smartphones.

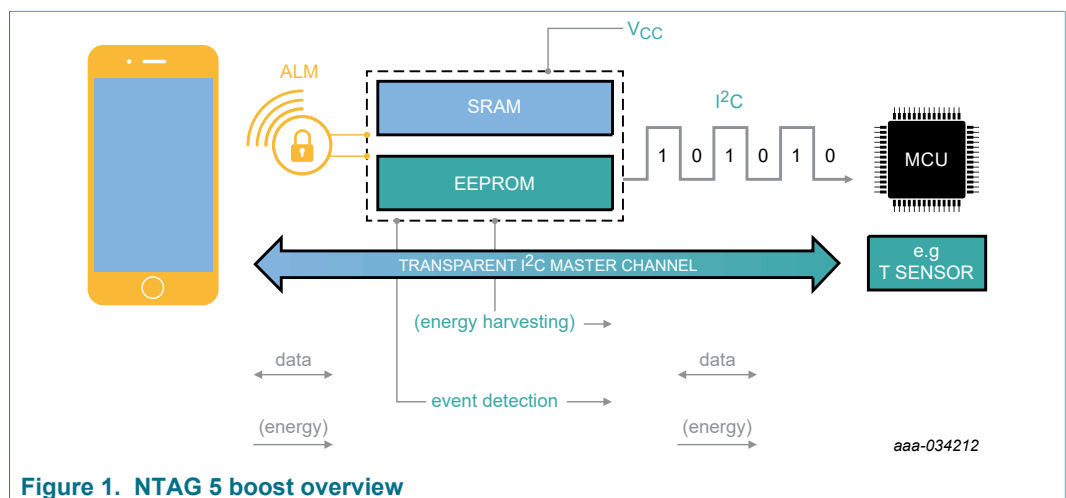


Figure 1. NTAG 5 boost overview

ALM allows construction of a compact yet highly reliable antenna, creating a significantly smaller footprint without compromising the read range. When operating in ALM mode, the read range is significantly longer than when operating in passive mode.

An energy-efficient design, equipped with a hard power-down mode and a standby current of typically less than 10  $\mu$ A, ensures long battery life.

2048 bytes (16384 bits) of user memory can be divided into three areas, and each area can use a different protection level, varying from no protection to 32-/64-bit password protection or up to 128-bit AES-protected read/write access with mutual authentication. Different parties in the value chain can have their own dedicated memory areas for storing access data.

The NTAG 5 boost comes with pre-programmed proof-of-origin functionality to verify authenticity. The elliptic curve cryptography (ECC) based originality signature can be locked or reprogrammed by the customer.

With NTAG 5 boost, the device can connect to the cloud with a single tap. The connection uses an NFC Forum-compliant data exchange mechanism involving 256 bytes (2048 bits) SRAM to ensure highly interoperable data transfers.



## 2 Features and benefits

- Antenna size reduction by a factor of 40, same read range as in passive load modulation
- Long battery life due to low standby current and hard power-down
- Adjustable security levels up to mutual AES authentication
- Flexible split between three open and/or protected memory areas
- Ensured authenticity of product through value chain
- Interoperable data exchange according to NFC Forum standards
- Interoperable and high performance NFC interface
  - [ISO/IEC 15693](#) and NFC Forum [Type 5 Tag](#) compliant
  - 64-bit Unique IDentifier
- Reliable and robust memory
  - 2048 bytes (16384 bits) user EEPROM on top of configuration memory
  - 256 bytes (2048 bits) SRAM for frequently changing data and pass-through mode
  - 40 years data retention
  - Write endurance of 1 000 000 cycles
- Configurable contact interface
  - [I<sup>2</sup>C slave](#) standard (100 kHz) and fast (400 kHz) mode
  - Transparent I<sup>2</sup>C master channel (for example, read sensors without an MCU)
  - One configurable event detection pin
  - Two GPIOs as multiplexed I<sup>2</sup>C lines
  - Two Pulse Width Modulation (PWM) channels as multiplexed GPIOs and/or ED pin
  - 1.62 V to 5.5 V supply voltage
- Scalable security for access and data protection
  - Disable NFC interface temporarily
  - Disable I<sup>2</sup>C interface temporarily
  - NFC PRIVACY mode
  - Read-only protection as defined in NFC Forum Type 5 Tag Specification
  - Full, read-only, or no memory access based on 32-bit password from both interfaces
  - Optional 64-bit password protection from NFC perspective
  - 128-bit AES authentication as defined in [ISO/IEC 15693](#)
  - ECC-based reprogrammable originality signature
- Multiple fast data transfer mode
  - Pass-through mode with 256 byte SRAM buffer
  - Standardized data transfer mode (PHDC, TNEP)
- Low-power budget application support
  - Energy harvesting with configurable output voltage up to 30 mW
  - Low-power standby current typically <10 μA
  - Hard power down current typically <0.25 μA
- Very robust architecture
  - -40 °C to 105 °C for EEPROM read, SRAM and register access
  - -40 °C to 85 °C for EEPROM write access
- Extensive product support package
  - Feature specific application notes
  - Development board including software and source code
  - Hands-on training

### 3 Applications

---

- Use cases
  - Simple dynamic secure pairing
  - Commissioning
  - Parameterization
  - Diagnosis
  - Firmware download
  - Low BoM and low-power data acquisition for sensors
  - Calibration
  - Trimming
  - Authenticity check and data protection
  - Late "in the box" configuration
  - LED driver configuration
  - NFC charging
- Applications
  - Lighting
  - Smart home
  - Hearable and Wearable
  - Consumer
  - Industrial
  - Gaming
  - Smart sensor
  - Smart metering

## 4 Ordering information

Table 1. Ordering information

Orderable part number	Package		Version
	Name	Description	
NTA53321G0FHKZ	XQFN16	NTAG 5 boost with I <sup>2</sup> C master/slave interface, AES authentication, ALM and 2048 bytes user EEPROM plastic, extremely thin quad flat package; no leads; 16 terminals	SOT1161-2
NTA53321G0FTTZ	TSSOP16	NTAG 5 boost with I <sup>2</sup> C master/slave interface, AES authentication, ALM and 2048 bytes user EEPROM plastic, thin shrink small outline package; 16 leads; 0.65 mm pitch; 5 mm x 4.4 mm x 1.1 mm body	SOT403-1
NTA53321G0FUAV	Wafer	NTAG 5 boost; 8 inch wafer, 150 μm thickness, on film frame carrier, electronic fail die marking according to SECS-II format)	-

**REMARK:** Wafer specification addendum is available after exchange of a non-disclosure agreement (NDA)

## 5 Marking

Table 2. Marking codes

Type number	Marking code			
	Line A	Line B	Line C	Line D
NTA53321G0FHK	A21	DBSN ASID	DYWW	-
NTA53321G0FTT	NA53321	DBID ASID	ZnDYY	WW

Used abbreviations:

ASID: Assembly Sequence ID

D: RHF-2006 indicator

DBID: Diffusion Batch ID

DBSN: Diffusion Batch Sequence Number

n: Assembly Centre Code

WW: week

Y or YY: year

Z: Diffusion Centre Code

## 6 Block diagram

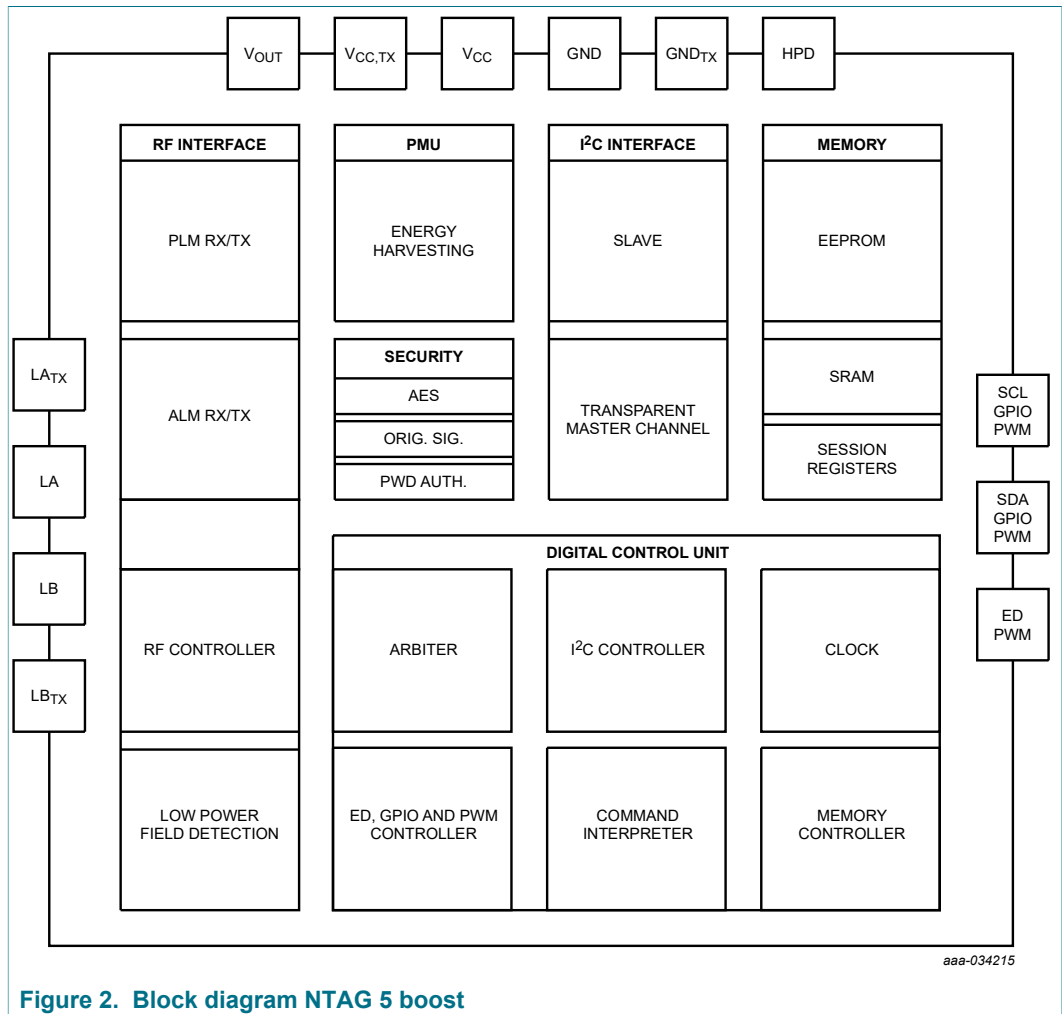


Figure 2. Block diagram NTAG 5 boost

## 7 Pinning information

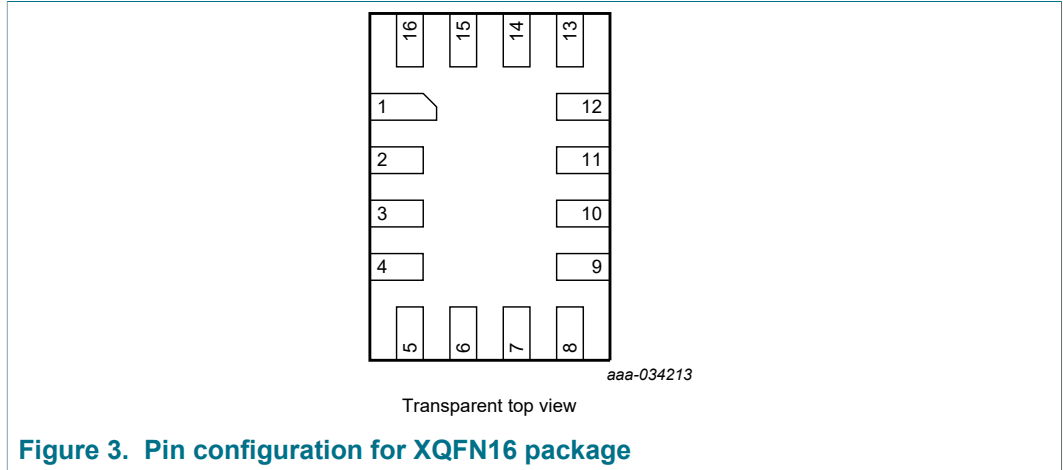


Figure 3. Pin configuration for XQFN16 package

Table 3. Pin description for XQFN16

Pin	Symbol	Description	When unused
1	GND	Ground	connect to GND
2	GND <sub>TX</sub>	Ground for ALM	connect to GND
3	V <sub>CC, TX</sub>	External power supply for ALM	keep floating
4	N.C.	not connected	keep floating
5	N.C.	not connected	keep floating
6	SDA/GPIO1/PWM1	Multiplexed serial data I <sup>2</sup> C, GPIO1 and PWM1	keep floating
7	SCL/GPIO0/PWM0	Multiplexed serial clock I <sup>2</sup> C, GPIO0 and PWM0	keep floating
8	ED/PWM0	Multiplexed event detection and PWM0	keep floating
9	V <sub>CC</sub>	External power supply	keep floating
10	HPD	Hard power down	keep floating
11	GND	Ground	connect to GND
12	V <sub>OUT</sub>	Energy harvesting voltage output	keep floating
13	LB <sub>TX</sub>	Antenna connection TX	keep floating
14	LB	Antenna connection	keep floating
15	LA	Antenna connection	keep floating
16	LA <sub>TX</sub>	Antenna connection TX	keep floating

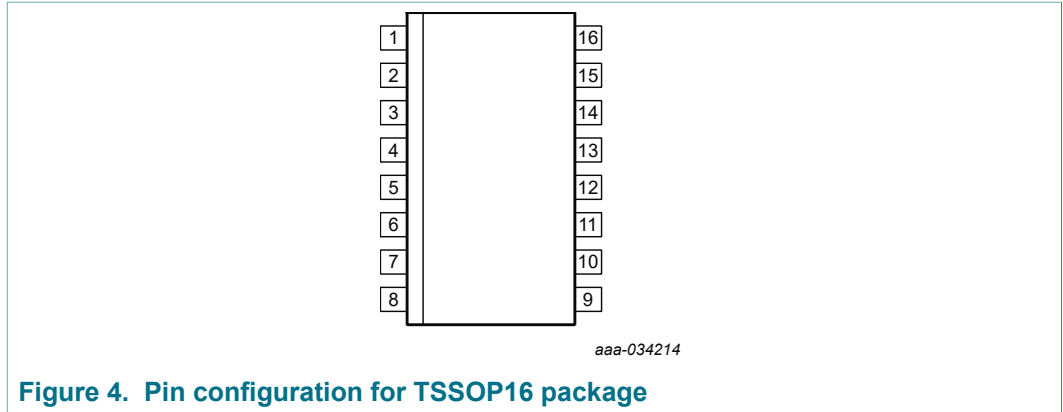


Figure 4. Pin configuration for TSSOP16 package

Table 4. Pin description for TSSOP16

Pin	Symbol	Description	When unused
1	LA	Antenna connection	keep floating
2	LA <sub>TX</sub>	Antenna connection TX	keep floating
3	GND	Ground	connect to GND
4	GND <sub>TX</sub>	Ground for ALM Ground	connect to GND
5	V <sub>CC, TX</sub>	External power supply for ALM	keep floating
6	N.C.	not connected	keep floating
7	N.C.	not connected	keep floating
8	SDA/GPIO1/PWM1	Multiplexed serial data I <sup>2</sup> C, GPIO1 and PWM1	keep floating
9	SCL/GPIO0/PWM0	Multiplexed serial clock I <sup>2</sup> C, GPIO0 and PWM0	keep floating
10	ED/PWM0	Multiplexed event detection and PWM0	keep floating
11	V <sub>CC</sub>	External power supply	keep floating
12	HPD	Hard power down	keep floating
13	GND	Ground	connect to GND
14	V <sub>OUT</sub>	Energy harvesting voltage output	keep floating
15	LB <sub>TX</sub>	Antenna connection TX	keep floating
16	LB	Antenna connection	keep floating



## 8 Functional description

### 8.1 Memory Organization

#### 8.1.1 General

The entire memory is divided into three different parts:

- User memory
  - This part of the memory is intended to be used to store user data. It is organized in blocks of 4 bytes each (see [Section 8.1.2](#)).
  - According to NFC Forum Type 5 Tag Specification, EEPROM block 0 contains the Capability Container directly followed by the NDEF Message TLV. If NTAG 5 boost is used in a proprietary way, any user data may be stored in the user memory. Direct read/write access with the standard READ BLOCK and WRITE BLOCK commands (see [Section 8.2.4.5](#)) to this part of the memory is possible depending on the related security and write protection conditions.
  - 16-bit counter
    - The last block of the EEPROM memory from NFC perspective contains the 16-bit counter and the counter protection flag (see [Section 8.1.2.1](#)). This counter is not accessible from I<sup>2</sup>C perspective.
- Configuration area
  - Within this part of the memory all configuration options are stored (see [Section 8.1.3](#)). This memory area can only be accessed with the READ CONFIG (see [Section 8.2.4.2.1](#)) or WRITE CONFIG (see [Section 8.2.4.2.2](#)) commands from NFC perspective. From I<sup>2</sup>C perspective, normal READ and WRITE commands are used.
  - The configuration area contains required security-related information, such as access keys with related privileges, headers, customer ID (CID), originality signature and many more which will be loaded at power-on reset.
  - Access to configuration blocks may be blocked at all or password protected with related configuration bits.
  - All session registers are accessible in the configuration area as long as not locked by LOCK\_SESSION\_REG. These configuration items can be changed on the fly and have immediate effect, but get lost after power-on reset.
- SRAM
  - SRAM is accessible when NTAG 5 boost is V<sub>CC</sub> supplied and SRAM\_ENABLE is set to 1b.
  - Volatile SRAM can be used for fast and frequent data transfer (see [Section 8.1.5](#)). With WRITE SRAM (see [Section 8.2.4.6.2](#)) and READ SRAM (see [Section 8.2.4.6.1](#)), the content is written or read.
  - When the SRAM gets mapped to user memory (the start address is always block 0 from both interfaces), then standard READ BLOCK and WRITE BLOCK commands can be used. This mechanism is used, e.g., for PHDC or dynamic pairing.
  - From I<sup>2</sup>C perspective, SRAM is always located from address 2000h to 203Fh.

WARNING: The content of bytes and bits defined as RFU SHALL NOT be changed.

#### 8.1.2 User memory

According to NFC Forum Type 5 Tag Specification, the user accessible EEPROM memory is divided into blocks. A block is the smallest access unit. For NTAG 5 boost,

each block consists of 4 bytes (1 block = 32 bits). Bit 0 in each byte represents the least significant bit (lsb) and bit 7 the most significant bit (msb), respectively.

User EEPROM map looks totally the same from NFC and I<sup>2</sup>C perspective.

The last block contains the 16-bit counter (see [Section 8.1.2.1](#)). It is only accessible from NFC perspective.

NTAG 5 boost offers 2048 bytes (16384 bits) of user memory.

**Table 5. User memory organization**

Block Address		Byte 0 (LSB)	Byte 1	Byte 2	Byte 3 (MSB)	Description
NFC	I <sup>2</sup> C					
00h	0000h	Capability Container				or user memory
01h	0001h	User Memory				
:	:					
1FEh	01FEh					
1FFh	-	C0	C1	00h	PROT	Counter

User data at delivery contains an NFC Forum-compliant capability container and an NDEF message containing the URL [www.nxp.com/nfc](http://www.nxp.com/nfc). First 6 blocks are initialized as illustrated in below table. The counter block is initialized with all 00h. Content of the rest of the user memory is undefined and contains random (rnd) data at delivery.

**Table 6. Memory content at delivery**

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
00h	0000h	E1h	40h	80h	09h
01h	0001h	03h	10h	D1h	01h
02h	0002h	0Ch	55h	01h	6Eh
03h	0003h	78h	70h	2Eh	63h
04h	0004h	6Fh	6Dh	2Fh	6Eh
05h	0005h	66h	63h	FEh	00h
06h	0006h	rnd	rnd	rnd	rnd
...	...	rnd	rnd	rnd	rnd
1FFh	-	00h	00h	00h	00h

**8.1.2.1 16-bit counter**

Last Block of the user memory contains the 16-bit counter. The block can be accessed with the standard read and write commands but special data format is required.

The standard protection conditions for the user memory are not valid for the counter block.

Counter block can only be accessed from NFC perspective.

The 16-bit counter can be

- preset to initial start value protected with the write password or by mutual authentication with a key with the Write privilege

- read
- increased by one, optionally protected with the read password or by mutual authentication with a key with the Read privilege

The counter can be read with an (EXTENDED) READ SINGLE BLOCK to the last block or (EXTENDED) READ MULTIPLE BLOCK command including the last block. The 4 byte data of the counter block provide the following information in [Table 7](#).

**Table 7. COUNTER BLOCK data structure**

Byte	Name	Value	Description
0	C0 (LSB)	00h - FFh	Counter value
1	C1 (MSB)	00h - FFh	
2	-	00h	RFU
3	PROT	00h	Incrementing of the counter value is not protected
		01h	Incrementing of the counter value is protected with the read password or by mutual authentication depending on the used security level

The counter can be preset to a start value with an (EXTENDED) WRITE SINGLE BLOCK command to counter block. The counter can only be preset to a start value after a SET PASSWORD command with the write password or a valid mutual authentication with a key with the Write privilege, depending on the used security level.

The PROT byte (data byte 3) value defines if the protection to increment the counter is enabled or disabled. If the protection is enabled, the read password or a valid mutual authentication with a key with the Read privilege is required to increment the counter value, again depending on the used security level.

The data for the (EXTENDED) WRITE SINGLE BLOCK command to preset the counter is defined in [Table 8](#).

**Remark:** A Preset counter value of 0x0001 is not possible, a (EXTENDED) WRITE SINGLE BLOCK command with that value will only increment the counter.

**Table 8. Preset counter data structure**

Byte	Name	Value	Description
0	C0	00h, 02h - FFh (LSB)	Counter value
1	C1	00h - FFh (MSB)	
2	-	00h	RFU
3	PROT	00h	Disable the protection to increment the counter
		01h	Enable the protection to increment the counter with read password or mutual authentication

To increment the counter by one with a (EXTENDED) WRITE SINGLE BLOCK command to counter block. If the protection to increment the counter is enabled, a SET PASSWORD command with the read password or a valid mutual authentication with a key with the Read privilege is required before.

The data for the (EXTENDED) WRITE SINGLE BLOCK command to increment the counter is defined in [Table 9](#).

**Remark:** The counter can only be incremented with the C0 and C1 values defined in [Table 9](#). Other values than that preset the counter if a SET PASSWORD command with the write password or a valid mutual authentication with a key with the Write privilege has been executed before or leads to an error message.

**Table 9. Increment counter data structure**

Byte	Name	Value	Description
0	C0	01h (LSB)	Value to increment the counter
1	C1	00h (MSB)	
2	-	00h	RFU
3	-	00h	RFU

### 8.1.3 Configuration memory

The configuration memory contains the security and configuration information. Access to this memory area is only possible with WRITE CONFIG (see [Section 8.2.4.2.2](#)) and READ CONFIG (see [Section 8.2.4.2.1](#)) commands depending on the initialization status.

Writing to blocks with only RFU bytes is not possible and results in error code 0Fh from NFC perspective and NAK from I<sup>2</sup>C perspective. Reading complete RFU blocks results in receiving all bytes 00h.

Changing RFU bytes and bits is not allowed and may result in unintended behavior.

From I<sup>2</sup>C perspective, the configuration can be accessed using READ MEMORY and WRITE MEMORY command. Block address of configuration area from I<sup>2</sup>C perspective starts from 1000h.

In [Table 10](#) all NFC\_KHs, NFC\_KPs and AES\_KEYs (all marked with an asterisk) are only available in AES mode. NFC\_KHs and NFC\_KPs are not available at all and set to RFU in plain password mode. In the area of KEY\_0 and KEY\_1, the plain passwords are stored (see [Section 8.1.3.9](#)). The rest of the KEY area is RFU in plain password mode.

Different features can be configured with CONFIG bits. Similar to all other configuration options, the effect does not take place in the current session. The effect takes place after POR. If immediate change is expected, related session register bytes or bits need to be used (see [Section 8.1.4](#)).

To which section each block belongs is defined in first column (Sec.). Sections might be locked by setting related bit to 1b (see [Section 8.1.3.33](#)).

**Table 10. Configuration Memory organization**

Sec.	Block Address		Byte 0	Byte 1	Byte 2	Byte 3	Description
	NFC	I <sup>2</sup> C					
0	00h	1000h	ORIGINALITY_SIGNATURE				32 byte Originality Signature (see <a href="#">Section 8.1.3.1</a> )
0	...	...					
0	07h	1007h					
0	08h	1008h	CH	RFU		Configuration Header (see <a href="#">Section 8.1.3.2</a> )	
0	09h	1009h	CID		RFU		Customer ID (see <a href="#">Section 8.1.3.3</a> )

Sec.	Block Address		Byte 0	Byte 1	Byte 2	Byte 3	Description
	NFC	I <sup>2</sup> C					
N/A	0Ah	100Ah	RFU				
N/A	0Bh	100Bh	RFU				
0	0Ch	100Ch	RFU	NFC_GCH	RFU		NFC Global Crypto Header (see <a href="#">Section 8.1.3.4</a> )
0	0Dh	100Dh	RFU	NFC_CCH	RFU		NFC Crypto Configuration Header (see <a href="#">Section 8.1.3.5</a> )
0	0Eh	100Eh	NFC_AUTH_LIMIT		RFU		NFC Authentication Limit Counter (see <a href="#">Section 8.1.3.6</a> )
N/A	0Fh	100Fh	RFU				
0	10h	1010h	RFU	NFC_KH0*	RFU		NFC Key Header 0 (see <a href="#">Section 8.1.3.7</a> )
0	11h	1011h	NFC_KP0*	RFU			NFC Key Privileges 0 (see <a href="#">Section 8.1.3.8</a> )
0	12h	1012h	RFU	NFC_KH1*	RFU		NFC Key Header 1 (see <a href="#">Section 8.1.3.7</a> )
0	13h	1013h	NFC_KP1*	RFU			NFC Key Privileges 1 (see <a href="#">Section 8.1.3.8</a> )
0	14h	1014h	RFU	NFC_KH2*	RFU		NFC Key Header 2 (see <a href="#">Section 8.1.3.7</a> )
0	15h	1015h	NFC_KP2*	RFU			NFC Key Privileges 2 (see <a href="#">Section 8.1.3.8</a> )
0	16h	1016h	RFU	NFC_KH3*	RFU		NFC Key Header 3 (see <a href="#">Section 8.1.3.7</a> )
0	17h	1017h	NFC_KP3*	RFU			NFC Key Privileges 3 (see <a href="#">Section 8.1.3.8</a> )
N/A	18h	1018h	RFU				
N/A	...	...					
N/A	1Fh	101Fh					
0	20h	1020h	KEY_0*				AES key 0 or NFC_PWD_0 to NFC_PWD_3 (see <a href="#">Section 8.1.3.9</a> )
0	21h	1021h					
0	22h	1022h					
0	23h	1023h					

Sec.	Block Address		Byte 0	Byte 1	Byte 2	Byte 3	Description
	NFC	I <sup>2</sup> C					
0	24h	1024h	KEY_1*				AES key 1 or NFC_PWD_4 to NFC_PWD_6 (see <a href="#">Section 8.1.3.9</a> )
0	25h	1025h					
0	26h	1026h					
0	27h	1027h					
0	28h	1028h	KEY_2*				AES key 2 or RFU (see <a href="#">Section 8.1.3.9</a> )
0	29h	1029h					
0	2Ah	102Ah					
0	2Bh	102Bh					
0	2Ch	102Ch	KEY_3*				AES key 3 or RFU (see <a href="#">Section 8.1.3.9</a> )
0	2Dh	102Dh					
0	2Eh	102Eh					
0	2Fh	102Fh					
1	30h	1030h	I2C_KH	RFU			I <sup>2</sup> C Key Header (see <a href="#">Section 8.1.3.10</a> )
1	31h	1031h	I2C_PP	I2C_PPC	RFU		I <sup>2</sup> C Protection Pointer and Config (see <a href="#">Section 8.1.3.11</a> )
1	32h	1032h	I2C_AUTH_LIMIT		RFU		Authentication Limit Counter (see <a href="#">Section 8.1.3.12</a> )
1	33h	1033h	I2C_PWD_0				I <sup>2</sup> C read password (see <a href="#">Section 8.1.3.9</a> )
1	34h	1034h	I2C_PWD_1				I <sup>2</sup> C write password (see <a href="#">Section 8.1.3.9</a> )
1	35h	1035h	I2C_PWD_2				Restricted AREA_1 I <sup>2</sup> C read password (see <a href="#">Section 8.1.3.9</a> )
1	36h	1036h	I2C_PWD_3				Restricted AREA_1 I <sup>2</sup> C write password (see <a href="#">Section 8.1.3.9</a> )
2	37h	1037h	CONFIG				Feature Configuration (see <a href="#">Section 8.1.3.13</a> )
3	38h	1038h	SYNC_DATA_BLOCK		RFU		Block may be used for data transfer synchronization (see <a href="#">Section 8.1.3.14</a> )

Sec.	Block Address		Byte 0	Byte 1	Byte 2	Byte 3	Description
	NFC	I <sup>2</sup> C					
3	39h	1039h	PWM_GPIO_CONFIG		RFU		PWM and GPIO Configuration (see <a href="#">Section 8.1.3.15</a> )
3	3Ah	103Ah	PWM0_ON_OFF				PWM1 Configuration (see <a href="#">Section 8.1.3.16</a> )
3	3Bh	103Bh	PWM1_ON_OFF				PWM1 Configuration (see <a href="#">Section 8.1.3.16</a> )
3	3Ch	103Ch	WDT_CONFIG			SRAM_COPY_BYTES	Watch Dog Timer Configuration (see <a href="#">Section 8.1.3.17</a> ) and SRAM Copy Bytes (see <a href="#">Section 8.1.5</a> )
3	3Dh	103Dh	EH_CONF	RFU	ED_CONF	RFU	Energy Harvesting (see <a href="#">Section 8.1.3.18</a> ) and Event Detection Pin (see <a href="#">Section 8.1.3.19</a> ) Configuration
3	3Eh	103Eh	I2C_SLAVE_CONFIG		I2C_MASTER_CONFIG		I <sup>2</sup> C Configuration (see <a href="#">Section 8.1.3.20</a> and <a href="#">Section 8.1.3.21</a> )
3	3Fh	103Fh	SEC_CONF	SRAM_CONF_PROT	PP_AREA_1		Device Security Configuration (see <a href="#">Section 8.1.3.22</a> ) SRAM and Configuration Protection (see <a href="#">Section 8.1.3.23</a> ) AREA_1 Protection Pointer (see <a href="#">Section 8.1.3.24</a> )
3	40h	1040h	ALM_CONF				ALM configuration (see <a href="#">Section 8.1.3.25</a> )
3	41h	1041h	ALM_LOOKUP_TABLE				ALM lookup table (see <a href="#">Section 8.1.3.25</a> )
3	...	...					
3	44h	1044h					
7	45h	1045	SRAM_DEFAULT				Default SRAM content (see <a href="#">Section 8.1.5</a> )
7	...	...					
7	54h	1054					
4	55h	1055h	AFI	RFU			Application Family Identifier (see <a href="#">Section 8.2.4.9.1</a> )

Sec.	Block Address		Byte 0	Byte 1	Byte 2	Byte 3	Description
	NFC	I <sup>2</sup> C					
4	56h	1056h	DSFID	RFU			DSFID (see <a href="#">Section 8.1.3.27</a> )
4	57h	1057h	EAS_ID		RFU		EAS ID (see <a href="#">Section 8.1.3.28</a> )
4	58h	1058h	NFC_PP_AREA_0H	NFC_PPC	RFU		NFC Protection Pointer (see <a href="#">Section 8.1.3.29</a> ) and NFC Protection Pointer Conditions (see <a href="#">Section 8.1.3.30</a> )
N/A	59h	1059h	RFU				
N/A	...	...					
N/A	69h	1069h					
5	6Ah	106Ah	NFC_LOCK_BLOCK		RFU		NFC Lock block configuration (see <a href="#">Section 8.6.1</a> )
5	...	...					
5	89h	1089h					
6	8Ah	108Ah	I2C_LOCK_BLOCK		RFU		I <sup>2</sup> C Lock block configuration (see <a href="#">Section 8.1.3.32</a> )
6	...	...					
6	91h	1091h					
8	92h	1092h	NFC_SECTION_LOCK	RFU			NFC section lock bytes (see <a href="#">Table 87</a> )
8	93h	1093h					
8	94h	1094h	I2C_SECTION_LOCK	RFU			I <sup>2</sup> C section lock bytes (see <a href="#">Table 89</a> )
8	95h	1095h					
N/A	96h	1096h	RFU from NFC perspective I2C_PWD_0_AUTH			I <sup>2</sup> C read password authenticate (see <a href="#">Section 8.1.3.9</a> )	
N/A	97h	1097h	RFU from NFC perspective I2C_PWD_1_AUTH			I <sup>2</sup> C write password authenticate (see <a href="#">Section 8.1.3.9</a> )	
N/A	98h	1098h	RFU from NFC perspective I2C_PWD_2_AUTH			Restricted AREA_1 I <sup>2</sup> C read password authenticate (see <a href="#">Section 8.1.3.9</a> )	
N/A	99h	1099h	RFU from NFC perspective I2C_PWD_3_AUTH			Restricted AREA_1 I <sup>2</sup> C write password authenticate (see <a href="#">Section 8.1.3.9</a> )	
N/A	9Ah	109Ah	RFU				



Sec.	Block Address		Byte 0	Byte 1	Byte 2	Byte 3	Description
	NFC	I <sup>2</sup> C					
N/A	...	...					
N/A	9Fh	109Fh					

**8.1.3.1 Originality Signature**

The Originality signature (see [Section 8.8](#)) is stored in first 8 blocks (block 00h to block 07h) of configuration memory and may be verified by the NFC device using the corresponding ECC public key. As the NXP originality signature is on default not locked, it may be re-programmed by the customer. If the originality check is not needed, it may even be used as additional 32 byte user EEPROM.

**Table 11. 32 Byte Originality Signature**

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
00h	1000h	SIG0 (LSB)	SIG1	SIG2	SIG3
01h	1001h	SIG4	SIG5	SIG6	SIG7
02h	1002h	SIG8	SIG9	SIG10	SIG11
03h	1003h	SIG12	SIG13	SIG14	SIG15
04h	1004h	SIG16	SIG17	SIG18	SIG19
05h	1005h	SIG20	SIG21	SIG22	SIG23
06h	1006h	SIG24	SIG25	SIG26	SIG27
07h	1007h	SIG28	SIG29	SIG30	SIG31 (MSB)

**8.1.3.2 Configuration Header**

The Configuration Header (CH) byte defines the access conditions of both, Customer ID and Originality Signature.

**Table 12. Configuration Header (CH) location**

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
08h	1008h	CH			RFU

Configuration Header byte can be read with READ CONFIG command (see [Section 8.2.4.2.1](#)) and written with WRITE CONFIG command (see [Section 8.2.4.2.2](#)). Once locked (set to E7h), CH byte cannot be updated anymore and Originality Signature and Customer ID gets locked permanently from NFC perspective.

From I<sup>2</sup>C perspective this block can be read and written if not locked by the I<sup>2</sup>C section lock. Once locked, CH byte cannot be updated anymore and Originality Signature and CID gets locked permanently from I<sup>2</sup>C perspective.

**Table 13. Configuration Header Codes**

Value	Mode	Write Access
81h	Writeable (default)	Yes

Value	Mode	Write Access
E7h	Locked	No
All others	Invalid	No

**8.1.3.3 Customer ID (CID)**

The Customer ID at delivery is C000h and can be reprogrammed and locked. It might be used to identify the product.

The two most significant bits (b7 and b6 of CID (MSB)) are always equal to 11b. Only CID[13-0] may be written by customer. Note, that other values of the two most significant bits are RFU.

When the CID is written with WRITE CONFIG command, the 2 most significant bits are always set to 11b. The input CID in WRITE CONFIG command (see [Section 8.2.4.2.2](#)) is bit wise ORed with C000h.

As long as not locked by the I<sup>2</sup>C section lock, CID maybe updated from I<sup>2</sup>C perspective with the same logic as from NFC perspective.

Example: When setting CID to 10AAh, resulting customer-specific CID is D0AAh.

**Table 14. Customer ID (CID) location**

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
09h	1009h	CID (LSB)	CID (MSB)	RFU	

The CID can be permanently locked by setting the Configuration Header to Locked state (see [Table 13](#)) using WRITE CONFIG command. Note, that Originality Signature gets locked, too.

**8.1.3.4 NFC Global Crypto Header**

The NFC Global Crypto Header (NFC\_GCH) defines the status and access of the

- NFC passwords in plain password mode and all other NFC features listed below
- NFC Keys
- NFC Protection Pointer
- NFC Protection Pointer Conditions
- NFC Key Headers
- NFC Key Privileges
- NFC Crypto Configuration Header
- EAS and AFI protection

As long as not locked by the RF section lock, the NFC Global Crypto Header can be written with WRITE CONFIG command (see [Section 8.2.4.2.2](#)). The programming of NFC Global Crypto Header can be done in only one direction from lower state to higher independently from the interface and it is irreversible.

Same rules apply from I<sup>2</sup>C perspective, as long as not locked by the I<sup>2</sup>C section lock.

Once locked (as per table below), GCH cannot be updated anymore.

Table 15. NFC Global Crypto Header (GCH) location

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
0Ch	100Ch	RFU	NFC_GCH	RFU	

Table 16. Global Crypto Header Configuration in plain password mode

Value	Status	Description
81h	Writable (default)	The NFC passwords can be read and written with the READ CONFIG and WRITE CONFIG commands.
E7h	Locked	The NFC passwords cannot be read and written with the READ CONFIG and WRITE CONFIG commands.
all others	Invalid	

NOTE: LOCK PASSWORD command is needed to lock NFC passwords permanently (see [Section 8.2.4.3.4](#)).

Table 17. Global Crypto Header Configuration Value in AES mode

Value	Status	Description
81h	Deactivated (default)	<p>The settings of the NFC Protection Pointer and the NFC Protection Pointer Condition are not activated. Read and write access to the user memory is possible independent of the settings without a previous mutual authentication.</p> <p>The Protection Pointer Address and the NFC Protection Pointer Condition byte can be modified with a PROTECT PAGE command without a previous mutual authentication except of the LOCK PAGE PROTECTION CONDITION command has been successfully executed before.</p> <p>The Keys and Key Privileges can be read and written with the READ CONFIG and WRITE CONFIG commands according to the status of the related Key Header.</p> <p>The settings for the EAS/AFI protection are not activated. Access with the related commands to EAS and AFI is possible without a previous mutual authentication.</p>
87h	Deactivated & privileges locked	<p>The status is the same as for 81h with the exception that the Key Privileges are locked and cannot longer be modified.</p> <p>The settings for the EAS/AFI protection are not activated. Access with the related commands to EAS and AFI is possible without a previous mutual authentication.</p>
C1h	Access right activated	<p>The settings of the Protection Pointer address and the Protection Pointer Condition are enabled. Read and write access protection is enabled according to the initialized values.</p> <p>The Keys and Key Privileges can be read and written with the READ CONFIG and WRITE CONFIG commands according to the status of the related Key Header.</p> <p>The settings for the EAS/AFI protection are enabled. Access with the related commands to EAS and AFI is only possible according to the EAS/AFI protection conditions.</p>

Value	Status	Description
C7h	Access right activated & privileges locked	The status is the same as for C1h with the exception that the Key Privileges are locked and cannot longer be modified. The settings for the EAS/AFI protection are enabled. Access with the related commands to EAS and AFI is only possible according to the EAS/AFI protection conditions.
E7h	Activated	The settings of the NFC Protection Pointer and the NFC Protection Pointer Condition are enabled. Read and write access protection is enabled according to the initialized values. All Key Header Privileges and Keys are locked cannot be modified The settings for the EAS/AFI protection are enabled. Access with the related commands to EAS and AFI is only possible according to the EAS/AFI protection conditions.
all others	Invalid	

**8.1.3.5 NFC Crypto Configuration Header**

The value of the NFC Crypto Configuration Header (NFC\_CCH) locks the NFC Authentication Limit to the defined value and can only be changed after authentication. NFC\_CCH can be written by using the WRITE CONFIGURATION command (see [Section 8.2.4.2.2](#)).

From I<sup>2</sup>C perspective, NFC\_CCH can be accessed with READ and WRITE command, as long as not locked by the I<sup>2</sup>C section lock.

**Table 18. Crypto Configuration Header (CCH) location**

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
0Dh	100Dh	RFU	NFC_CCH	RFU	

**Table 19. Crypto Configuration Header Values**

Value	Mode	Write Access
81h	Unlocked (default)	Authentication limit can be modified.
E7h	Locked	Authentication limit is locked and can only be modified after mutual authentication with a key with activated crypto configuration privilege or authenticating with the write password. In 64-bit password mode both, the read and write password are required, depending on the used security level.
All others	Invalid	

**8.1.3.6 NFC Authentication Limit Counter**

The NFC Authentication Limit Counter is a feature to limit the number of authentications. When enabled, the counter is incremented for every CHALLENGE (see [Section 8.2.4.4.5](#)) or AUTHENTICATION (see [Section 8.6.4](#)) command. Both, positive and negative attempts get counted. On default, the counter is not enabled (NFC\_AUTH\_LIMIT = 0000h).

In plain password mode, NTAG 5 boost implements the NFC Authentication Limit in counting negative Password Authentication attempts with the SET PASSWORD command, except for the Privacy password. The counter will be reset automatically to zero after a successful authentication.

**Table 20. NFC Authentication Limit Counter (NFC\_AUTH\_LIMIT) location**

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
0Eh	100Eh	NFC_AUTH_LIMIT (LSB)	NFC_AUTH_LIMIT (MSB)	RFU	

Byte 0 of Block 0Eh is LSB and Byte 1 is MSB of the NFC Authentication Limit counter value.

The Authentication limit is enabled with the most significant bit of Byte 1 is set to 1b. The remaining 15 bits of NFC\_AUTH\_LIMIT are defining the preset value.

The start value for the Authentication Limit can be preset with a WRITE CONFIG command (see [Section 8.2.4.2.2](#)) if

- the Crypto Config Header is not set to "Locked" and the NFC Global Crypto Header is not set to "Activated" or
- a valid mutual authentication with a key with the Crypto Config privilege set has been executed before.

In plain password mode, the Counter can be written with a WRITE CONFIG command (see [Section 8.2.4.2.2](#)) if

- the Crypto Config Header is not set to "Locked" and NFC Global Crypto Header is not set to "Locked", or
- a valid SET\_PASSWORD command with the write password has been executed before. In 64-bit password mode, both read and write passwords are required.

Examples:

- 8000h enables and presets the authentication limit to 0, which means the maximum number of authentications (32767) before a preset is required again
- F000h enables and presets the authentication limit to 28672

If the NFC Authentication Limit is enabled, the authentication limit value is increased by one at each CHALLENGE or AUTHENTICATE command (first step only). As soon as the value of the Authentication limit reaches

- FF00h: only mutual authentication will be accepted to reset the authentication limit
- FFFFh: no further authentication is possible any longer. This status is irreversible.

**Remark:** The absolute maximum authentication limit value is FFFEh before a preset is required, otherwise the authentication is irreversibly locked (no longer available).

From I<sup>2</sup>C perspective, this block can be read and written if not locked by the I<sup>2</sup>C section lock.

### 8.1.3.7 NFC Key Header

The NFC Key Header bytes (NFC\_KH0, NFC\_KH1, NFC\_KH2, NFC\_KH3) define the status for the related NFC key.

The programming of NFC Key Header can be done in only one direction from lower state to higher and it is irreversible from NFC perspective.

Same rules apply from I<sup>2</sup>C perspective, as long as not locked by the I<sup>2</sup>C section lock. Once locked (as per table below) cannot be updated anymore from I2C interface.

Table below shows the location of the NFC Key Headers in the configuration memory. When using password authentication, these bytes and blocks are all RFU.

**Table 21. NFC Key Header (KHx) location**

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
10h	1010h	RFU	NFC_KH0*	RFU	
12h	1012h	RFU	NFC_KH1*	RFU	
14h	1014h	RFU	NFC_KH2*	RFU	
16h	1016h	RFU	NFC_KH3*	RFU	

If the Global Crypto Header is in the "Activate and locked" state, all crypto settings (Key Headers, Keys and Privileges) are locked and cannot longer be modified.

**Table 22. NFC Key Header Values**

Value	Status	Description
81h	Not active (default)	The related Key and the Key Privileges can be read and written with the READ CONFIG and WRITE CONFIG commands. The related key is not active and cannot be used. The Key Privileges of the related Key are not valid.
E7h	Active and locked	The related Key and its privileges are active and locked and cannot be modified any longer. The related Key cannot be read or written with the READ CONFIG and WRITE CONFIG commands. The related Key Header and Key Privileges can only be read with the READ CONFIG command.
FFh	Disabled	Related Key is disabled and cannot be used

**Remark:** It is recommended to set the key header for not required keys to disabled (FFh).

**8.1.3.8 NFC Key Privileges**

The NFC Key Privileges bytes define the privileges for the related key. Writing with WRITE CONFIG command to the related block depends on the status of the related NFC Key Header (see [Table 22](#))

From I<sup>2</sup>C perspective, this block can be read and written if not locked by the I<sup>2</sup>C section lock.

Table shows the location of the Key Privileges in the configuration memory.

**Table 23. NFC Key Privileges (KPx) location**

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
11h	1011h	NFC_KP0	RFU		

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
13h	1013h	NFC_KP1		RFU	
15h	1015h	NFC_KP2		RFU	
17h	1017h	NFC_KP3		RFU	

Table 24 shows the definition of the Key Privileges bytes KP<sub>x</sub>. The bits define the privileges for the related key used with mutual authentication. If the related bit is set to 1b, the access for the dependent area/feature is granted after mutual authentication with the related key.

Table 24. Definition of NFC Key Privileges bytes KP<sub>x</sub>

Bit	Privilege	Description
7	Restricted AREA_1 Write	Write access to restricted user memory AREA_1
6	Restricted AREA_1 Read	Read access to restricted user memory AREA_1
5	Crypto Config	Preset of Authentication Limit
4	EAS/AFI	Access for write alike command for EAS and AFI as following: PROTECT EAS/AFI SET EAS RESET EAS LOCK EAS WRITE EAS ID WRITE AFI LOCK AFI
3	Destroy	Access to the DESTROY functionality
2	Privacy	Enable/disable of the PRIVACY mode
1	Write	Write access to protected user memory area
0	Read	Read access to protected user memory area

### 8.1.3.9 Keys and passwords

The keys or passwords are stored in the configuration memory. The usage of the individual keys depends on the related Key Privileges.

The state of the keys (including read and write access with the READ CONFIG and WRITE CONFIG commands) depends on the status of the related Key Header and the status of the Global Crypto Header.

From I<sup>2</sup>C perspective, these blocks can be written if not locked by the I<sup>2</sup>C section lock.

Table below shows the location of the 128-bit AES Table 25 keys in the configuration memory.

Table 25. Key location

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
20h	1020h	KEY0_0 (LSB)	KEY0_1	KEY0_2	KEY0_3

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
21h	1021h	KEY0_4	KEY0_5	KEY0_6	KEY0_7
22h	1022h	KEY0_8	KEY0_9	KEY0_10	KEY0_11
23h	1023h	KEY0_12	KEY0_13	KEY0_14	KEY0_15 (MSB)
24h	1024h	KEY1_0 (LSB)	KEY1_1	KEY1_2	KEY1_3
25h	1025h	KEY1_4	KEY1_5	KEY1_6	KEY1_7
26h	1026h	KEY1_8	KEY1_9	KEY1_10	KEY1_11
27h	1027h	KEY1_12	KEY1_13	KEY1_14	KEY1_15 (MSB)
28h	1028h	KEY2_0 (LSB)	KEY2_1	KEY2_2	KEY2_3
29h	1029h	KEY2_4	KEY2_5	KEY2_6	KEY2_7
2Ah	102Ah	KEY2_8	KEY2_9	KEY2_10	KEY2_11
2Bh	102Bh	KEY2_12	KEY2_13	KEY2_14	KEY2_15 (MSB)
2Ch	102Ch	KEY3_0 (LSB)	KEY3_1	KEY3_2	KEY3_3
2Dh	102Dh	KEY3_4	KEY3_5	KEY3_6	KEY3_7
2Eh	102Eh	KEY3_8	KEY3_9	KEY3_10	KEY3_11
2Fh	102Fh	KEY3_12	KEY3_13	KEY3_14	KEY3_15 (MSB)

If using NFC plain password mode, block addresses 20h to 26h (from NFC perspective) and 1020h to 1026h (from I<sup>2</sup>C perspective) are used to store seven 32-bit passwords as shown in [Table 26](#). Blocks from 27h up to and including 2Fh are not used in plain password mode and are set to RFU.

Default password bytes of Privacy and Destroy password are all 0Fh, Read, Write and EAS/AFI password bytes have a default value of all 00h.

The usage of passwords, read and write access to passwords depends upon NFC Global Crypto Header settings.

**Table 26. Plain Password location**

Block Address		Byte 0	Byte 1	Byte 2	Byte 3	Description
NFC	I <sup>2</sup> C					
20h	1020h	NFC_PWD0_0 (LSB)	NFC_PWD0_1	NFC_PWD0_2	NFC_PWD0_3 (MSB)	Read Password
21h	1021h	NFC_PWD1_0 (LSB)	NFC_PWD1_1	NFC_PWD1_2	NFC_PWD1_3 (MSB)	Write Password
22h	1022h	NFC_PWD2_0 (LSB)	NFC_PWD2_1	NFC_PWD2_2	NFC_PWD2_3 (MSB)	Privacy Password
23h	1023h	NFC_PWD3_0 (LSB)	NFC_PWD3_1	NFC_PWD3_2	NFC_PWD3_3 (MSB)	Destroy Password
24h	1024h	NFC_PWD4_0 (LSB)	NFC_PWD4_1	NFC_PWD4_2	NFC_PWD4_3 (MSB)	EAS/AFI Password
25h	1025h	NFC_PWD5_0 (LSB)	NFC_PWD5_1	NFC_PWD5_2	NFC_PWD5_3 (MSB)	AREA_1 Read Password



Block Address		Byte 0	Byte 1	Byte 2	Byte 3	Description
NFC	I <sup>2</sup> C					
26h	1026h	NFC_PWD6_0 (LSB)	NFC_PWD6_1	NFC_PWD6_2	NFC_PWD6_3 (MSB)	AREA_1 Write Password

For I<sup>2</sup>C password authentication, there are four 32-bit I<sup>2</sup>C passwords (default value of all password bytes is 00h) which can be read or written by the host depending on the I<sup>2</sup>C Key Header configuration.

Table 27. I<sup>2</sup>C Password location

Block Address		Byte 0	Byte 1	Byte 2	Byte 3	Description
NFC	I <sup>2</sup> C					
33h	1033h	I2C_PWD0_0 (LSB)	I2C_PWD0_1	I2C_PWD0_2	I2C_PWD0_3 (MSB)	Read Password
34h	1034h	I2C_PWD1_0 (LSB)	I2C_PWD1_1	I2C_PWD1_2	I2C_PWD1_3 (MSB)	Write Password
35h	1035h	I2C_PWD2_0 (LSB)	I2C_PWD2_1	I2C_PWD2_2	I2C_PWD2_3 (MSB)	AREA_1 Read Password
36h	1036h	I2C_PWD3_0 (LSB)	I2C_PWD3_1	I2C_PWD3_2	I2C_PWD3_3 (MSB)	AREA_1 Write Password

Password authentication from I<sup>2</sup>C perspective is done by using I<sup>2</sup>C write command with the password to the I<sup>2</sup>C Key authenticate location. This will make the I<sup>2</sup>C logic to enter the Authenticated state if the key matches with the respective key.

Table 28. I<sup>2</sup>C Key Authenticate Password location

Block Address	Byte 0	Byte 1	Byte 2	Byte 3	Description
1096h	I2C_PWD0_0_AUTH (LSB)	I2C_PWD0_1_AUTH	I2C_PWD0_2_AUTH	I2C_PWD0_3_AUTH (MSB)	Read Authenticate
1097h	I2C_PWD1_0_AUTH (LSB)	I2C_PWD1_1_AUTH	I2C_PWD1_2_AUTH	I2C_PWD1_3_AUTH (MSB)	Write Authenticate
1098h	I2C_PWD2_0_AUTH (LSB)	I2C_PWD2_1_AUTH	I2C_PWD2_2_AUTH	I2C_PWD2_3_AUTH (MSB)	AREA_1 Read Authenticate
1099h	I2C_PWD3_0_AUTH (LSB)	I2C_PWD3_1_AUTH	I2C_PWD3_2_AUTH	I2C_PWD3_3_AUTH (MSB)	AREA_1 Write Authenticate

### 8.1.3.10 I<sup>2</sup>C Key Header

The Host interface can access the user memory with I<sup>2</sup>C READ and I<sup>2</sup>C WRITE commands. The memory access may be protected by plain password depending on the access conditions.

The programming of I<sup>2</sup>C Key Header can be done in only one direction from lower state to higher state and it is irreversible.

The NFC interface can access this block with READ CONFIG and WRITE CONFIG commands if not locked by NFC section lock. Once locked as per table below, I2C\_KH cannot be updated from the NFC interface anymore.

Table below shows the location of the I<sup>2</sup>C Key Header Configuration in the configuration memory:

Table 29. I<sup>2</sup>C Key Header (I2C\_KH) location

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
30h	1030h	I2C_KH		RFU	

Table 30. I<sup>2</sup>C Key Header Values

Value	Status	Description
81h	Not active (default)	The related I <sup>2</sup> C passwords, I <sup>2</sup> C Protection Pointer and I <sup>2</sup> C Protection Pointer Condition can be read and written with the I <sup>2</sup> C READ and I <sup>2</sup> C WRITE commands. The related password is not active and cannot be used.
C3h	Active	The I <sup>2</sup> C passwords are active. After authentication with write key the I2C_PP, I2C_PPC and Key header can be written and I2C Keys can be read or written.
E7h	Active and locked	The I <sup>2</sup> C key is active and locked and cannot be modified any longer. The related key cannot be read or written with the I <sup>2</sup> C READ or I <sup>2</sup> C WRITE command. The I2C_PP and I2C_PPC can be read with the READ command, but writing with the WRITE command is not possible any longer.
all other	Disabled	Key is disabled and cannot be used

8.1.3.11 I<sup>2</sup>C Protection Pointer and Condition

The I<sup>2</sup>C Protection pointer (I2C\_PP) defines the address of the user memory where the user memory below PP\_AREA\_1 is divided into AREA\_0-L and AREA\_0-H, two arbitrarily sized sections with independent access conditions, which are defined by condition byte (I2C\_PPC).

The NFC interface can access this configuration block with READ CONFIG and WRITE CONFIG commands if not locked by NFC section lock.

Table 31. I<sup>2</sup>C Protection Pointer and Configuration location

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
31h	1031h	I2C_PP	I2C_PPC	RFU	RFU

The protection pointer address (I2C\_PP) defines the base address where the user memory below PP\_ARA\_1 is divided into higher and lower sections.

- The memory address below protection pointer address is called I<sup>2</sup>C AREA\_0-L.
- The memory address above and including protection pointer address is called I<sup>2</sup>C AREA\_0-H
- I<sup>2</sup>C AREA\_0-H ends, where AREA\_1 starts. Note it is the same position (PP\_AREA\_1) as from NFC perspective (see [Section 8.1.3.24](#)).

Below is an example where the memory is divided into to areas with I2C\_PP address set to 0050h. In this example PP\_AREA\_1 is pointing outside the user EEPROM, which means I<sup>2</sup>C AREA\_0-H ends at the end of the user memory.

Default value of I2C\_PP is FFh.

Table 32. I<sup>2</sup>C Memory organization example

Block	Byte 0	Byte 1	Byte 2	Byte 3	Description
0000h					AREA_0-L

Block	Byte 0	Byte 1	Byte 2	Byte 3	Description
0001h					
0002h					
...	...	...	...	...	
004Eh					
004Fh					
0050h					AREA_0-H
0051h					
:	:	:	:	:	
01FEh					
01FFh	RFU				Note: Counter page from NFC perspective

If the protection pointer address is set to block 0, then the entire user memory is defined as AREA\_0-H.

The protection pointer address and protection pointer configuration can be only changed under following conditions:

- I<sup>2</sup>C Key Header Configuration status is "Not Active"
- I<sup>2</sup>C Key Header Configuration status is "Active and read/write after authentication" with read and write privilege key I<sup>2</sup>C authentication has been executed before successfully.

The protection pointer address and protection pointer configuration cannot be changed when I<sup>2</sup>C configuration status is "Active and Locked". This status is irreversible.

Configuration area and SRAM access can be protected by I<sup>2</sup>C authentication when related bits in SRAM\_CONF\_PROT are set to 1b.

The default value of I2C\_PPC is 00h. Which means full access to AREA\_0-L and AREA\_0-H.

**Table 33. I<sup>2</sup>C Protection Pointer Configuration (I2C\_PPC)**

Bit	Name	Value	Description
7	RFU	0b	
6	RFU	0b	
5	WRITE_AREA_H	0b	AREA_0-H is not write protected
		1b	AREA_0-H is write protected
4	REA_AREA_H	0b	AREA_0-H is not read protected
		1b	AREA_0-H is read protected
3	RFU	0b	
2	RFU	0b	
1	WRITE_AREA_L	0b	AREA_0-L is not write protected
		1b	AREA_0-L is write protected
0	READ_AREA_L	0b	AREA_0-L is not read protected
		1b	AREA_0-L is read protected

8.1.3.12 I<sup>2</sup>C Authentication Limit Counter

The I<sup>2</sup>C Authentication Limit Counter is a feature to limit the maximum number of failed authentications from I<sup>2</sup>C perspective. The counter is incremented for failed password write to block 1096h to 1099h. The counter resets on positive authentication.

The NFC interface can access this block with READ CONFIG and WRITE CONFIG commands if not locked by NFC section lock.

Table 34. I<sup>2</sup>C Authentication Limit Counter (I2C\_AUTH\_LIMIT) location

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
32h	1032h	I2C_AUTH_LIMIT (LSB)	I2C_AUTH_LIMIT (MSB)	RFU	

Byte 0 of Block 32h is LSB and Byte 1 is MSB of the I<sup>2</sup>C Authentication Limit counter value. Default value of I2C\_AUTH\_LIMIT is 00h.

The start value for the Authentication Limit can be preset with an I<sup>2</sup>C WRITE command if

- I2C Key header is not locked or Not active
- I2C Key header is Active then after valid authentication with a write key.

The Authentication limit is enabled with the most significant bit of Byte 1 is set to 1b. The remaining 15 bit of I2C\_AUTH\_LIMIT are defining the preset value.

Example:

- 8000h enables and presets the authentication limit to 0, which means the maximum number of failed authentications before a successful is required again is 32767.

**Remark:** The absolute maximum authentication limit value is FFEh before a positive authentication is required, otherwise the authentication is irreversibly locked (no longer available).

As soon as the value of the Authentication limit reaches

- FFFFh: no further authentication possible any longer. This status is irreversible.

8.1.3.13 Configuration

Different features can be configured with CONFIG bits. The effect does not take place in the current session. The effect takes place after POR. All config bits can be read and written from both interfaces.

Table 35. Configuration Bytes Location (CONFIG)

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
37h	1037h	CONFIG_0	CONFIG_1	CONFIG_2	RFU

On POR, all CONFIG bits are copied to CONFIG\_REG (see [Section 8.1.4.2](#)).

Table 36. Configuration Definition (CONFIG\_0)

Bit	Name	Value	Description
7	SRAM_COPY_EN	0b	SRAM copy on POR disabled (default)
		1b	SRAM copy on POR enabled

Bit	Name	Value	Description
6 to 4	RFU	0b	
3	EH_MODE	00b	RFU
		01b	
2	EH_MODE	10b	Energy harvesting optimized for low field strength (default)
		11b	Energy harvesting optimized for high field strength
1	LOCK_SESSION_REG	0b	NFC Write access to all session register (default)
		1b	No NFC write access to session registers A3h to A7h
0	AUTO_STANDBY_MODE_EN	0b	Normal Operation Mode (Default)
		1b	IC enters standby mode after boot if there is no RF field present automatically.

Table 37. Configuration Definition (CONFIG\_1)

Bit	Name	Value	Description
7	EH_ARBITER_MODE_EN	0b	In energy harvesting use case, ARBITER_MODE needs to be set with session registers after startup (default)
		1b	ARBITER_MODE is set automatically in any case after startup
6	ALM_PLM	0b	PLM
		1b	ALM mode when supplied by VCC else PLM (default)
5	USE_CASE_CONF	00b	I <sup>2</sup> C slave (default)
		01b	I <sup>2</sup> C master
4	USE_CASE_CONF	10b	GPIO/PWM
		11b	All host interface functionality disabled and pads are in 3-state mode
3	ARBITER_MODE	00b	Normal Mode (default)
		01b	SRAM mirror mode
2	ARBITER_MODE	10b	SRAM passes through mode
		11b	SRAM PHDC mode
1	SRAM_ENABLE	0b	SRAM is not accessible (default)
		1b	SRAM is available (when VCC supplied)
0	PT_TRANSFER_DIR	0b	Data transfer direction is I <sup>2</sup> C to NFC (default)
		1b	Data transfer direction is NFC to I <sup>2</sup> C

Table 38. Configuration Definition (CONFIG\_2)

Bit	Name	Value	Description
7	GPIO1_IN	00b	Receiver disabled

Bit	Name	Value	Description
6		01b	Plain input with weak pull-up
		10b	Plain input
		11b	Plain input with weak pull-down (Default)
5	GPIO0_IN	00b	Receiver disabled (Default)
		01b	Plain input with weak pull-up
		10b	Plain input
4		11b	Plain input with weak pull-down (Default)
3	EXTENDED_COMMANDS_SUPPORTED	0b	Extended commands are disabled
		1b	Extended commands are supported (Default)
2	LOCK_BLOCK_COMMAND_SUPPORTED	0b	Lock block commands are disabled
		1b	Lock block commands are supported (Default)
1	GPIO1_SLEW_RATE	0b	Low Speed GPIO
		1b	High Speed GPIO (Default)
0	GPIO0_SLEW_RATE	0b	Low Speed GPIO
		1b	High Speed GPIO (Default)

### 8.1.3.14 Synchronization Block

These bytes define the block address of the user memory being a terminator block. Whenever there is read or written to the block address as specified in SYNCH\_DATA\_BLOCK from the NFC interface the respective status bit is set. And also ED pin is set accordingly if configured to detect SYNCH\_DATA\_BLOCK access.

Table 39. Synchronization Block Bytes Location (SYNCH\_DATA\_BLOCK)

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
38h	1038h	SYNCH_DATA_BLOCK (LSB)	SYNCH_DATA_BLOCK (MSB)	RFU	

Table 40. Synchronization Block Register Bytes Location (SYNCH\_DATA\_BLOCK\_REG)

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
A2h	10A2h	SYNCH_DATA_BLOCK_REG (LSB)	SYNCH_DATA_BLOCK_REG (MSB)	RFU	

### 8.1.3.15 Pulse Width Modulation and GPIO configuration

These configuration bytes define the various configuration bits for GPIO/PWM use case (see Section 8.1.3.13). All features can be configured from NFC and I<sup>2</sup>C perspective as long NTAG 5 boost is configured for slave mode. For details refer to Section 8.3.3 and Section 8.3.4.

Table 41. PWM and GPIO Configuration Location (PWM\_GPIO\_CONFIG)

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
39h	1039h	PWM_GPIO_CONFIG_0	PWM_GPIO_CONFIG_1	RFU	

Table 42. PWM and GPIO Configuration Definition (PWM\_GPIO\_CONFIG\_0)

Bit	Name	Value	Description
7	SDA_GPIO1_OUT_STATUS	0b	Output status on pad is LOW (default)
		1b	Output status on pad is HIGH
6	SCL_GPIO0_OUT_STATUS	0b	Output status on pad is LOW (default)
		1b	Output status on pad is HIGH
5 to 4	RFU	00b	
3	SDA_GPIO1	0b	Output (Default)
		1b	Input
2	SCL_GPIO0	0b	Output (Default)
		1b	Input
1	SDA_GPIO1_PWM1	0b	GPIO (Default)
		1b	PWM
0	SCL_GPIO0_PWM0	0b	GPIO (Default)
		1b	PWM

Table 43. PWM and GPIO Configuration Definition (PWM\_GPIO\_CONFIG\_1 and PWM\_GPIO\_CONFIG\_1\_REG)

Bit	Name	Value	Description
7	PWM1_PRESCALE	00b	Pre-scalar configuration for PWM1 channel (default 00b)
6			
5	PWM0_PRESCALE	00b	Pre-scalar configuration for PWM0 channel (default 00b)
4			
3	PWM1_RESOLUTION_CONF	00b	6-bit resolution (default)
		01b	8-bit resolution
		10b	10-bit resolution
		11b	12-bit resolution
2	PWM0_RESOLUTION_CONF	00b	6-bit resolution (default)
		01b	8-bit resolution
		10b	10-bit resolution
		11b	12-bit resolution

### 8.1.3.16 Pulse Width Modulation duty cycle settings

Details can be found in PWM Mode section (see [Section 8.3.4](#)).

**Table 44. Pulse Width Modulation Duty Cycle Configuration Location (PWMx\_ON and PWMx\_OFF)**

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
3Ah	103Ah	PWM0_ON (LSB)	PWM0_ON (MSB)	PWM0_OFF (LSB)	PWM0_OFF (MSB)
3Bh	103Bh	PWM1_ON (LSB)	PWM1_ON (MSB)	PWM1_OFF (LSB)	PWM1_OFF (MSB)

**Table 45. Pulse Width Modulation Duty Cycle Session Register Location (PWMx\_ON and PWMx\_OFF)**

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
A4h	10A4h	PWM0_ON_REG (LSB)	PWM0_ON_REG (MSB)	PWM0_OFF_REG (LSB)	PWM0_OFF_REG (MSB)
A5h	10A5h	PWM1_ON_REG (LSB)	PWM1_ON_REG (MSB)	PWM1_OFF_REG (LSB)	PWM1_OFF_REG (MSB)

**Table 46. Pulse Width Modulation ON time Configuration Definition (PWMx\_ON and PWMx\_ON\_REG)**

Bit	Name	Default Value	Description
7 to 4	RFU	all 0b	
3 to 0	PWMx_ON (MSB)	all 0b	coded time PWM channel x output will be asserted HIGH
7 to 0	PWMx_ON (LSB)	all 0b	

**Table 47. Pulse Width Modulation OFF time Configuration Definition (PWMx\_OFF and PWMx\_OFF\_REG)**

Bit	Name	Default Value	Description
7 to 4	RFU	all 0b	
3 to 0	PWMx_OFF (MSB)	all 0b	coded time PWM channel x output will be asserted LOW
7 to 0	PWMx_OFF (LSB)	all 0b	

PWM on and off times are coded by using maximum 12 bits. To code for example, PWM0\_ON as 0123h, PWM0\_ON (LSB) is set to 23h, and PWM0\_ON (MSB) is set to 01h.

### 8.1.3.17 Watch Dog Timer settings

I<sup>2</sup>C Watch Dog Timer settings can be adjusted and enabled via configuration bytes from both interfaces. Related session registers are read only. Watch Dog Timer 16-bit default value is 0848h. Details can be found in WDT section (see [Section 8.3.1.3](#)).



Table 48. Watch Dog Timer Configuration Location (WDT\_CONFIG)

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
3Ch	103Ch	WDT_CONFIG (LSB)	WDT_CONFIG (MSB)	WDT_ENABLE	SRAM_COPY_BYTES

Table 49. Watch Dog Timer Configuration Register Location (WDT\_CONFIG\_REG)

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
A6h	10A6h	WDT_CONFIG_REG (LSB)	WDT_CONFIG_REG (MSB)	WDT_EN_REG	RFU

Table 50. Watch Dog Timer Enable Definition (WDT\_ENABLE and WDT\_EN\_REG)

Bit	Name	Default Value	Description
7 to 5	RFU	all 0b	
4	RFU	1b	
3 to 1	RFU	all 0b	
0	WDT_ENABLE	0b	Watch Dog Timer disabled
		1b	Watch Dog Timer enabled (default)

### 8.1.3.18 Energy harvesting settings

Energy harvesting configuration controls the behavior of the energy harvesting output pin. If DISABLE\_POWER\_CHECK is 0b and energy harvesting is enabled with EH\_ENABLE is 1b, only when the applied field strength is sufficient to generate configured minimum output load current (EH\_IOUT\_SEL) and voltage (EH\_VOUT\_SEL), the energy harvesting output is enabled.

If energy harvesting will be enabled during the session with register bits, EH\_IOUT\_SEL and EH\_VOUT\_SEL define the needed output power. However, DISABLE\_POWER\_CHECK and EH\_ENABLE bits need to be set to 0b in this case.

Details can be found in energy harvesting section (see [Section 8.5](#)).

Table 51. Energy harvesting Configuration Location (EH\_CONFIG)

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
3Dh	103Dh	EH_CONFIG	RFU	ED_CONFIG	RFU

Table 52. Energy harvesting Configuration Value Definition (EH\_CONFIG)

Bit	Name	Value	Description
7	RFU		
6	EH_VOUT_I_SEL	000b	>0.4 mA (Default)

Bit	Name	Value	Description
5		001b	>0.6 mA
		010b	>1.4 mA
		011b	>2.7 mA
		100b	>4.0 mA
		101b	>6.5 mA
4		110b	>9.0 mA
		111b	>12.5 mA
3	DISABLE_POWER_CHECK	0b	Only if sufficient power can be harvested, VOUT will be enabled (default)
		1b	Power level will not be checked, VOUT will be enabled immediately after startup
2	EH_VOUT_V_SEL	00b	1.8 V (Default)
		01b	2.4 V
1		10b	3 V
		11b	RFU
0	EH_ENABLE	0b	Energy harvesting disabled (default)
		1b	Energy harvesting enabled

8.1.3.19 Event detection pin configuration settings

Event detection and field detection functionality define the behavior of the active low ED pin depending on various events. As this pin is an open-drain active low implementation, ED pin state ON means that signal is LOW and OFF means that signal is HIGH. More details can be found in ED section (see [Section 8.3.2](#)).

Table 53. Event Detection Configuration Location (ED\_CONFIG)

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
3Dh	103Dh	EH_CONFIG	RFU	ED_CONFIG	RFU

Table 54. Event Detection Configuration Register Location (ED\_CONFIG\_REG)

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
A8h	10A8h	ED_CONFIG_REG	RFU		

Table 55. Event Detection Definition (ED\_CONFIG and ED\_CONFIG\_REG)

Bit	Name	Value	ED pin state	Description
7 to 4	RFU	0000b	N/A	
3 to 0	Disable ED	0000b	OFF	Event detection pin disabled (default)

NTAG 5 boost - NFC Forum-compliant I<sup>2</sup>C bridge for tiny devices

Bit	Name	Value	ED pin state	Description
	NFC Field detect	0001b	ON	NFC field present
			OFF	NFC field absent
	PWM	0010b	ON	Pulse width modulation signal during OFF period
			OFF	Pulse width modulation signal during On period
	I <sup>2</sup> C to NFC pass-through	0011b	ON	Last byte of SRAM data has been read via NFC; host can access SRAM again
			OFF	<ul style="list-style-type: none"> <li>Last byte written by I<sup>2</sup>C, or</li> <li>NFC off, or</li> <li>V<sub>CC</sub> is off</li> </ul>
	NFC to I <sup>2</sup> C pass-through	0100b	ON	Last byte written by NFC; host can read data from SRAM
			OFF	<ul style="list-style-type: none"> <li>Last byte has been read from I<sup>2</sup>C, or</li> <li>NFC off, or</li> <li>V<sub>CC</sub> off</li> </ul>
	Arbiter lock	0101b	ON	Arbiter locked access to NFC interface
			OFF	Lock to NFC interface released
	NDEF Message TLV Length	0110b	ON	Length byte(block 1, byte 1) is not ZERO
			OFF	Length byte (block 1, byte1) is ZERO
	Stand-by mode	0111b	ON	IC is NOT in standby mode
			OFF	IC is in standby mode
	WRITE command indication	1000b	ON	Start of programming cycle during WRITE command
			OFF	Start of response to WRITE command or NFC off
	READ command indication	1001b	ON	Start of read cycle during READ command
			OFF	<ul style="list-style-type: none"> <li>End of read access, or</li> <li>NFC off</li> </ul>
	Start of command indication	1010b	ON	Start of (any) command
			OFF	<ul style="list-style-type: none"> <li>End of response to command, or</li> <li>NFC off</li> </ul>
	READ from SYNCH_BLOCK	1011b	ON	Data read from SYNCH_BLOCK
			OFF	Event needs to be cleared by setting b0 of ED_RESET_REG to 1b or NFC off
	WRITE to SYNCH_BLOCK	1100b	ON	Data written to SYNCH_BLOCK
			OFF	Event needs to be cleared by setting b0 of ED_RESET_REG to 1b or NFC off
	Software Interrupt	1101b	ON	1101b written to ED_CONFIG
			OFF	Event needs to be cleared by setting b0 of ED_RESET_REG to 1b
	RFU	1110b	N/A	
	RFU	1111b	N/A	

Table 56. Event Detection Clear Register Location (ED\_INTR\_CLEAR\_REG)

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
ABh	10ABh	ED_INTR_CLEAR_REG		RFU	

Table 57. Event Detection Clear Register (ED\_INTR\_CLEAR\_REG)

Bit	Name	Value	Description
7 to 1	RFU	all 0b	
0	ED_INTR_CLEAR	1b	write 1b to release ED pin

ED pin is cleared i.e. released when writing 01h to the ED clear register. The bit gets automatically cleared after clearing the ED pin.

### 8.1.3.20 I<sup>2</sup>C slave configuration settings

I<sup>2</sup>C slave functionality can be configured with the following configuration registers from both interfaces.

Table 58. I<sup>2</sup>C Slave Configuration Location

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
3Eh	103Eh	I2C_SLAVE_ADDR	I2C_SLAVE_CONFIG	I2C_MASTER_SCL_LOW	I2C_MASTER_SCL_HIGH

Table 59. I<sup>2</sup>C Slave Configuration Definition (I2C\_SLAVE\_ADDR)

Bit	Name	Value	Description
7	RFU	0b	
6 to 0	I2C_SLAVE_ADDR	54h	I <sup>2</sup> C slave address used in slave configuration

Table 60. I<sup>2</sup>C Slave Configuration Definition (I2C\_SLAVE\_CONFIG)

Bit	Name	Value	Description
7 to 3	RFU	00000b	
2	I2C_S_REPEATED_START	0b	Slave does not reset if repeated start is received (Default)
		1b	I <sup>2</sup> C slave resets internal state machine on repeated start
1	RFU	0b	
0	RFU	0b	

### 8.1.3.21 I<sup>2</sup>C master clock configuration settings

I<sup>2</sup>C Master baud rate and SCL high and low can be configured by the related configuration values.

Details about the formula can be found in I<sup>2</sup>C master section.

Table 61. I<sup>2</sup>C Master Clock Settings Configuration Location (I2C\_MASTER\_CONFIG)

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
3Eh	103Eh	I2C_SLAVE_ADDR	I2C_SLAVE_CONFIG	I2C_MASTER_SCL_LOW	I2C_MASTER_SCL_HIGH

Table 62. I<sup>2</sup>C Master Clock Configuration Definition (I2C\_MASTER\_SCL\_LOW)

Bit	Name	Default Value	Description
7	RFU		
6 to 0	I2C_MASTER_SCL_LOW	09h	I <sup>2</sup> C master configuration SCL low period. Default: 400 kHz, 41%duty cycle.

Table 63. I<sup>2</sup>C Master Clock Configuration Definition (I2C\_MASTER\_SCL\_HIGH)

Bit	Name	Default Value	Description
7	RFU		
6 to 0	I2C_MASTER_SCL_HIGH	02h	I <sup>2</sup> C master configuration SCL HIGH period. Default: 400 kHz, 41%duty cycle.

### 8.1.3.22 Device security configuration bytes

NTAG 5 boost features scale-able security. The level of security can be selected with DEV\_SEC\_CONFIG byte.

SRAM\_CONF\_PROT is described in [Section 8.1.3.23](#).

PP\_AREA\_1 is described in [Section 8.1.3.24](#).

Table 64. Device Security Configuration Byte location

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
3Fh	103Fh	DEV_SEC_CONFIG	SRAM_CONF_PROT	PP_AREA_1 (LSB)	PP_AREA_1 (MSB)

The IC security features can be enabled or disabled or can choose different security options available from NFC perspective:

- AES authentication scheme
- Plain password feature

Table 65. Device Security Byte Definition (DEV\_SEC\_CONFIG)

Bit	Name	Value	Description
7 to 5	Security Lock	010b	block 3Fh with device security configuration bytes is locked
		101b	block 3Fh is writeable (default)
		other	RFU
4	RFU	00b	

Bit	Name	Value	Description
3	NFC Security		
2		other	RFU
1		010b	AES
0		101b	plain password (default)

### 8.1.3.23 SRAM and Configuration protection

Access to complete SRAM and blocks 37h to 54h of configuration area can be restricted with SRAM\_CONFIG\_PROT byte.

Table 66. SRAM and Configuration Byte location

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
3Fh	103Fh	DEV_SEC_CONFIG	SRAM_CONF_PROT	PP_AREA_1 (LSB)	PP_AREA_1 (MSB)

Table 67. SRAM and Configuration Protection (SRAM\_CONFIG\_PROT)

Bit	Name	Value	Description
7 to 6	RFU	00b	
5	I2C_CONFIG_W	0b	Configuration area is not write protected from I <sup>2</sup> C perspective (default)
		1b	Configuration area is write protected from I <sup>2</sup> C perspective
4	I2C_CONFIG_R	0b	Configuration area is not read protected from I <sup>2</sup> C perspective (Default)
		1b	Configuration area is read protected from I <sup>2</sup> C perspective
3	NFC_SRAM_W	0b	SRAM is not write protected from NFC perspective (Default)
		1b	SRAM is write protected from NFC perspective
2	NFC_SRAM_R	0b	SRAM is not read protected from NFC perspective (Default)
		1b	SRAM is read protected from NFC perspective
1	NFC_CONFIG_W	0b	Configuration area is not write protected from NFC perspective (Default)
		1b	Configuration area is write protected from NFC perspective
0	NFC_CONFIG_R	0b	Configuration area is not read protected from NFC perspective (Default)
		1b	Configuration area is read protected from NFC perspective

### 8.1.3.24 Restricted AREA\_1 pointer

The AREA\_1 Pointer (PP\_AREA\_1) can be configured by directly writing PP\_AREA\_1 byte to configuration memory using WRITE CONFIG command (see [Section 8.2.4.2.2](#)). This 16-bit block address is the same for NFC and I<sup>2</sup>C perspective. The default value is FFFFh.

Table 68. Restricted AREA\_1 Pointer location

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
3Fh	103Fh	DEV_SEC_CONFIG	SRAM_CONF_PROT	PP_AREA_1 (LSB)	PP_AREA_1 (MSB)

In below example, NFC protection pointer (NFC\_PP\_AREA\_0H) is set to 50h and PP\_AREA\_1 is set to 0060h, e.g. PP\_AREA\_1 (LSB) is 60h and PP\_AREA\_1 (MSB) is 00h. This example illustrates the view from NFC perspective.

From I<sup>2</sup>C perspective Area 1 will be the same, however Area 0-L and Area 0-H may look different depending on I<sup>2</sup>C protection pointer.

Table 69. Memory organization example from NFC perspective

Block	Byte 0	Byte 1	Byte 2	Byte 3	Description
0000h					AREA_0-L
0001h					
0002h					
...	...	...	...	...	
004Fh					
0050h					AREA_0-H
0051h					
...	...	...	...	...	
005Fh					
0060h					AREA_1
0061h					
...	...	...	...	...	
01FEh					
01FFh	C0	C1	00h	PROT	Counter

### 8.1.3.25 Active NFC configuration

For active communication, dependent on the antenna size, a proper matching circuit is needed and the IC needs to be configured to keep load modulation amplitude values within ISO/IEC 15693 and NFC Forum Type 5 Tag specification. The voltage on LA and LB shall not exceed 1.8 V in ALM mode. How to configure ALM in your application is described in Antenna Design Guide application note [AN12339](#).

Table 70. Active NFC Configuration location

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
40h	1040h	ALM_CONF_00	ALM_CONF_01	ALM_CONF_02	ALM_CONF_03
41h	1041h	ALM_LUT_00	ALM_LUT_01	ALM_LUT_02	ALM_LUT_03
42h	1042h	ALM_LUT_04	ALM_LUT_05	ALM_LUT_06	ALM_LUT_07
43h	1043h	ALM_LUT_08	ALM_LUT_09	ALM_LUT_10	ALM_LUT_11
44h	1044h	ALM_LUT_12	ALM_LUT_13	ALM_LUT_14	ALM_LUT_15

Table 71. ALM Configuration 0 (ALM\_CONF\_00)

Bit	Name	Value	Description
7 to 6	RFU	10b	
5 to 3	typical field detect threshold rms value @ 13.56 MHz	000b	35 mV
		001b	44 mV
		010b	53 mV (default)
		011b	62 mV
		100b	71 mV
		101b	80 mV
		110b	89 mV
		111b	98 mV
2 to 0	RFU	001b	

Table 72. ALM Configuration 1 (ALM\_CONF\_01)

Bit	Name	Value	Description
7	RFU	1b	
6 to 2	Static phase offset	xxxxxb	Static phase offset (default = 11111b)
1 to 0	RFU	00b	

Table 73. ALM Configuration 2 (ALM\_CONF\_02)

Bit	Name	Value	Description
7 to 0	RFU	5Eh	

Table 74. ALM Configuration 3 (ALM\_CONF\_03)

Bit	Name	Value	Description
7 to 6	RFU	11b	
5 to 3	PLL track delay	xxxb	Extra delay in 6.78 MHz clock cycles (default = 001b)
2 to 0	RFU	111b	

Table 75. ALM Look-up-table (ALM\_LUT\_dd)

Bit	Name	Value	Description
7 to 5	Dynamic phase adjust	xxxb	adjust phase in 11.25° steps
4	Enable BPSK	0b	ASK
		1b	BPSK
3 to 0	RON	0000b	1574 Ω
		0001b	716 Ω



Bit	Name	Value	Description
		0010b	414 Ω
		0011b	248 Ω
		0100b	123 Ω
		0101b	82 Ω
		0110b	62 Ω
		0111b	49 Ω
		1000b	41 Ω
		1001b	35 Ω
		1010b	31 Ω
		1011b	27 Ω
		1100b	25 Ω
		1101b	22 Ω
		1110b	21 Ω
		1111b	17 Ω

**Table 76. Active NFC Configuration Default Content**

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
40h	1040h	81h	FCh	5Eh	CFh
41h	1041h	1Fh	1Fh	17h	14h
42h	1042h	13h	13h	12h	12h
43h	1043h	10h	10h	F0h	F0h
44h	1044h	F0h	F0h	F0h	F0h

**8.1.3.26 Application Family Identifier**

The Application Family Identifier (AFI) represents the type of application targeted by the device and is used to extract from all the ICs present only the ICs meeting the required application criteria.

AFI can be configured using WRITE AFI command (see [Section 8.2.4.9.1](#)) or directly writing AFI byte to configuration memory using WRITE CONFIG command (see [Section 8.2.4.2.2](#)).

Default value of AFI is 00h.

From I<sup>2</sup>C perspective, this byte can be accessed with normal READ and WRITE commands as long as not locked by I<sup>2</sup>C section locks.

**Table 77. Application Family Identifier (AFI) location**

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
55h	1055h	AFI	RFU		

**8.1.3.27 Data Storage Format Identifier**

The Data Storage Format Identifier may indicate how the data is structured in the VICC memory. If not used, this byte shall be set to 00h, which is the default value.

The Data Storage Format Identifier (DSFID) can be configured using WRITE DSFID command (see [Section 8.2.4.9.3](#)) or directly writing DSFID byte to configuration memory using WRITE CONFIG command (see [Section 8.2.4.2.2](#)).

From I<sup>2</sup>C perspective, this byte can be accessed with normal READ and WRITE commands as long as not locked by I<sup>2</sup>C section locks.

**Table 78. Data Storage Format Identifier (DSFID) location**

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
56h	1056h	DSFID			RFU

**8.1.3.28 Electronic Article Surveillance ID**

The Electronic Article Surveillance ID (EAS ID) can be configured using WRITE EAS ID (see [Section 8.2.4.9.10](#)) command or directly writing EAS\_ID byte to configuration memory using WRITE CONFIG command (see [Section 8.2.4.2.2](#)).

Default value of EAS\_ID is 0000h.

From I<sup>2</sup>C perspective, this byte can be accessed with normal READ and WRITE commands as long as not locked by I<sup>2</sup>C section locks.

**Table 79. Electronic Article Surveillance ID (EASID) location**

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
57h	1057h	EAS_ID (LSB)	EAS_ID (MSB)		RFU

**8.1.3.29 NFC protection pointer**

The NFC protection pointer (NFC\_PP\_AREA\_0H) can be configured using PROTECT PAGE command (see [Section 8.2.4.3.6](#)) or directly writing NFC\_PP\_AREA\_0H byte to configuration memory using WRITE CONFIG command (see [Section 8.2.4.2.2](#)).

Default value is FFh.

**Table 80. NFC Protection Pointer (NFC PP) location**

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
58h	1058h	NFC_PP_AREA_0H	NFC_PPC	RFU	RFU

In below example, NFC protection pointer is set to 50h. PP\_AREA\_1 is out side of the EEPROM area in this example. This example illustrates the view from NFC perspective.

**Table 81. Memory organization example**

Block	Byte 0	Byte 1	Byte 2	Byte 3	Description
0000h					AREA_0-L

Block	Byte 0	Byte 1	Byte 2	Byte 3	Description
0001h					
0002h					
...	...	...	...	...	
004Fh					
0050h					AREA_0-H
0051h					
...	...	...	...	...	
01FEh					
01FFh	C0	C1	00h	PROT	Counter

**8.1.3.30 NFC Protection Pointer Conditions**

The NFC Protection Pointer Conditions (NFC PPC) can be configured using PROTECT PAGE command (see [Section 8.2.4.3.6](#)) or directly writing NFC\_PPC byte to configuration memory using WRITE CONFIG command (see [Section 8.2.4.2.2](#)) as defined in table below.

**Table 82. NFC Protection Pointer Conditions (NFC\_PPC) location**

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
58h	1058h	NFC_PP_AREA_0H	NFC_PPC	RFU	RFU

**Table 83. NFC Protection Pointer Configuration (NFC\_PPC)**

Bit	Name	Value	Description
7	RFU	0b	
6	RFU	0b	
5	Write AREA_0_H	0b	AREA_0-H is not write protected (Default)
		1b	AREA_0-H is write protected
4	Read AREA_0_H	0b	AREA_0-H is not read protected (Default)
		1b	AREA_0-H is read protected
3	RFU	0b	
2	RFU	0b	
1	Write AREA_0_L	0b	AREA_0-L is not write protected (Default)
		1b	AREA_0-L is write protected
0	Read AREA_0_L	0b	AREA_0-L is not read protected (Default)
		1b	AREA_0-L is read protected

**8.1.3.31 NFC lock bytes**

User blocks can be blocked from writing by the NFC interface. These bits are one time programmable. Once written to 1b, they cannot be changed back to 0b. Each bit locks

one block of user memory area (e.g., bit 0 of byte 0 locks block 0). These bytes can be written by NFC and I<sup>2</sup>C. The access to these bytes for the particular interface can be restricted by configuring the device SECTION\_LOCK (see [Table 86](#)).

**Table 84. NFC Lock Block Configuration location**

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
6Ah	106Ah	NFC_LOCK_BL00	NFC_LOCK_BL01	RFU	RFU
...	...	...	...	RFU	RFU
89h	1089h	NFC_LOCK_BL62	NFC_LOCK_BL63	RFU	RFU

**8.1.3.32 I<sup>2</sup>C lock bytes**

User blocks can be blocked from writing by the I<sup>2</sup>C interface. These bits are one time programmable. Once written 1b, they cannot be changed back to 0b. Each bit locks 4 blocks of user memory area (e.g., bit 0 of byte 0 locks blocks 0, 1, 2 and 3). These bytes can be written by NFC and I<sup>2</sup>C. The access to these bytes for the particular interface can be restricted by configuring the device configuration section locks bytes (see [Table 86](#)). I2C\_LOCK\_BL00 – bit 0 – will lock user blocks 0,1,2,3 from I<sup>2</sup>C perspective.

**Table 85. I<sup>2</sup>C Lock Block Configuration location**

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
8Ah	108Ah	I2C_LOCK_BL00	I2C_LOCK_BL01	RFU	RFU
...	...	...	...	RFU	RFU
91h	1091h	I2C_LOCK_BL14	I2C_LOCK_BL15	RFU	RFU

**8.1.3.33 Device configuration section lock bytes**

Lock bits are provided to lock different sections of the configuration area. 16 bits for each interface are provided to define access conditions for different sections of the configuration area.

First column in the configuration memory table (see [Table 10](#)) defines the affiliated blocks of each section.

**Table 86. Device configuration section lock bytes location**

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
92h	1092h	NFC_LOCK_0	RFU		
93h	1093h	NFC_LOCK_1	RFU		
94h	1094h	I2C_LOCK_0	RFU		
95h	1095h	I2C_LOCK_1	RFU		

These section lock configurations are provided to allow customer to initialize NTAG 5 boost during customer configuration from either I<sup>2</sup>C or NFC interface. After the configuration is done, it is recommended to write the appropriate lock conditions and lock the device configuration bytes.

These lock bytes take the highest priority above all locks. Different section access conditions have to be chosen appropriately, so that the other interface does not change and corrupt the other interface security configuration.

If I2C\_LOCK\_0 and NFC\_LOCK\_0 bits are set to 1b, then the lock bytes cannot be updated and gets locked permanently. If any interface is not locked i.e. any one of the I2C\_LOCK\_0 and NFC\_LOCK\_0 bits are 0b, then the particular interface can unlock the other.

**Table 87. NFC configuration section lock byte 0 definition (NFC\_SECTION\_LOCK\_0)**

Bit	Name	Value	Description
7	Section 7	0b	Section 7 is writable by NFC
		1b	Section 7 is not writable by NFC
6	Section 6	0b	Section 6 is writable by NFC
		1b	Section 6 is not writable by NFC
5	Section 5	0b	Section 5 is writable by NFC
		1b	Section 5 is not writable by NFC
4	Section 4	0b	Section 4 is writable by NFC
		1b	Section 4 is not writable by NFC
3	Section 3	0b	Section 3 is writable by NFC
		1b	Section 3 is not writable by NFC
2	Section 2	0b	Section 2 is writable by NFC
		1b	Section 2 is not writable by NFC
1	Section 1	0b	Section 1 is writable by NFC
		1b	Section 1 is not writable by NFC
0	Section 0	0b	Section 0 is writable by NFC
		1b	Section 0 is not writable by NFC

**Table 88. NFC configuration section lock Byte 1 definition (NFC\_SECTION\_LOCK\_1)**

Bit	Name	Value	Description
7	Section 8	0b	Section 8 is writable by NFC
		1b	Section 8 is not writeable by NFC
6	Section 6	0b	Section 6 is readable by NFC
		1b	Section 6 is not readable by NFC
5	Section 5	0b	Section 5 is readable by NFC
		1b	Section 5 is not readable by NFC
4	Section 4	0b	Section 4 is readable by NFC
		1b	Section 4 is not readable by NFC
3	Section 3	0b	Section 3 is readable by NFC
		1b	Section 3 is not readable by NFC
2	Section 2	0b	Section 2 is readable by NFC

Bit	Name	Value	Description
		1b	Section 2 is not readable by NFC
1	Section 1	0b	Section 1 is readable by NFC
		1b	Section 1 is not readable by NFC
0	Section 0	0b	Section 0 is readable by NFC
		1b	Section 0 is not readable by NFC

Table 89. I<sup>2</sup>C configuration section lock byte 0 definition (I2C\_SECTION\_LOCK\_0)

Bit	Name	Value	Description
7	Section 7	0b	Section 7 is writable by I <sup>2</sup> C
		1b	Section 7 is not writable by I <sup>2</sup> C
6	Section 6	0b	Section 6 is writable by I <sup>2</sup> C
		1b	Section 6 is not writable by I <sup>2</sup> C
5	Section 5	0b	Section 5 is writable by I <sup>2</sup> C
		1b	Section 5 is not writable by I <sup>2</sup> C
4	Section 4	0b	Section 4 is writable by I <sup>2</sup> C
		1b	Section 4 is not writable by I <sup>2</sup> C
3	Section 3	0b	Section 3 is writable by I <sup>2</sup> C
		1b	Section 3 is not writable by I <sup>2</sup> C
2	Section 2	0b	Section 2 is writable by I <sup>2</sup> C
		1b	Section 2 is not writable by I <sup>2</sup> C
1	Section 1	0b	Section 1 is writable by I <sup>2</sup> C
		1b	Section 1 is not writable by I <sup>2</sup> C
0	Section 0	0b	Section 0 is writable by I <sup>2</sup> C
		1b	Section 0 is not writable by I <sup>2</sup> C

Table 90. I<sup>2</sup>C configuration section lock byte 1 definition (I2C\_SECTION\_LOCK\_1)

Bit	Name	Value	Description
7	Section 8	0b	Section 8 writable by I <sup>2</sup> C
		1b	Section 8 is not writeable by I <sup>2</sup> C
6	Section 6	0b	Section 6 is readable by I <sup>2</sup> C
		1b	Section 6 is not readable by I <sup>2</sup> C
5	Section 5	0b	Section 5 is readable by I <sup>2</sup> C
		1b	Section 5 is not readable by I <sup>2</sup> C
4	Section 4	0b	Section 4 is readable by I <sup>2</sup> C
		1b	Section 4 is not readable by I <sup>2</sup> C
3	Section 3	0b	Section 3 is readable by I <sup>2</sup> C

Bit	Name	Value	Description
		1b	Section 3 is not readable by I <sup>2</sup> C
2	Section 2	0b	Section 2 is readable by I <sup>2</sup> C
		1b	Section 2 is not readable by I <sup>2</sup> C
1	Section 1	0b	Section 1 is readable by I <sup>2</sup> C
		1b	Section 1 is not readable by I <sup>2</sup> C
0	Section 0	0b	Section 0 is readable by I <sup>2</sup> C
		1b	Section 0 is not readable by I <sup>2</sup> C

Note: Section 7 (default SRAM content) and Section 8 (Device configuration lock bytes) are always readable.

In case of not readable and/or not writeable, IC responds with an NACK from I<sup>2</sup>C perspective and with an error from NFC perspective, when trying to access locked sections.

### 8.1.4 Session registers

After POR, the content of the configuration settings (see [Section 8.1.3](#)) is loaded into the session register. The values of session registers can be changed during a session. Change to session registers take effect immediately, but only for the current communication session. After POR, the session registers values will again contain the configuration register values as before.

To change the default behavior, changes to the related configuration bytes are needed, but the related effect will only be visible after the next POR.

Session registers starting from block A3h until the end may be write protected with LOCK\_REGISTER bit.

Reading and writing the session registers via I<sup>2</sup>C can only be done with READ REGISTER and WRITE REGISTER commands.

Most of the parameters are defined in the configuration memory section.

Table 91. Session Register Location

Block Address		Byte 0	Byte 1	Byte 2	Byte 3	Remark
NFC	I <sup>2</sup> C					
A0h	10A0h	STATUS_REG		RFU		Status Register (see <a href="#">Section 8.1.4.1</a> )
A1h	10A1h	CONFIG_REG				Configuration (see <a href="#">Section 8.1.4.2</a> )
A2h	10A2h	SYNC_DATA_BLOCK_REG		RFU		Block Address (see <a href="#">Section 8.1.4.3</a> )
A3h	10A3h	PWM_GPIO_CONFIG_REG		RFU		PWM and GPIO Configuration (see <a href="#">Section 8.1.4.4</a> )

Block Address		Byte 0	Byte 1	Byte 2	Byte 3	Remark
NFC	I <sup>2</sup> C					
A4h	10A4h	PWM0_ON_OFF_REG				PWM1 Configuration (see <a href="#">Section 8.1.4.5</a> )
A5h	10A5h	PWM1_ON_OFF_REG				PWM1 Configuration (see <a href="#">Section 8.1.4.5</a> )
A6h	10A6h	WDT_CONFIG_REG			RFU	Watch Dog Timer Configuration (see <a href="#">Section 8.1.4.6</a> )
A7h	10A7h	EH_CONFIG_REG	RFU			Energy Harvesting Configuration (see <a href="#">Section 8.1.4.7</a> )
A8h	10A8h	ED_CONFIG_REG	RFU			Event detection functionality (see <a href="#">Section 8.1.4.8</a> )
A9h	10A9h	I2C_SLAVE_CONFIG_REG		RFU		I <sup>2</sup> C Slave Configuration (see <a href="#">Section 8.1.4.9</a> )
AAh	10AAh	RESET_GEN_REG	RFU			Reset Register (see <a href="#">Section 8.1.4.10</a> )
ABh	10ABh	ED_INTR_CLEAR_REG	RFU			Clear Event Detection (see <a href="#">Section 8.1.4.11</a> )
ACh	10ACh	I2C_M_S_ADD_REG	I2C_M_LEN_REG	ALM_STATUS_REG	RFU	I <sup>2</sup> C Master Configuration (see <a href="#">Section 8.1.4.12</a> )
ADh	10ADh	I2C_M_STATUS_REG	RFU			I <sup>2</sup> C Master Configuration (see <a href="#">Section 8.1.4.12</a> )
A Eh	10AEh	RFU				
AFh	10AFh	RFU				

### 8.1.4.1 Status register

Different status of NTAG 5 boost can be known by reading status register. The status register can be read by READ\_CONFIG.

Some of the registers may be cleared. Setting status bits to 1b is not possible at all.

**Table 92. Status Register Location**

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
A0h	10A0h	STATUS0_REG	STATUS1_REG	RFU	



Table 93. Status 0 Register

Bit	Name	Access		Value	Description
		NFC	I <sup>2</sup> C		
7	EEPROM_WR_BUSY	R	R	0b	EEPROM is not busy
				1b	EEPROM is busy (programming cycle ongoing)
6	EEPROM_WR_ERROR	R/W	R/W	0b	all data written successfully
				1b	EEPROM write error happened. This bit needs to be cleared.
5	SRAM_DATA_READY	R	R	0b	data not yet ready used in pass-through mode
				1b	data ready, used in pass-through mode
4	SYNCH_BLOCK_WRITE	R	R/W	0b	data has NOT been written to SYNCH_BLOCK
				1b	data had been written to SYNCH_BLOCK
3	SYNCH_BLOCK_READ	R	R/W	0b	data has NOT been read from SYNCH_BLOCK
				1b	data had been read from SYNCH_BLOCK
2	PT_TRANSFER_DIR	R	R	0b	I <sup>2</sup> C to NFC pass-through direction
				1b	NFC to I <sup>2</sup> C pass-through direction
1	VCC_SUPPLY_OK	R	R	0b	VCC supply not present
				1b	VCC supply available
0	NFC_FIELD_OK	R	R	0b	No NFC field present
				1b	NFC field present

Table 94. Status 1 Register

Bit	Name	Access		Value	Description
		NFC	I <sup>2</sup> C		
7	VCC_BOOT_OK	R	R	0b	VCC boot not done
				1b	VCC boot done
6	NFC_BOOT_OK	R	R	0b	NFC boot not done
				1b	NFC boot done
5	ACTIVE_NFC_OK	R	R	0b	ALM RF not OK
				1b	ALM RF OK
4	GPIO1_IN_STATUS	R	R	0b	GPIO_1 input is LOW
				1b	GPIO_1 input is HIGH
3	GPIO0_IN_STATUS	R	R	0b	GPIO_0 input is LOW

Bit	Name	Access		Value	Description
		NFC	I <sup>2</sup> C		
				1b	GPIO_0 input is HIGH
2	ALM_PLM	R	R	0b	Only Passive Load Modulation supported
				1b	Active Load Modulation supported
1	I2C_IF_LOCKED	R	R/W	0b	I <sup>2</sup> C Interface not locked by arbiter
				1b	Arbiter locked to I <sup>2</sup> C
0	NFC_IF_LOCKED	R	R	0b	NFC interface not locked by arbiter
				1b	Arbiter locked to NFC

### 8.1.4.2 Configuration register

On POR all CONFIG bits (see [Section 8.1.3.13](#)) are copied to CONFIG\_REG. Some of these features may be changed with CONFIG\_REG bits during a session. These features are valid at once. Most of them are just read only from NFC and/or I<sup>2</sup>C perspective.

Table 95. Configuration Register Location (CONFIG\_REG)

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
A1h	10A1h	CONFIG_0_REG	CONFIG_1_REG	CONFIG_2_REG	RFU

Table 96. Configuration Definition (CONFIG\_0\_REG)

Bit	Name	Access		Value	Description
		NFC	I <sup>2</sup> C		
7	SRAM_COPY_EN	R	R	0b	SRAM copy on POR disabled
				1b	SRAM copy on POR enabled
6	RFU	R	R	0b	
5	DISABLE_NFC	R	R/W	0b	NFC interface enabled
				1b	NFC interface disabled, no response to NFC commands
4	RFU	R	R	0b	
3	RFU	R	R	0b	
2	RFU	R	R	0b	
1	RFU	R	R	0b	
0	AUTO_STANDBY_MODE_EN	R	R/W	0b	Normal Operation Mode
				1b	IC enters standby mode after boot if there is no RF field present automatically.

Table 97. Configuration Definition (CONFIG\_1\_REG)

Bit	Name	Access		Value	Description
		NFC acc.	I <sup>2</sup> C acc.		
7	RFU	R	R	0b	
6	ALM_PLM	R	R	0b	PLM mode only
				1b	ALM mode in case of V <sub>CC</sub> supply
5	USE_CASE_CONF	R	R	00b	I <sup>2</sup> C slave
				01b	I <sup>2</sup> C master
				10b	GPIO/PWM
4				11b	All host interface functionality disabled and pads are in 3-state mode
3	ARBITER_MODE	R	R/W	00b	Normal Mode
				01b	SRAM mirror mode
				10b	SRAM passes through mode
2				11b	SRAM PHDC mode
1	SRAM_ENABLED	R	R	0b	SRAM is not accessible
				1b	SRAM is accessible
0	PT_TRANSFER_DIR	R	R/W	0b	Data transfer direction is I <sup>2</sup> C to NFC
				1b	Data transfer direction is NFC to I <sup>2</sup> C

Table 98. Configuration Definition (CONFIG\_2\_REG)

Bit	Name	Access		Value	Description
		NFC acc.	I <sup>2</sup> C acc.		
7	GPIO1_IN	R	R	00b	Receiver disabled
				01b	Plain input with weak pull-up
				10b	Plain input
6				11b	Plain input with weak pull-down
5	GPIO0_IN	R	R	00b	Receiver disabled
				01b	Plain input with weak pull-up
				10b	Plain input
4				11b	Plain input with weak pull-down
3	EXTENDED_COMMANDS_SUPPORTED	R	R	0b	Extended commands are disabled
				1b	Extended commands are supported
2	LOCK_BLOCK_COMMAND_SUPPORTED	R	R	0b	Lock block commands are disabled
				1b	Lock block commands are supported
1	GPIO1_SLEW_RATE	R	R	0b	Low-Speed GPIO
				1b	High-Speed GPIO
0	GPIO0_SLEW_RATE	R	R	0b	Low-Speed GPIO

Bit	Name	Access		Value	Description
		NFC acc.	I <sup>2</sup> C acc.		
				1b	High-Speed GPIO

### 8.1.4.3 Synchronization block register

The terminator block maybe changed during one session from both interfaces (see [Section 8.1.3.14](#)). Both interfaces do have read and write access.

### 8.1.4.4 Pulse Width Modulation and GPIO configuration register

These session register bytes define the various configurations for GPIO/PWM use case (see [Section 8.1.3.13](#)). From NFC perspective IN\_STATUS bits are read only, all others maybe changed during a session. From I<sup>2</sup>C perspective, all bits are read only. For details refer to [Section 8.3.3](#) and [Section 8.3.4](#).

Table 99. PWM and GPIO Configuration Register Location (PWM\_GPIO\_CONFIG\_REG)

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
A3h	10A3h	PWM_GPIO_CONFIG_0_REG	PWM_CONFIG_1_REG	RFU	

Table 100. PWM and GPIO Configuration Register Definition (PWM\_GPIO\_CONFIG\_0\_REG)

Bit	Name	Access		Value	Description
		NFC acc.	I <sup>2</sup> C acc.		
7	SDA_GPIO1_OUT_STATUS	R/W	R	0b	Output status on pad is LOW
				1b	Output status on pad is HIGH
6	SCL_GPIO0_OUT_STATUS	R/W	R	0b	Output status on pad is LOW
				1b	Output status on pad is HIGH
5	SDA_GPIO1_SDA_IN_STATUS	R	R	0b	Input status
				1b	
4	SCL_GPIO0_IN_STATUS	R	R	0b	Input status
				1b	
3	SDA_GPIO1	R/W	R	0b	Output
				1b	Input
2	SCL_GPIO0	R/W	R	0b	Output
				1b	Input
1	SDA_GPIO1_PWM1	R/W	R	0b	GPIO
				1b	PWM
0	SCL_GPIO0_PWM0	R/W	R	0b	GPIO
				1b	PWM

Table 101. PWM and GPIO Configuration Register Definition (PWM\_GPIO\_CONFIG\_1\_REG)

Bit	Name	Access		Value	Description
		NFC Acc.	I <sup>2</sup> C Acc.		
7	PWM1_PRESCALE	R/W	R	00b	Pre-scalar configuration for PWM1 channel
6					
5	PWM0_PRESCALE	R/W	R	00b	Pre-scalar configuration for PWM0 channel
4					
3	PWM1_RESOLUTION_CONF	R/W	R	00b	6-bit resolution
2				01b	8-bit resolution
				10b	10-bit resolution
				11b	12-bit resolution
1	PWM0_RESOLUTION_CONF	R/W	R	00b	6-bit resolution
0				01b	8-bit resolution
				10b	10-bit resolution
				11b	12-bit resolution

**8.1.4.5 Pulse Width Modulation duty cycle register**

The PWM duty cycle maybe changed during one session from NFC perspective (see [Section 8.1.3.16](#)).

As long as the I<sup>2</sup>C slave use case is enabled, these settings can also be done from I<sup>2</sup>C perspective.

**8.1.4.6 Watch Dog Timer register**

Watch Dog Timer register settings are read only (see [Section 8.1.3.17](#)).

**8.1.4.7 Energy harvesting register**

Energy harvesting registers may be used to enable energy harvesting during one NFC session. In this case, EH\_ENABLE bit of EH\_CONFIG byte in block 3Dh is set to 0b. Required EH\_VOUT\_I\_SEL and EH\_VOUT\_V\_SEL need to be set in that EH\_CONFIG byte. Desired energy harvesting mode (EH\_MODE) needs to be configured in CONFIG\_0 byte of block 37h. In case of energy harvesting is enabled already during boot (EH\_ENABLE bit of EH\_CONFIG is 1b), or energy harvesting is not used at all, this register byte gives no information.

Setting EH\_TRIGGER to 1b is needed to trigger power detection.

Polling for bit EH\_LOAD\_OK should be used to check, if sufficient energy is available. Only if EH\_LOAD\_OK = 1b, energy harvesting may be enabled via session registers by writing 09h to this byte.

There is only read only access from I<sup>2</sup>C perspective.

Details can be found in energy harvesting section (see [Section 8.5](#)).

Table 102. Energy Harvesting Configuration Register Location (EH\_CONFIG\_REG)

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
A7h	10A7h	EH_CONFIG_REG		RFU	

Table 103. Energy Harvesting Register Value Definition (EH\_CONFIG\_REG)

Bit	Name	Access		Value	Description
		NFC Acc.	I <sup>2</sup> C Acc.		
7	EH_LOAD_OK	R	R	0b	Field is not sufficient to provide configured power on V <sub>OUT</sub> . Do not enable energy harvesting.
				1b	Minimum desired energy available. V <sub>OUT</sub> may be enabled. As soon as EH_ENABLE is set to 1b, this bit gets cleared automatically.
6 to 4	RFU	R	R		
3	EH_TRIGGER	R/W	R	0b	When reading, this byte this bit is RFU and the value is undefined and may be 0b or 1b.
				1b	When writing to this byte, this bit needs to be set to 1b always
2 to 1	RFU	R	R		
0	EH_ENABLE	R/W	R	0b	Energy Harvesting disabled (default)
				1b	Energy Harvesting enabled

#### 8.1.4.8 Event detection register

Event detection and field detection functionality define the behavior of the ED pin depending on various events. Indicated event may be changed during one session from both interfaces. More details can be found in ED section (see [Section 8.3.2](#)).

#### 8.1.4.9 I<sup>2</sup>C slave register settings

I<sup>2</sup>C slave settings can be read out from both interfaces.

I<sup>2</sup>C interface can be disabled via NFC interface only. Repeated start functionality can be enabled via I<sup>2</sup>C interface only.

Table 104. I<sup>2</sup>C Slave Configuration Register Location

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
A9h	10A9h	I2C_SLAVE_ADDR_REG	I2C_SLAVE_CONFIG_REG	RFU	

Table 105. I<sup>2</sup>C Slave Configuration Definition (I2C\_SLAVE\_ADDR\_REG)

Bit	Name	Access		Value	Description
		NFC	I <sup>2</sup> C		
7	RFU	R	R	0b	
6 to 0	I <sup>2</sup> C_SLAVE_ADDR	R	R	54h	I <sup>2</sup> C slave address used in slave configuration

Table 106. I<sup>2</sup>C Slave Configuration Definition (I2C\_SLAVE\_CONFIG\_REG)

Bit	Name	Access		Value	Description
		NFC	I <sup>2</sup> C		
7 to 5	RFU	R	R	000b	
4	I2C_WDT_EXPIRED	R	R	0b	WDT did not expire in previous transaction
				1b	Previous transaction was not successful. WDT expired. This bit gets cleared automatically, when new transaction is triggered.
3	I2C_SOFT_RESET	R/W	R/W	0b	Setting this bit to 1b, resets the I <sup>2</sup> C state machine and releases the SCL/SDA lines. This bit gets cleared automatically.
2	I2C_S_REPEATED_START	R	R/W	0b	Slave does not reset if repeated start is received (Default)
				1b	I <sup>2</sup> C slave resets internal state machine on repeated start
1	RFU	R	R	0b	
0	DISABLE_I <sup>2</sup> C (see <a href="#">Section 8.3.1.1.6</a> )	R/W	R	0b	I <sup>2</sup> C interface enabled (default)
				1b	I <sup>2</sup> C interface disabled

#### 8.1.4.10 System reset generation

System reset can be generated by both the interfaces by writing to RESET\_GEN\_REG register using WRITE CONFIG command (see [Section 8.2.4.2.2](#)). Writing E7h will trigger the system reset. This byte gets automatically reset after the system reset.

Table 107. RESET\_GEN\_REG location

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
AAh	10AAh	RESET_GEN_REG		RFU	

#### 8.1.4.11 Clear event detection register

Event detection pin is cleared i.e. released when writing 01h to the Clear Event Detection Register from NFC or I<sup>2</sup>C interface. The bit gets cleared after releasing the ED pin automatically. Other values are RFU.

Table 108. ED\_INTR\_CLEAR\_REG location

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
ABh	10ABh	ED_INTR_CLEAR_REG		RFU	

8.1.4.12 I<sup>2</sup>C master status registers

I<sup>2</sup>C master status can be checked by reading I<sup>2</sup>C master session register bits from NFC perspective. There is no write access to these registers. Details can be found in I<sup>2</sup>C master section.

NOTE: In I<sup>2</sup>C master mode there is no access to the registers from I<sup>2</sup>C perspective. As ALM\_STATUS register is only accessible from I<sup>2</sup>C perspective, lookup table for ALM needs to be filled before switching to master mode.

Table 109. I<sup>2</sup>C Master Configuration Register Location

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
ACh	10ACh	I2C_M_S_ADD_REG	I2C_M_LEN_REG	ALM_STATUS_REG	RFU
ADh	10ADh	I2C_M_STATUS_REG		RFU	

Table 110. I<sup>2</sup>C Slave Address used in I<sup>2</sup>C master transaction (I2C\_M\_S\_ADD\_REG)

Bit	Name	Access		Value	Description
		NFC	I <sup>2</sup> C		
7	I2C_M_RS_EN	R	R	0b	STOP condition generated at the end of transaction
				1b	no STOP condition generated
6 to 0	I2C_M_S_ADD	R	R	00h	I <sup>2</sup> C slave of last addressed slave

Table 111. I<sup>2</sup>C Master Data Length Definition (I2C\_M\_LEN\_REG)

Bit	Name	Value	Description
7 to 0	I2C_M_LEN_REG	all 0b	Codes the data length written to or read from I <sup>2</sup> C slave during last I <sup>2</sup> C transaction. Data length in bytes is I2C_M_LEN_REG plus 1.

Table 112. I<sup>2</sup>C Master Status Definition (I2C\_M\_STATUS\_REG)

Bit	Name	Access		Value	Description
		NFC	I <sup>2</sup> C		
7 to 4	RFU	R	R	0000b	
3	I2C_M_WDT_EXPIRED	R	R	0b	WDT did not expire in last transaction
				1b	WDT expired in last transaction. This bit resets automatically, when new transaction is triggered.



Bit	Name	Access		Value	Description
		NFC	I <sup>2</sup> C		
2 to 1	I2C_M_TRANS_STATUS	R	R	00b	Reset value, automatically when new transaction starts
				01b	Address NAK
				10b	Data NAK
				11b	Transaction successful
0	I2C_M_BUSY	R	R	0b	I <sup>2</sup> C Master interface ready. No transaction ongoing
				1b	I <sup>2</sup> C Master interface busy. Transaction ongoing

### 8.1.4.13 ALM status registers

ALM status register maybe used to read back field digitization value from I<sup>2</sup>C perspective. There is no write access to this register.

NOTE: From NFC perspective ALM\_STATUS\_REG is RFU.

Table 113. ALM Status Register Location (ALM\_STATUS\_REG)

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
ACh	10ACh	I2C_M_S_ADD_REG	I2C_M_LEN_REG	ALM_STATUS_REG RFU from NFC perspective	RFU

Table 114. ALM Status Register (ALM\_STATUS\_REG)

Bit	Name	Value	Description
7 to 4	RFU	0000b	
3 to 0	FIELD_DIGIT	0000b	ALM field digitization input

### 8.1.5 SRAM

For frequently changing data, a volatile memory of 256 bytes with unlimited write endurance is built in. The 256 bytes are mapped in a similar way as done in the EEPROM, i.e., 256 bytes are seen as 64 pages of 4 bytes.

SRAM is only available when supplied by V<sub>CC</sub> and SRAM\_ENABLE bit is set to 1b.

The SRAM can be mirrored in the User Memory from block 00h to 3Fh for access from both interfaces as illustrated in the table below. This allows using NFC Forum read and write commands to access the SRAM.

The lock block condition (e.g. user EEPROM blocks are set to read-only) is not valid for the mirrored SRAM.

The access conditions (e.g. first blocks of user EEPROM are password protected) are valid for the mirrored SRAM, too, for both interfaces.

The access conditions for READ SRAM and WRITE SRAM commands can be restricted with SRAM\_CONF\_PROT security bits.

From I<sup>2</sup>C perspective, SRAM is located in blocks 2000h to 203Fh and can be accessed at any time without any protection.

NOTE: SRAM values are not initialized during boot and may have arbitrary data.

Table 115. SRAM mirroring

Block Address		Byte 0	Byte 1	Byte 2	Byte 3	Description
NFC	I <sup>2</sup> C					
00h	0000h	SRAM				When SRAM is mirrored to user memory, READ and WRITE commands address SRAM
01h	0001h	:	:	:	:	
02h	0002h					
03h	0003h					
:	:					
3Fh	003Fh	SRAM				From block 40h onwards, READ and WRITE commands always address EEPROM
40h	0040h	EEPROM				
:	:	:	:	:	:	
1FEh	01FEh	EEPROM				
1FFh	N/A	C0	C1	00h	PROT	

In the pass-through mode, READ and WRITE SRAM commands should be used to transfer data between the two interfaces.

After POR the SRAM can be pre-loaded with default data (e.g. NDEF message) depending on SRAM\_COPY\_ENABLE bit in CONFIG register. If SRAM\_COPY\_ENABLE bit and SRAM\_ENABLE both are set to 1b, then the copy feature will be triggered after POR. The startup time depends upon the number of bytes (maximum 64) to be copied.

In the table below, an example is illustrated where 48 bytes are copied to the SRAM on POR.

Table 116. SRAM mirroring with default content

Block	Byte 0	Byte 1	Byte 2	Byte 3	Description
0	SRAM_DEF00	SRAM_DEF01	SRAM_DEF02	SRAM_DEF03	When SRAM is mirrored and SRAM_COPY_BYTES is equal to 30h, 48 bytes will be copied to SRAM after POR automatically
:	...SRAM_DEF...				
11	SRAM_DEF44	SRAM_DEF45	SRAM_DEF46	SRAM_DEF47	
12	SRAM				Rest of mirrored SRAM will be random data
:	:	:	:	:	
63	SRAM				
64	EEPROM				From block 64 onwards again EEPROM will be addressed
:	:	:	:	:	
510	EEPROM				
511	C0	C1	00h	PROT	Counter

Table 117. SRAM COPY BYTES (SRAM\_COPY\_BYTES)

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
3Ch	103Ch	WDT_CONFIG		WDT_ENABLE	SRAM_COPY_BYTES

Table 118. SRAM\_COPY\_BYTES Definition

Bit	Name	Description
7 to 6	RFU	RFU
5 to 0	SRAM_COPY_BYTES	6-bit length field. Defines the number of bytes (SRAM_COPY_BYTES + 1) to be copied from CONFIG memory to SRAM at startup. Maximum is 3Fh. Higher values are RFU. Default all bits are 0b.

Table 119. SRAM Default Content location

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
45h	1045h	SRAM_DEF00	SRAM_DEF01	SRAM_DEF02	SRAM_DEF03
...	...	...SRAM_DEF...			
54h	1054h	SRAM_DEF60	SRAM_DEF61	SRAM_DEF62	SRAM_DEF63

## 8.2 NFC interface

The definition of the NFC interface is according to the [ISO/IEC 15693](#) and [NFC Forum Type 5 Tag](#). The details of passive and active communication mode are described in [Section 8.2.1](#) and [Section 8.2.2](#).

Supported bitrates for different modes and communication directions are listed in tables below.

Table 120. Bit rates from reader to tag

Mode	Condition	1.66 kbps	26 kbps
passive	NFC only	yes	yes
passive	V <sub>CC</sub> supplied	yes	yes
active	V <sub>CC</sub> supplied	no	yes

Table 121. Bit rates from tag to reader

Mode	Condition	6 kbps	26 kbps	53 kbps
passive	single subcarrier	yes	yes	yes
passive	dual subcarrier	yes	yes	no
active	single subcarrier	no	yes	yes
active	dual subcarrier	no	no	no

**8.2.1 Passive communication mode**

Main uses cases for passive communication mode are Smart Metering, Home automation and in the box configuration. With antenna sizes of Class 4 or bigger, energy harvesting on the one side and long-distance read/write access to the EEPROM is possible in a very efficient way.

**8.2.2 Active communication mode**

When the application has only limited space for the antenna and/or metal environment is close by, only with active communication mode a reasonable reading/writing distance can be achieved. In this mode, the IC needs to be V<sub>CC</sub> and V<sub>CC\_TX</sub> supplied. However, the NTAG 5 boost is designed very energy efficiently. Average standby current for NFC sensing is typically less than 10 µA. In hard power down the IC consumes typically less than 0.25 µA.

The output driver as illustrated below is a simple push-pull implementation directly connected to V<sub>CC\_TX</sub>. When in receive mode, it is configured to high Z.

Design guidelines and recommendations how to find proper matching circuit can be found in the antenna design guide app note [AN12339](#).

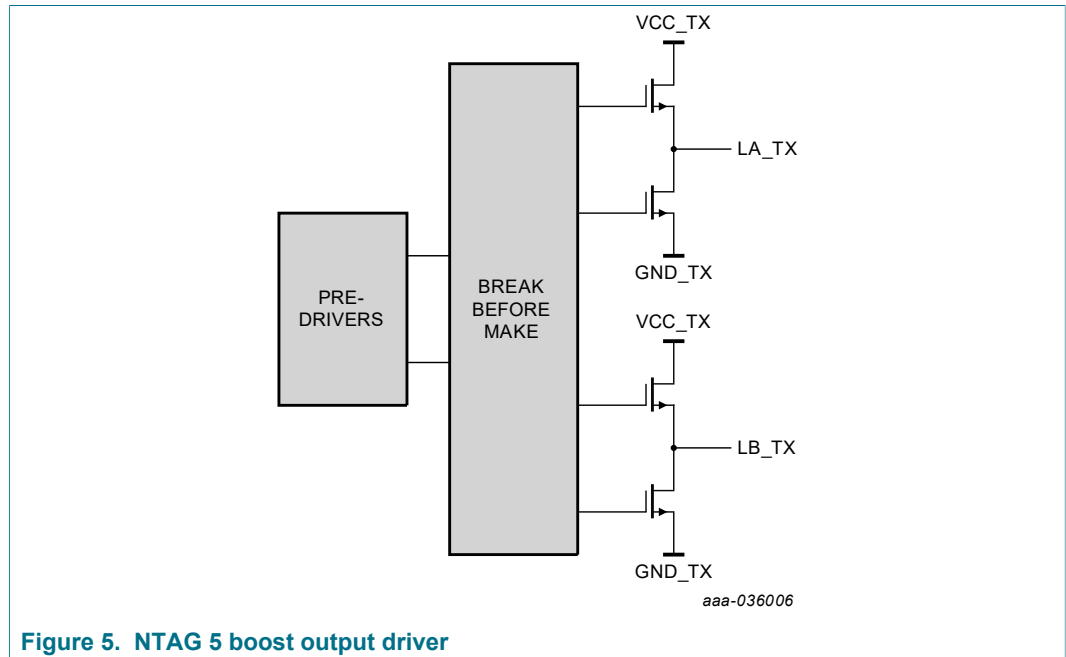


Figure 5. NTAG 5 boost output driver

**8.2.3 State diagram and state transitions**

The state diagram illustrates the different states and state transitions of NTAG 5 boost. The SELECTED SECURE state is only available, when AES security is enabled.

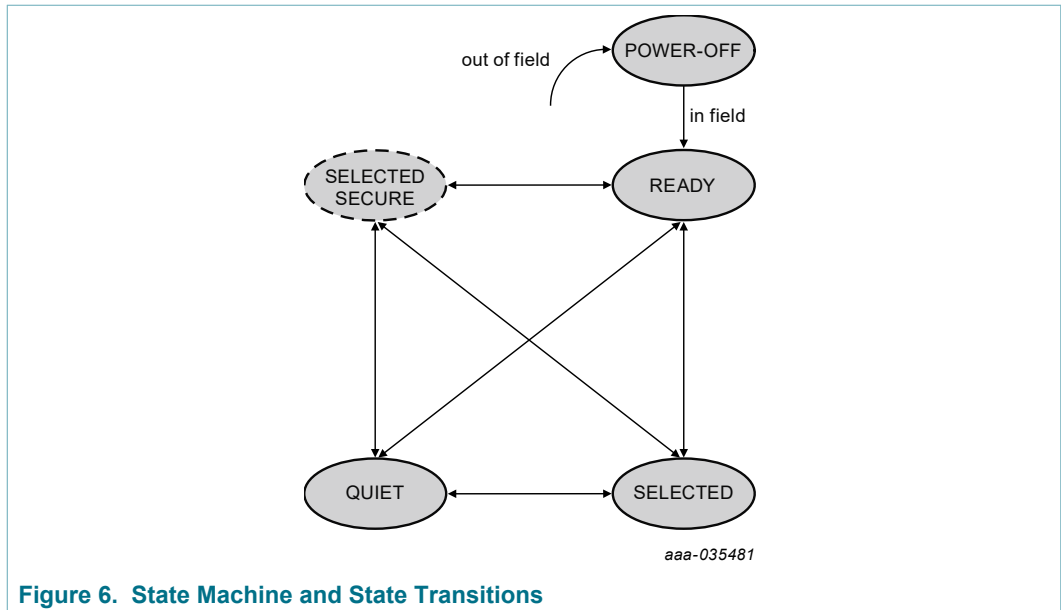


Figure 6. State Machine and State Transitions

8.2.3.1 POWER-OFF state

8.2.3.1.1 State transitions from and to POWER-OFF state

If NFC field is switched off or below, H<sub>MIN</sub> NTAG 5 boost goes to POWER-OFF state.

POWER-OFF state will be left to READY state no later than 1 ms after NTAG 5 boost is powered by an NFC field greater than H<sub>MIN</sub>.

NOTE: When loading default data to mirrored SRAM, start-up time, dependent on the pre-loaded bytes, might be greater than 1 ms.

8.2.3.2 READY state

8.2.3.2.1 Transitions between READY and SELECTED state

Transition from READY to SELECTED state is done when

- receiving a SELECT command with a matching UID

8.2.3.2.2 Transitions between READY and QUIET state

Transition from READY to QUIET state is done when

- receiving a STAY QUIET command with a matching UID
- receiving a (FAST) INVENTORY READ command (extended mode) with Quiet\_Flag set

8.2.3.2.3 Transitions between READY and SELECTED SECURE state

Transition from READY to SELECTED SECURE state is done when

- mutual authentication with AUTHENTICATE command with a matching UID was successful

#### 8.2.3.2.4 Commands which stay in READY state

NTAG 5 boost stays in READY state when

- receiving any other command where Select\_Flag is not set

#### 8.2.3.3 SELECTED state

##### 8.2.3.3.1 Transitions between SELECTED and READY state

Transition from SELECTED to READY state is done by

- receiving a RESET TO READY command where Select\_Flag is set
- receiving a SELECT command with a different UID

##### 8.2.3.3.2 Transitions between SELECTED and QUIET state

Transition from SELECTED to QUIET state is done when

- receiving a STAY QUIET command with a matching UID

##### 8.2.3.3.3 Transitions between SELECTED and SELECTED SECURE state

Transition from SELECTED to SELECTED SECURE state is done by

- mutual authentication with AUTHENTICATE command with Select\_Flag set was successful

##### 8.2.3.3.4 Commands which stay in SELECTED state

NTAG 5 boost stays in SELECTED state when

- receiving any other command where Select\_Flag is set

#### 8.2.3.4 SELECTED SECURE state

##### 8.2.3.4.1 Transitions between SELECTED SECURE and READY state

Transition from SELECTED SECURE to READY state is done by

- receiving a RESET\_TO\_READY command where Select\_Flag is set
- receiving a SELECT command with a different UID
- receiving a CHALLENGE command
- receiving a new AUTHENTICATE command

##### 8.2.3.4.2 Transitions between SELECTED SECURE and QUIET state

Transition from SELECTED SECURE to QUIET state is done when

- receiving a STAY QUIET command with a matching UID

##### 8.2.3.4.3 Transitions between SELECTED SECURE and SELECTED state

Transition from SELECTED SECURE to SELECTED state is done by

- receiving a SELECT command with a matching UID

##### 8.2.3.4.4 Commands which stay in SELECTED SECURE state

NTAG 5 boost stays in SELECTED SECURE state when

- receiving READBUFFER command
- receiving any other command where Select\_Flag is set

### 8.2.3.5 QUIET state

#### 8.2.3.5.1 Transitions between QUIET and READY state

Transition from QUIET to READY state is done by

- receiving a RESET\_TO\_READY command

#### 8.2.3.5.2 Transitions between QUIET and SELECTED state

Transition from QUIET to SELECTED state is done by

- receiving a SELECT command with a matching UID

#### 8.2.3.5.3 Transitions between QUIET and SELECTED SECURE state

Transition from QUIET to SELECTED SECURE state is done when

- mutual authentication with AUTHENTICATE command with a matching UID was successful

#### 8.2.3.5.4 Commands which stay in QUIET state

NTAG 5 boost stays in QUIET state when

- receiving any other command where Addressed\_Flag is set AND Inventory\_Flag is not set

## 8.2.4 Command set

ISO/IEC 15693 mandatory commands are

- INVENTORY
- STAY QUIET

NFC Forum Type 5 Tag mandatory commands are

- READ SINGLE BLOCK
- WRITE SINGLE BLOCK
- LOCK SINGLE BLOCK

On top of those, all optional commands of ISO/IEC 15693 are implemented. Several customer-specific commands are implemented to, e.g., improve overall transaction time. These custom commands all use NXP manufacturer code 04h.

A complete list of all supported commands is given in below table.

**Table 122. NFC command set supported by NTAG 5 boost**

Code	ISO/IEC 15693	NFC Forum T5T	Command name
01h	Mandatory	Mandatory	INVENTORY (see <a href="#">ISO/IEC 15693</a> and <a href="#">Digital Protocol</a> )
02h	Mandatory	Mandatory	STAY QUIET (see <a href="#">ISO/IEC 15693</a> ) and <a href="#">Type 5 Tag - SLPV_REQ</a> )
20h	Optional	Mandatory	READ SINGLE BLOCK (see <a href="#">ISO/IEC 15693</a> and <a href="#">Type 5 Tag - READ_SINGLE_BLOCK_REQ</a> )

NTAG 5 boost - NFC Forum-compliant I<sup>2</sup>C bridge for tiny devices

Code	ISO/IEC 15693	NFC Forum T5T	Command name
21h	Optional	Mandatory in READ/WRITE state	WRITE SINGLE BLOCK (see <a href="#">ISO/IEC 15693</a> and <a href="#">Type 5 Tag - WRITE_SINGLE_BLOCK_REQ</a> )
22h	Optional	Optional	LOCK BLOCK (see <a href="#">ISO/IEC 15693</a> and <a href="#">Type 5 Tag - LOCK_SINGLE_BLOCK_REQ</a> )
23h	Optional	Optional	READ MULTIPLE BLOCKS ( <a href="#">ISO/IEC 15693</a> and <a href="#">Type 5 Tag - READ_MULTIPLE_BLOCK_REQ</a> )
25h	Optional	Optional	SELECT (see <a href="#">ISO/IEC 15693</a> and <a href="#">Type 5 Tag - SELECT_REQ</a> )
26h	Optional	Not defined	RESET TO READY (see <a href="#">ISO/IEC 15693</a> )
27h	Optional	Not defined	WRITE AFI (see <a href="#">ISO/IEC 15693</a> )
28h	Optional	Not defined	LOCK AFI (see <a href="#">ISO/IEC 15693</a> )
29h	Optional	Not defined	WRITE DSFID (see <a href="#">ISO/IEC 15693</a> )
2Ah	Optional	Not defined	LOCK DSFID (see <a href="#">ISO/IEC 15693</a> )
2Bh	Optional	Not defined	GET SYSTEM INFORMATION (see <a href="#">ISO/IEC 15693</a> )
2Ch	Optional	Not defined	GET MULTIPLE BLOCK SECURITY STATUS (see <a href="#">ISO/IEC 15693</a> )
2Dh	Optional	Not defined	FAST READ MULTIPLE BLOCKS (see <a href="#">ISO/IEC 15693</a> )
30h	Optional	Mandatory when supporting 2 byte addressing	EXTENDED READ SINGLE BLOCK (see <a href="#">ISO/IEC 15693</a> and <a href="#">Type 5 Tag - EXTENDED_READ_SINGLE_BLOCK_REQ</a> )
31h	Optional	Mandatory when supporting 2 byte addressing in READ/WRITE state	EXTENDED WRITE SINGLE BLOCK (see <a href="#">ISO/IEC 15693</a> and <a href="#">Type 5 Tag - EXTENDED_WRITE_SINGLE_BLOCK_REQ</a> )
32h	Optional	Optional	EXTENDED LOCK BLOCK (see <a href="#">ISO/IEC 15693</a> and <a href="#">Type 5 Tag - EXTENDED_LOCK_SINGLE_BLOCK_REQ</a> )
33h	Optional	Optional	EXTENDED READ MULTIPLE BLOCK ( <a href="#">ISO/IEC 15693</a> and <a href="#">Type 5 Tag - EXTENDED_READ_MULTIPLE_BLOCK_REQ</a> )
35h	Optional	Not defined	AUTHENTICATE (see <a href="#">ISO/IEC 15693</a> ) is only available in AES mode
39h	Optional	Not defined	CHALLENGE (see <a href="#">ISO/IEC 15693</a> ) is only available in AES mode
3Ah	Optional	Not defined	READBUFFER (see <a href="#">ISO/IEC 15693</a> ) is only available in AES mode
3Bh	Optional	Not defined	EXTENDED GET SYSTEM INFORMATION (see <a href="#">ISO/IEC 15693</a> )
3Ch	Optional	Not defined	EXTENDED GET MULTIPLE BLOCK SECURITY STATUS (see <a href="#">ISO/IEC 15693</a> )
3Dh	Optional	Not defined	FAST EXTENDED READ MULTIPLE BLOCKS (see <a href="#">ISO/IEC 15693</a> )
A0h	Custom	Not defined	INVENTORY READ (see <a href="#">Section 8.2.4.5.1</a> )
A1h	Custom	Not defined	FAST INVENTORY READ (see <a href="#">Section 8.2.4.5.2</a> )
A2h	Custom	Not defined	SET EAS (see <a href="#">Section 8.2.4.9.5</a> )
A3h	Custom	Not defined	RESET EAS (see <a href="#">Section 8.2.4.9.6</a> )
A4h	Custom	Not defined	LOCK EAS (see <a href="#">Section 8.2.4.9.7</a> )



Code	ISO/IEC 15693	NFC Forum T5T	Command name
A5h	Custom	Not defined	EAS ALARM (see <a href="#">Section 8.2.4.9.8</a> )
A6h	Custom	Not defined	PROTECT EAS/AFI (see <a href="#">Section 8.2.4.9.9</a> )
A7h	Custom	Not defined	WRITE EAS ID (see <a href="#">Section 8.2.4.9.10</a> )
ABh	Custom	Not defined	GET NXP SYSTEM INFORMATION (see <a href="#">Section 8.2.4.9.14</a> )
B2h	Custom	Not defined	GET RANDOM NUMBER (see <a href="#">Section 8.2.4.3.1</a> )
B3h	Custom	Not defined	SET PASSWORD (see <a href="#">Section 8.2.4.3.2</a> )
B3h	Custom	Not defined	DISABLE NFC PRIVACY (see <a href="#">Section 8.2.4.3.10</a> )
B4h	Custom	Not defined	WRITE PASSWORD (see <a href="#">Section 8.2.4.3.3</a> )
B5h	Custom	Not defined	LOCK PASSWORD (see <a href="#">Section 8.2.4.3.4</a> )
B6h	Custom	Not defined	PROTECT PAGE (see <a href="#">Section 8.2.4.3.6</a> )
B7h	Custom	Not defined	LOCK PAGE PROTECTION CONDITION (see <a href="#">Section 8.2.4.3.7</a> )
B9h	Custom	Not defined	DESTROY (see <a href="#">Section 8.2.4.3.8</a> )
BAh	Custom	Not defined	ENABLE NFC PRIVACY (see <a href="#">Section 8.2.4.3.9</a> )
BBh	Custom	Not defined	64 BIT PASSWORD PROTECTION (see <a href="#">Section 8.2.4.3.5</a> )
BDh	Custom	Not defined	READ SIGNATURE (see <a href="#">Section 8.2.4.7.1</a> )
C0h	Custom	Not defined	READ CONFIGURATION (see <a href="#">Section 8.2.4.2.1</a> )
C1h	Custom	Not defined	WRITE CONFIGURATION (see <a href="#">Section 8.2.4.2.2</a> )
C2h	Custom	Not defined	PICK RANDOM UID (see <a href="#">Section 8.2.4.9.15</a> )
D2h	Custom	Not defined	READ SRAM (see <a href="#">Section 8.2.4.6.1</a> )
D3h	Custom	Not defined	WRITE SRAM (see <a href="#">Section 8.2.4.6.2</a> )
D4h	Custom	Not defined	WRITE I2C (see <a href="#">Section 8.2.4.8.1</a> )
D5h	Custom	Not defined	READ I2C (see <a href="#">Section 8.2.4.8.2</a> )

All command/responses are sent/received in the request/response format as defined in [ISO/IEC 15693](#) and [NFC Forum Type 5 Tag specification](#).

#### 8.2.4.1 Commands for state transitions

Following commands are implemented for all possible state transitions according to ISO/IEC 15693.

- INVENTORY
- STAY QUIET
- SELECT
- RESET TO READY

On top of these commands, NTAG 5 boost offers

- INVENTORY READ in extended mode (see [Section 8.2.4.5.1](#))
- FAST INVENTORY READ in extended mode (see [Section 8.2.4.5.2](#))
- AUTHENTICATE to move to SELECTED SECURE state (see [Section 8.2.4.4.4](#))

8.2.4.2 Configuration operations

8.2.4.2.1 READ CONFIGURATION

Command code = C0h

The READ CONFIG command returns configuration memory content starting with the first block defined by the Block Address and reads Number of Blocks + 1 configuration blocks.

Access to the configuration blocks depends on the status and definition of the related block within the configuration memory (see [Section 8.1.3](#)).

If one of the requested configuration blocks is not accessible due to the actual status, NTAG 5 boost will respond with Error\_flag set.

A READ CONFIG command can read one or multiple blocks of the following areas of the configuration memory within one command execution:

- Block 00h to block 17h
- Keys can only be read separately if "NOT active" or after a mutual authentication with a key with the Crypto Config privilege set in "active" state
  - Key0: block 20h-23h
  - Key1: block 24h-27h
  - Key2: block 28h-2Bh
  - Key3: block 2Ch-2Fh
- rest of configuration memory

Only Option\_flag = 0b is supported.

Table 123. READ CONFIG request format

Flags	READ CONFIG	Manuf. code	UID	Block Address	Number of Blocks	CRC16
8 bits	8 bits	8 bits	64 bits (optional)	8 bits	8 bits	16 bits

Table 124. READ CONFIG response format when Error\_flag is NOT set

Flags	Data	CRC16
8 bits	(Number of blocks + 1) times 32 bits	16 bits

Table 125. READ CONFIGURATION response format when Error\_flag is set

Flags	Error Code	CRC16
8 bits	8 bits	16 bits

8.2.4.2.2 WRITE CONFIGURATION

Command code = C1h

The WRITE CONFIG command writes the 4 byte data to the requested block address of the configuration memory.

Access to the configuration blocks depends on the status and definition of the related block within the configuration memory (see [Section 8.1.3](#)).

If the requested configuration block is not write accessible due to the actual status, NTAG 5 boost will respond with Error\_flag set.

The timing of the command is write alike.

Option\_flag = 0b and Option\_flag = 1b is supported and is in accordance with ISO/IEC 15693 write-alike commands.

**Table 126. WRITE CONFIG request format**

Flags	WRITE CONFIG	Manuf. code	UID	Block Address	Data	CRC16
8 bits	8 bits	8 bits	64 (optional)	8 bits	32 bits	16 bits

**Table 127. WRITE CONFIG response format when Error\_flag is NOT set**

Flags	CRC16
8 bits	16 bits

**Table 128. WRITE CONFIG response format when Error\_flag is set**

Flags	Error Code	CRC16
8 bits	8 bits	16 bits

### 8.2.4.3 PWD Authentication

NTAG 5 boost can be configured to be used for plain password authentication.

#### 8.2.4.3.1 GET RANDOM NUMBER

##### Command code = B2h

The GET RANDOM NUMBER command is required to receive a 16-bit random number. The passwords that will be transmitted with the SET PASSWORD, ENABLE/DISABLE NFC PRIVACY and DESTROY commands have to be calculated with the password and the random number (see [Section 8.2.4.3.2](#)).

**Table 129. GET RANDOM NUMBER request format**

Flags	GET RANDOM NUMBER	Manuf. code	UID	CRC16
8 bits	8 bits	8 bits	64 bits (optional)	16 bits

**Table 130. GET RANDOM NUMBER response format when Error\_flag is NOT set**

Flags	Random_Number	CRC16
8 bits	16 bits	16 bits

Table 131. GET RANDOM NUMBER response format when Error\_flag is set

Flags	Error Code	CRC16
8 bits	8 bits	16 bits

## 8.2.4.3.2 SET PASSWORD

**Command code = B3h**

The SET PASSWORD command enables the different passwords to be transmitted to the IC to access the different protected functionalities of the following commands. The SET PASSWORD command has to be executed just once for the related password if the IC is powered.

**Remark:** The SET PASSWORD command can only be executed in addressed or selected mode and the timing of the SET PASSWORD command is write alike.

The XOR password has to be calculated with the password and two times the received random number from the last GET RANDOM NUMBER command:

$XOR\_Password[31:0] = Password[31:0] XOR \{Random\_Number[15:0], Random\_Number[15:0]\}$ .

The different passwords are addressed with the password identifier.

Only Option\_flag = 0b is supported.

Table 132. SET PASSWORD request format

Flags	SET PASSWORD	Manuf. code	UID	Password identifier	XOR password	CRC16
8 bits	8 bits	8 bits	64 bits (optional)	8 bits	32 bits	16 bits

Table 133. Password Identifier

Password Identifier	Password
01h	Read
02h	Write
04h	see <a href="#">Section 8.2.4.3.10</a>
08h	Destroy
10h	EAS/AFI
40h	Read from AREA_1
80h	Write to AREA_1

Table 134. SET PASSWORD response format when Error\_flag is NOT set

Flags	CRC16
8 bits	16 bits

Table 135. SET PASSWORD response format when Error\_flag is set

Flags	Error Code	CRC16
8 bits	8 bits	16 bits

**Remark:** If the IC receives an invalid password, it will not execute any following command until a Power-On Reset (POR) (NFC reset) is executed.

#### 8.2.4.3.3 WRITE PASSWORD

##### Command code = B4h

The WRITE PASSWORD command enables a new password to be written into the related memory if the related old password has already been transmitted with a SET PASSWORD command and the addressed password is not locked (see [Section 8.2.4.3.4](#)).

**Remark:** The WRITE PASSWORD command can only be executed in addressed or SELECTED mode. The new password takes effect immediately which means that the new password has to be transmitted with the SET PASSWORD command to access protected blocks/pages.

The different passwords are addressed with the password identifier as defined in [Table 133](#).

The timing of the command is write-alike.

Option\_flag = 0b and Option\_flag = 1b is supported and is in accordance with ISO/IEC 15693 write-alike commands.

Table 136. WRITE PASSWORD request format

Flags	WRITE PASSWORD	Manuf. code	UID	Password identifier	Password	CRC16
8 bits	8 bits	8 bits	64 bits (optional)	8 bits	32 bits	16 bits

Table 137. WRITE PASSWORD response format when Error\_flag is NOT set

Flags	CRC16
8 bits	16 bits

Table 138. WRITE PASSWORD response format when Error\_flag is set

Flags	Error Code	CRC16
8 bits	8 bits	16 bits

#### 8.2.4.3.4 LOCK PASSWORD

##### Command code = B5h

The LOCK PASSWORD command enables the addressed password to be locked if the related password has already been transmitted with a SET PASSWORD command. A locked password cannot be changed.

The different passwords are addressed with the password identifier (see [Table 133](#)).

The timing of the command is write alike.

Option\_flag = 0b and Option\_flag = 1b is supported and is in accordance with ISO/IEC 15693 write-alike commands.

**Table 139. LOCK PASSWORD request format**

Flags	LOCK PASSWORD	Manuf. code	UID	Password identifier	CRC16
8 bits	8 bits	8 bits	64 bits (optional)	8 bits	16 bits

**Table 140. LOCK PASSWORD response format when Error\_flag is NOT set**

Flags	CRC16
8 bits	16 bits

**Table 141. LOCK PASSWORD response format when Error\_flag is set**

Flags	Error Code	CRC16
8 bits	8 bits	16 bits

8.2.4.3.5 64 BIT PASSWORD PROTECTION

**Command code = BBh**

The 64-bit PASSWORD PROTECTION command enables NTAG 5 boost to be instructed that both, Read and Write passwords are required to get access to password protected blocks. This mode can be enabled if the Read and Write passwords have been transmitted first with a SET PASSWORD command.

If the 64-bit password protection is enabled, both passwords are required for read & write access to protected blocks.

Once the 64-bit password protection is enabled, a change back to 32-bit password protection (read and write password) is not possible.

**Remark:** A retransmission of the passwords is not required after the execution of the 64-bit PASSWORD PROTECTION command.

**Remark:** The 64-bit PASSWORD PROTECTION does not include the 16-bit counter block.

The timing of the command is write alike.

Option\_flag = 0b and Option\_flag = 1b is supported and is in accordance with ISO/IEC 15693 write-alike commands.

**Table 142. 64 BIT PASSWORD PROTECTION request format**

Flags	64 BIT PASSWORD PROTECTION	Manuf. code	UID	CRC16
8 bits	8 bits	8 bits	64 bits (optional)	16 bits

**Table 143. 64 BIT PASSWORD PROTECTION response format when Error\_flag is NOT set**

Flags	CRC16
8 bits	16 bits

**Table 144. 64 BIT PASSWORD PROTECTION response format when Error\_flag is NOT set**

Flags	Error Code	CRC16
8 bits	8 bits	16 bits

8.2.4.3.6 PROTECT PAGE

**Command code = B6h**

The PROTECT PAGE command defines the protection pointer address of the user memory to divide the user memory into two arbitrarily sized pages and defines the access conditions for the two pages.

The protection pointer address defines the base address of the higher user memory segment Page 0-H. All block addresses smaller than the protection pointer address are in the user memory segment Page 0-L.

Table below shows an example of the user memory segmentation with the protection pointer address NFC\_PP\_AREA\_0H 14h.

**Remark:** In the example below PP\_AREA\_1 is pointing outside the user memory.

**Table 145. Memory organization**

Block	Byte 0	Byte 1	Byte 2	Byte 3	Description
00h					Page 0-L
01h					
02h					
:	:	:	:	:	
12h					
13h					Page 0-H
14h					
15h					
:	:	:	:	:	
1FFh	C0	C1	00	Protection	Counter

**Remark:** If the protection pointer address is set to block 0, the entire user memory is defined as Page 0-H.

The access conditions and the protection pointer address can be changed under the following circumstances for plain password mode:

- The related passwords (Read and Write password) have been transmitted first with the SET PASSWORD command.
- The page protection condition is not locked (see [Section 8.2.4.3.7](#))

The access conditions and the protection pointer address can be changed under the following circumstances for AES mode:

- The Global Crypto Header is set to "Deactivated" or
- if the Global Crypto Header is not set to "Deactivated" and a valid mutual authentication with a key with read and write privileges has been executed before and the page protection condition is not locked (see [Section 8.2.4.3.7](#)).

The timing of the command is write alike.

Option\_flag = 0b and Option\_flag = 1b is supported and is in accordance with ISO/IEC 15693 write-alike commands.

**Table 146. PROTECT PAGE request format**

Flags	PROTECT PAGE	Manuf. code	UID	Protection pointer address	Extended protection status	CRC16
8 bits	8 bits	8 bits	64 bits (optional)	8 bits	8 bits	16 bits

**Remark:** The IC only accepts protection pointer address values from 00h to FFh. The block containing the 16-bit counter is excluded from the standard user memory protection scheme.

**Table 147. Extended Protection status byte**

Bit	Name	Value	Description
7	RFU	0b	
6	RFU	0b	
5	WH	0b	Page 0-H is not write protected
		1b	Page 0-H is write protected
4	RH	0b	Page 0-H is not read protected
		1b	Page 0-H is read protected
3	RFU	0b	
2	RFU	0b	
1	WL	0b	Page 0-L is not write protected
		1b	Page 0-L is write protected
0	RL	0b	Page 0-L is not read protected
		1b	Page 0-L is read protected

**Table 148. Protection status bits definition in plain password mode**

WH/WL	RH/RL	32-bit Protection	64-bit Protection
0b	0b	Public	Public
0b	1b	Read and Write protected by the Read password	Read and Write protected by the Read plus Write password
1b	0b	Write protected by the Write password	Write protected by the Read plus Write password
1b	1b	Read protected by the Read password and Write protected by the Read and Write password	Read and Write protected by the Read plus Write password



**Table 149. Protection status bits definition in AES mode**

WH/WL	RH/RL	Protection
0b	0b	Public
0b	1b	Read and Write protected: Mutual authentication with a key with read privilege is required
1b	0b	Write protected: Mutual authentication with a key with write privilege is required
1b	1b	Read and Write protected: Mutual authentication with a key with read and write privileges is required

**Table 150. PROTECT PAGE response format when Error\_flag is NOT set**

Flags	CRC16
8 bits	16 bits

**Table 151. PROTECT PAGE response format when Error\_flag is set**

Flags	Error Code	CRC16
8 bits	8 bits	16 bits

The information about the stored settings of the protection pointer address and access conditions can be read with the GET NXP SYSTEM INFORMATION command (see [Section 8.2.4.9.14](#)).

**8.2.4.3.7 LOCK PAGE PROTECTION CONDITION**

**Command code = B7h**

The LOCK PAGE PROTECTION CONDITON command locks the protection pointer address and the status of the page protection conditions.

The LOCK PAGE PROTECTION CONDITON command can be successfully executed under the following circumstances:

- The Global Crypto Header is set to "Deactivated".
- If the Global Crypto Header is not set to "Deactivated" and a valid mutual authentication with a key with read and write privileges has been executed before.

The timing of the command is write alike.

Option\_flag = 0b and Option\_flag = 1b is supported and is in accordance with ISO/IEC 15693 write-alike commands.

**Table 152. LOCK PAGE PROTECTION CONDITION request format**

Flags	LOCK PAGE PROTECTION CONDITION	Manuf. code	UID	Protection pointer address	CRC16
8 bits	8 bits	8 bits	64 bits (optional)	8 bits	16 bits

**Table 153. LOCK PAGE PROTECTION CONDITION response format when Error\_flag is NOT set**

Flags	CRC16
8 bits	16 bits

**Table 154. LOCK PAGE PROTECTION CONDITION response format when Error\_flag is set**

Flags	Error Code	CRC16
8 bits	8 bits	16 bits

**Remark:** If the transmitted protection pointer address does not match with the stored address the IC will respond according to the error handling.

#### 8.2.4.3.8 DESTROY

##### Command code = B9h

In plain password mode the DESTROY command disables NTAG 5 boost if the destroy password is correct. This command is irreversible and NTAG 5 boost will never respond to any command neither NFC nor I<sup>2</sup>C again.

The DESTROY command can only be executed in addressed or SELECTED mode.

The XOR password has to be calculated with the password and two times the received random number from the last GET RANDOM NUMBER command:

$XOR\_Password[31:0] = Password[31:0] XOR \{Random\_Number[15:0], Random\_Number[15:0]\}$ .

The timing of the command is write alike.

Option\_flag = 0b and Option\_flag = 1b is supported and is in accordance with ISO/IEC 15693 write-alike commands.

NOTE: In AES mode, Mutual Authentication with Purpose\_MAM2 = 1011b needs to be executed to destroy NTAG 5 boost.

**Table 155. DESTROY request format**

Flags	DESTROY	Manuf. code	UID	XOR password	CRC16
8 bits	8 bits	8 bits	64 bits (optional)	16 bits	16 bits

**Table 156. DESTROY response format when Error\_flag is NOT set**

Flags	CRC16
8 bits	16 bits

**Table 157. DESTROY response format when Error\_flag is set**

Flags	Error Code	CRC16
8 bits	8 bits	16 bits

8.2.4.3.9 ENABLE NFC PRIVACY

**Command code = BAh**

The ENABLE NFC PRIVACY command in plain password mode enables NFC PRIVACY mode (see [Section 8.7](#)) for NTAG 5 boost if the Privacy password is correct.

The XOR password has to be calculated with the password and two times the received random number from the last GET RANDOM NUMBER command:

$$\text{XOR\_Password}[31:0] = \text{Password}[31:0] \text{ XOR } \{ \text{Random\_Number}[15:0], \text{Random\_Number}[15:0] \}.$$

To get out of the NFC PRIVACY mode, the valid Privacy password has to be transmitted to the IC with the DISABLE NFC PRIVACY command.

The timing of the command is write alike.

Option\_flag = 0b and Option\_flag = 1b is supported and is in accordance with ISO/IEC 15693 write-alike commands.

NOTE: In AES mode, Mutual Authentication with Purpose\_MAM2 = 1001b needs to be executed to enable the NFC PRIVACY mode for NTAG 5 boost.

**Table 158. ENABLE NFC PRIVACY request format**

Flags	SET PASSWORD	IC Mfg code	UID	XOR password	CRC16
8 bits	8 bits	8 bits	64 bits optional	32 bits	16 bits

**Table 159. ENABLE NFC PRIVACY response format when Error\_flag is NOT set**

Flags	CRC16
8 bits	16 bits

**Table 160. ENABLE NFC PRIVACY response format when Error\_flag is set**

Flags	Error Code	CRC16
8 bits	8 bits	16 bits

8.2.4.3.10 DISABLE NFC PRIVACY

**Command code = B3h**

The DISABLE NFC PRIVACY command moves the NTAG 5 boost out of the NFC PRIVACY mode.

**Remark:** The timing of the DISABLE PRIVACY command is write alike.

The XOR password has to be calculated with the password and two times the received random number from the last GET RANDOM NUMBER command:

$$\text{XOR\_Password}[31:0] = \text{Password}[31:0] \text{ XOR } \{ \text{Random\_Number}[15:0], \text{Random\_Number}[15:0] \}.$$

The Privacy identifier is 04h.

Option\_flag = 1b and Option\_flag = 0b are supported.

**Table 161. DISABLE NFC PRIVACY request format**

Flags	SET PASSWORD	Manuf. code	UID	Privacy identifier	XOR password	CRC16
8 bits	8 bits	8 bits	64 bits (optional)	8 bits	32 bits	16 bits

**Table 162. DISABLE NFC PRIVACY response format when Error\_flag is NOT set**

Flags	CRC16
8 bits	16 bits

**Table 163. DISABLE NFC PRIVACY response format when Error\_flag is set**

Flags	Error Code	CRC16
8 bits	8 bits	16 bits

**Remark:** If the IC receives an invalid password, it will not execute any following command until a Power-On Reset (POR) (NFC reset) is executed.

#### 8.2.4.4 AES Authentication

##### 8.2.4.4.1 Introduction

NXP implements a scalable security approach. This means, during lifetime the security level may be changed. In production, NXP configures the IC, that plain password mode is enabled. With the DEV\_SEC\_CONFIG byte (see [Table 65](#)) AES mutual authentication may be activated. Only in case NFC Security is set to AES mode following commands and AES mutual authentication is enabled. To lock the security level, the Security Lock bits need to be set to 010b (see [Table 65](#)).

##### 8.2.4.4.2 PROTECT PAGE

See [Section 8.2.4.3.6](#)

##### 8.2.4.4.3 LOCK PAGE PROTECTION CONDITION

See [Section 8.2.4.3.7](#)

##### 8.2.4.4.4 AUTHENTICATE

As defined in [ISO/IEC 15693](#) and [ISO/IEC 29167-10](#).

Command code = 35h

CSI code= 00h (AES Crypto Suite)

The AUTHENTICATE command allows the interrogator to perform the following authentication procedures as defined in [ISO/IEC 15693](#):

- Tag Authentication (TAM1)
- Mutual Authentication (MAM1, MAM2)

After receiving a valid AUTHENTICATE command, NTAG 5 boost calculates the response and as soon as the calculation is finalized, the response with the result of the crypto calculation (b3 flag is set) is sent. Only for tag authentication the calculation result is additionally stored in the response buffer (b2 flag is set).

NTAG 5 boost supports the Crypto Suite AES128 as defined in ISO/IEC 29167-10.

Option\_flag = 0b and Option\_flag = 1b is supported and is in accordance with ISO/IEC 15693 write-alike commands.

**Table 164. AUTHENTICATE request format**

Flags	AUTHENTICATE	UID	CSI	Message	CRC16
8 bits	8 bits	64 bits (optional)	8 bits	Depending on TAM1 (96 bit), MAM1 (96 bit) or MAM2 (136 bit)	16 bits

[Table 165](#) defines the response of NTAG 5 boost to an AUTHENTICATE command.

For more detailed information, refer to ISO/IEC 29167-10.

**Table 165. AUTHENTICATE response format when Error\_flag is NOT set(in process reply)**

Flags	Barker Code	TResponse	CRC16
8 bits (b2 and b3 is set)	8 bits (Done flag is set)	Depending on TAM1 (128 bit), MAM1(176 bit) or MAM2 (0 bit)	16 bits

**Table 166. AUTHENTICATE response format when Error\_flag is set**

Flags	Error Code	CRC16
8 bits	8 bits	16 bits

**Tag Authentication (TAM1)**

[Table 167](#) defines the message within the AUTHENTICATE command for tag authentication (TAM1).

For more detailed information, refer to ISO/IEC 29167-10.

**Table 167. Message format for TAM1**

	AuthMethod	CustomData	TAM1_RFU	KeyID	IChecksum_TAM1
# of bits	2	1	5	8	80
Description	00b	0b	00000b	[7:0]	random interrogator challenge

[Table 168](#) defines the response of NTAG 5 boost to an AUTHENTICATE command.

For more detailed information, refer to ISO/IEC 29167-10.

**Table 168. TResponse for TAM1**

TResponse TAM1 (128 bit)
AES-ECB-ENC(Key[KeyID].ENC_key,C_TAM1[15:0]    TRnd_TAM1[31:0]    IChallenge_TAM1[79:0])

**Mutual Authentication (MAM1, MAM2)**

The mutual authentication is a two-pass authentication procedure. The first AUTHENTICATE command (MAM1) is executing the tag authentication and gets the challenge from NTAG 5 boost. The second AUTHENTICATE command (MAM2) is executing the interrogator authentication.

[Table 169](#) defines the message within the AUTHENTICATE command for mutual authentication (MAM1).

For more detailed information, refer to ISO/IEC 29167-10.

**Table 169. Message format for MAM1**

	AuthMethode	Step	MAM1_RFU	KeyID	IChallenge_MAM1
# of bits	2	2	4	8	80
Description	10b	00b	0000b	[7:0]	random interrogator challenge

[Table 170](#) defines the response of NTAG 5 boost to an AUTHENTICATE command for MAM1.

For more detailed information, refer to ISO/IEC 29167-10.

**Table 170. TResponse for MAM1**

TResponse MAM1 (176 bit)
AES-ECB-ENC(Key[KeyID].ENC_key,C_TAM1[15:0]    TChallenge_MAM1[31:0]    IChallenge_TAM1[79:0])    TChallenge_MAM1[79:32]

[Table 171](#) defines the message within the AUTHENTICATE command for mutual authentication (MAM2).

For more detailed information, refer to ISO/IEC 29167-10.

**Table 171. Message format for MAM2**

	AuthMethode	Step	MAM2_RFU	IResponse
# of bits	2	2	4	128
Description	10b	01b	0000b	AES-DEC(, Key[KeyID].ENC_key,C_MAM2[11:0]    Purpose_MAM2[3:0]    IChallenge_MAM1[31:0]    TChallenge_MAM1[79:0])

[Table 172](#) defines valid values for Purpose\_MAM2 for NTAG 5 boost.

**Table 172. Definition of Purpose\_MAM2**

	Purpose_MAM2	Description
Standard	0000b	Mutual Authentication
	all other 0xxxb	RFU
NXP specific	1000b	Disable NFC Privacy Mode until NFC field reset
	1001b	Enable NFC Privacy Mode
	1010b	Disable NFC Privacy Mode
	1011b	Destroy NTAG 5 boost
	all other 1xxxb	RFU

[Table 173](#) defines the response of NTAG 5 boost to an AUTHENTICATE command for MAM2.

For more detailed information, refer to ISO/IEC 29167-10.

**Table 173. TResponse for MAM2**

TResponse MAM2 (0 bit)
Empty message (no data)

8.2.4.4.5 CHALLENGE

As defined in [ISO/IEC 15693](#) and ISO/IEC 29167-10.

Command code = 39h

CSI code= 00h (AES Crypto Suite)

The CHALLENGE command transmits the message (challenge) to NTAG 5 boost to authenticate as defined in [ISO/IEC 15693](#).

The CHALLENGE command can only be executed in the READY state and in not addressed mode.

After receiving a valid CHALLENGE command, NTAG 5 boost starts with the crypto calculation.

If the calculation is finalized, NTAG 5 boost will respond to a valid READBUFFER command with the result of the crypto calculation based on the previous CHALLENGE command message.

NTAG 5 boost supports the Crypto Suite AES128 as defined in ISO/IEC 29167-10.

Only Option\_flag = 0b is supported.

**Table 174. CHALLENGE request format**

Flags	CHALLENGE	UID	CSI	Message	CRC16
8 bits	8 bits	64 bits (optional)	8 bits	96 bits	16 bits

[Table 175](#) defines the message within the CHALLENGE command.

For more detailed information, refer to ISO/IEC 29167-10.

**Table 175. Message format**

	AuthMethode	CustomData	TAM1_RFU	KeyID	IChallenge_TAM1
# of bits	2	1	5	8	80
Description	00b	0b	00000b	[7:0]	random interrogator challenge

No response is sent on a CHALLENGE command.

For more detailed information, refer to ISO/IEC 29167-10.

8.2.4.4.6 READBUFFER

As defined in [ISO/IEC 15693](#) and ISO/IEC 29167-10.

Command code = 3Ah

The READBUFFER command allows the interrogator to request the crypto calculation result based on a valid previous CHALLENGE command from NTAG 5 boost.

NTAG 5 boost supports the Crypto Suite AES128 as defined in ISO/IEC 29167-10.

Only Option\_flag = 0b is supported.

**Table 176. READBUFFER request format**

Flags	READBUFFER	UID	CRC16
8 bits	8 bits	64 bits (optional)	16 bits

For more detailed information, refer to ISO/IEC 29167-10.

[Table 177](#) and [Table 178](#) defines the response of NTAG 5 boost to a READBUFFER command.

For more detailed information, refer to ISO/IEC 29167-10.

**Table 177. READBUFFER response format when Error\_flag is NOT set**

Flags	TResponse	CRC16
8 bits	128 bits (see Table 122)	16 bits

**Table 178. TResponse**

TResponse
AES-ECB-ENC(Key[KeyID].ENC_key,C_TAM1[15:0]    TRnd_TAM1[31:0]    IChallenge_TAM1[79:0])

**Table 179. READBUFFER response format when Error\_flag is set**

Flags	Error Code	CRC16
8 bits	8 bits	16 bits

**8.2.4.5 Memory operations**

Following commands are implemented for accessing user memory according to ISO/IEC 15693.

- READ SINGLE BLOCK
- WRITE SINGLE BLOCK
- LOCK BLOCK
- READ MULTIPLE BLOCKS up to 3Fh blocks
- EXTENDED READ SINGLE BLOCK
- EXTENDED WRITE SINGLE BLOCK
- EXTENDED LOCK BLOCK
- EXTENDED READ MULTIPLE BLOCKS up to 3Fh blocks

On top of these commands, NTAG 5 boost offers INVENTORY READ and FAST INVENTORY READ

**8.2.4.5.1 INVENTORY READ**

**Command code = A0h**



When receiving the INVENTORY READ request, NTAG 5 boost performs the same as the anti-collision sequence, with the difference that instead of the UID and the DSFID, the requested response is defined by additional options.

The INVENTORY READ command provides two modes which are defined by the most significant bit of the mask length byte as follows:

- Standard mode (most significant bit of mask length byte equal 0b) (see [Section 8.2.4.5.1.1](#))
- Extended mode (most significant bit of mask length byte equal 1b)  
The extended mode offers additional features to optimize the inventory procedure for different requirements (see [Section 8.2.4.5.1.2](#))

The INVENTORY READ command may also be transmitted in addressed or SELECTED mode. Then the command behaves similar to a READ or READ MULTIPLE BLOCK (see [Section 8.2.4.5.1.3](#)).

8.2.4.5.1.1 Standard mode

If most significant bit of mask length byte is equal 0b the INVENTORY READ command is used in the standard mode.

If the Inventory\_flag is set to 1b and an error is detected, NTAG 5 boost remains silent.

If the Option flag is set to 0b, n blocks of data are transmitted. If the Option flag is set to 1b, n blocks of data and the part of the UID which is not part of the mask are transmitted.

The request contains:

- Flags
- INVENTORY READ command code
- IC manufacturer code
- AFI (if AFI\_flag is set to 1b)
- Mask length (most significant bit equal 0b)
- Mask value (if mask length > 00h)
- First block number to be read
- Number of blocks to be read
- CRC 16

Table 180. INVENTORY READ request format

Flags	INVENTORY READ	Manuf. code	AFI	Mask length	Mask value	First block number	Number of blocks	CRC16
8 bits	8 bits	8 bits	8 bits (optional)	8 bits	0 to 8 bytes	8 bits	8 bits	16 bits

If the Inventory\_flag is set to 1b, only NTAG 5 boost in the READY or SELECTED (SECURE) state will respond (same behavior as in the INVENTORY command). The meaning of Flags bits 7 to 4 is as defined in [ISO/IEC 15693](#).

The INVENTORY READ command can also be transmitted in the addressed or SELECTED mode (see [Section 8.2.4.5.1.3](#)).

The number of blocks in the request is one less than the number of blocks that NTAG 5 boost returns in its response.

If the Option\_flag in the request is set to logic 0b the response contains:

**Table 181. INVENTORY READ response format: Option flag logic 0b**

Flags	Data	CRC16
8 bits	Number of blocks times 32 bits	16 bits

NTAG 5 boost reads the requested block(s) and sends back their value in the response. The mechanism and timing of the INVENTORY READ command performs the same as the INVENTORY command which is defined in [ISO/IEC 15693](#).

If the Option\_flag in the request is set to logic 1b, the response contains:

**Table 182. INVENTORY READ response format: Option flag logic 1b**

Flags	Rest of UID which is not part of the mask and slot number	Data	CRC16
8 bits	0 to 64 bit, always a multiple of 8 bits	Number of blocks times 32 bits	16 bits

NTAG 5 boost reads the requested block(s) and sends back their value in the response. Additionally the bytes of the UID, which are not parts of the mask and the slot number in case of 16 slots, are returned. Instead of padding with zeros up to the next byte boundary, the corresponding bits of the UID are returned. The mechanism and timing of the INVENTORY READ command perform the same as the INVENTORY command which is defined in [ISO/IEC 15693](#).

**Remark:** The number of bits of the retransmitted UID can be calculated as follows:

- 16 slots: 60 bits (bit 64 to bit 4) - mask length rounded up to the next byte boundary
- 1 slot: 64 bits - mask length rounded up to the next byte boundary

**Remark:** If the sum of first block number and number of blocks exceeds the total available number of user blocks, the number of transmitted blocks is less than the requested number of blocks. This means that the last returned block is the highest available user block, followed by the 16-bit CRC and the EOF.

Example: mask length = 30 bits

Returned: bit 64 to bit 4 (30 bits) = 30 gives 4 bytes

**Table 183. Example: mask length = 30**

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	UID
mask value including padding with zeros				-				transmitted by interrogator
				returned value				transmitted by NTAG 5 boost

8.2.4.5.1.2 Extended Mode

If the most significant bit of the Mask Length byte is equal 1b the response format is defined by the extended option byte.

The request contains:

- Flags
- Inventory Read command code
- IC Manufacturer code
- AFI (if the AFI flag is set to 1b)

- Mask length (most significant bit equal 1b)
- Extended Options
- Mask value (if mask length > 0)
- First Block Number to be read, if specified in extended options byte
- Number of Blocks to be read, if specified in extended options byte
- CRC 16

**Table 184. Inventory Read (extended mode) request format**

Flags	INVENTOR READ	Manuf. code	AFI	Mask Length	ext.Options	Mask Value	First block number	Number of blocks	CRC 16
8 bits	8 bits	8 bits	8 bits (optional)	8 bits	8 bits	0 to 64 bits	8 bits (optional)	8 bits (optional)	16 bits

If the Inventory\_flag is set to 1b, only NTAG 5 boost in the READY or SELECTED (SECURE) state will respond (same behavior as in the INVENTORY command). The meaning of flags 5 to 8 is in accordance with table 5 in [ISO/IEC 15693](#).

The INVENTORY READ command can also be transmitted in the addressed or SELECTED mode (see [Section 8.2.4.5.1.3](#)).

**Table 185. Extended options**

Bit	Name	Value	Feature
7	RFU	0	
6	RFU	0	
5	QUIET	0	remain in current state
		1	go to QUIET state after response
4	SKIP_DATA	0	NTAG 5 boost will add the user memory blocks in the response as requested with first block number byte and number of blocks byte in the command
		1	No user memory data is requested, first block number byte and number of blocks byte shall not be transmitted in the command
3	CID_RESPONSE	0	Custom ID (CID) will be NOT transmitted in the response
		1	Custom ID (CID) will be transmitted in the response
2	CID_COMPARE	0	No CID is transmitted in the command
		1	16-bit CID will be transmitted in the command and only NTAG 5 boost with the same CID will respond
1	UID_MODE	0	UID will be transmitted as in regular mode (truncated reply depending on least significant 7 bits value of mask length and the mask value)
		1	Complete UID will be transmitted (independent from mask length)
0	EAS_MODE	0	NTAG 5 boost responds independent from the EAS status
		1	Respond only, when EAS is enabled

If the Option\_flag in the request is set to 1b the response contains the truncated or complete UID depending on the extended option UID\_MODE bit.

If the Option\_flag in the request is set to 0b the UID is not part of the response.

Table 186. Inventory Read (extended mode) response format: Option\_flag 1b

Flags	Optional truncated UID OR complete UID	Optional data	CRC16
8 bits	0 to 64 bits	Block length	16 bits
	Multiple of 8 bits	Repeated as needed	

The mechanism and timing of the INVENTORY READ command performs the same as at the INVENTORY command which is defined in [ISO/IEC 15693](#).

If the UID is requested in the truncated format the retransmitted UID can be calculated as follows:

16 slots: 64 - 4 - mask length rounded up to the next byte boundary

1 slot: 64 - mask length rounded up to the next byte boundary

Example: mask length = 30 Returned: 64 - 4 - 30 = 30 gives 4 bytes

Table 187. Example

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	UID
mask value incl. padding with zeros								transmitted by Interrogator
				returned value				transmitted by NTAG 5 boost

8.2.4.5.1.3 Addressed and SELECTED mode

The INVENTORY READ command can also be transmitted in the addressed or SELECTED mode. In this case, the Inventory\_flag is set to 0 and the meaning of flags 7 to 4 is in accordance with ISO/IEC 15693.

In the addressed or selected mode, the INVENTORY READ command behaves similar to a READ or READ MULTIPLE BLOCK command.

In the addressed mode, it is recommended to address the IC with a mask length of 64 and to transmit the complete UID in the mask value field.

In the selected mode (IC has been selected with a valid SELECT command before), it is recommended to address the IC with a mask length of 0 (and do not transmit the mask value field).

**Remark:** If the INVENTORY READ command is used in the addressed or selected mode, the AFI shall not be transmitted and the IC will only respond in the first-time slot.

8.2.4.5.2 FAST INVENTORY READ

**Command code = A1h**

When receiving the FAST INVENTORY READ command, NTAG 5 boost behaves the same as the INVENTORY READ command with the following exceptions:

The data rate in the direction NTAG 5 boost to the reader is twice as defined in [ISO/IEC 15693](#) depending on the Datarate\_flag 53 kbit (high data rate) or 13 kbit (low data rate).

The data rate from the reader to NTAG 5 boost and the time between the rising edge of the EOF from the reader to NTAG 5 boost remains unchanged (stays the same as defined in [ISO/IEC 15693](#)).

Only the single subcarrier mode is supported for the response to the FAST INVENTORY READ command.

**8.2.4.6 SRAM operations**

When SRAM is mirrored to user EEPROM address space, standard READ BLOCK and WRITE BLOCK commands can be used. To have a more efficient way to access the 256 bytes SRAM, READ SRAM and WRITE SRAM are implemented

**8.2.4.6.1 READ SRAM**

**Command code = D2h**

This command can only be used, when NTAG 5 boost is powered via V<sub>CC</sub> end SRAM\_ENABLE bit (see [Table 37](#)) is set to 1b.

When receiving READ SRAM desired SRAM blocks will be returned.

NTAG 5 boost returns only the requested blocks. The blocks are numbered from 00h to 3Fh. The number of blocks in the request is one less than the number of blocks that NTAG 5 boost returns in its response. EXAMPLE: A value of 06h in the "Number of Blocks" field requests to read 7 blocks. A value of 00h requests to read a single block from SRAM.

If SRAM is read or write protected a valid authentication needs to be proceeded.

It is recommended to use this command in pass-through mode.

Only Option\_flag = 0b is supported.

**Table 188. READ SRAM request format**

Flags	READ SRAM	Manuf. code	UID	Block Address	Number of Blocks	CRC16
8 bits	8 bits	8 bits	64 bits (optional)	8 bits	8 bits	16 bits

**Table 189. READ SRAM response format when Error\_flag is NOT set**

Flags	Block Security Status (optional) + Data	CRC16
8 bits	(Number of Blocks+1) x 32 bits Data	16 bits

Block Security Status and Data bytes repeat as a duple.

**Table 190. READ SRAM response format when Error\_flag is set**

Flags	Error Code	CRC16
8 bits	8 bits	16 bits

**8.2.4.6.2 Write SRAM**

**Command code = D3h**

This command can only be used, when NTAG 5 boost is powered via V<sub>CC</sub> end SRAM\_ENABLE bit (see [Table 37](#)) is set to 1b.

When receiving WRITE SRAM desired SRAM blocks will be written to the SRAM. It is recommended to use this command in pass-through mode because of performance reasons.

If SRAM is write protected a valid authentication needs to be preceded.

The blocks are numbered from 00h to 3Fh. The number of blocks in the request is one less than the number of blocks that the VICC shall write. E.g., to write one block to SRAM Number of Blocks is coded as 00h.

Option\_flag = 0b and Option\_flag = 1b is supported and is in accordance with ISO/IEC 15693 write-alike commands.

**Table 191. WRITE SRAM request format**

Flags	WRITE SRAM	IC Mfg code	UID	Block Address	Number of blocks	Data	CRC16
8 bits	8 bits	8 bits	64 bits (optional)	8 bits	8 bits	(Number of blocks + 1) times 32 bits	16 bits

**Table 192. WRITE SRAM response format when Error\_flag is NOT set**

Flags	CRC16
8 bits	16 bits

**Table 193. WRITE SRAM response format when Error\_flag is set**

Flags	Error Code	CRC16
8 bits	8 bits	16 bits

### 8.2.4.7 Originality Signature

#### 8.2.4.7.1 READ SIGNATURE

##### Command code = BDh

The READ SIGNATURE command returns an IC-specific, 32 byte ECC signature. How to change and / or lock the originality signature is described in [Section 8.8](#).

Only Option\_flag = 0b is supported.

**Table 194. READ SIGNATURE request format**

Flags	READ SIGNATURE	Manuf. code	UID	CRC16
8 bits	8 bits	8 bits	64 bits (optional)	16 bits

**Table 195. READ SIGNATURE response format when Error\_flag is NOT set**

Flags	Originality Signature	CRC16
8 bits	256 bits	16 bits

**Table 196. READ SIGNATURE response format when Error\_flag is set**

Flags	Error Code	CRC16
8 bits	8 bits	16 bits

Details on how to validate the signature is provided in [AN11350](#).

**8.2.4.8 I<sup>2</sup>C Transparent Channel**

NTAG 5 boost offers an NFC to I<sup>2</sup>C bridge. With this mode, different I<sup>2</sup>C slaves (e.g., sensors) can be connected without a microcontroller. There shall be no other active I<sup>2</sup>C master on the same bus. The needed power for the sensors may be provided with NTAG 5 boost energy harvesting capability.

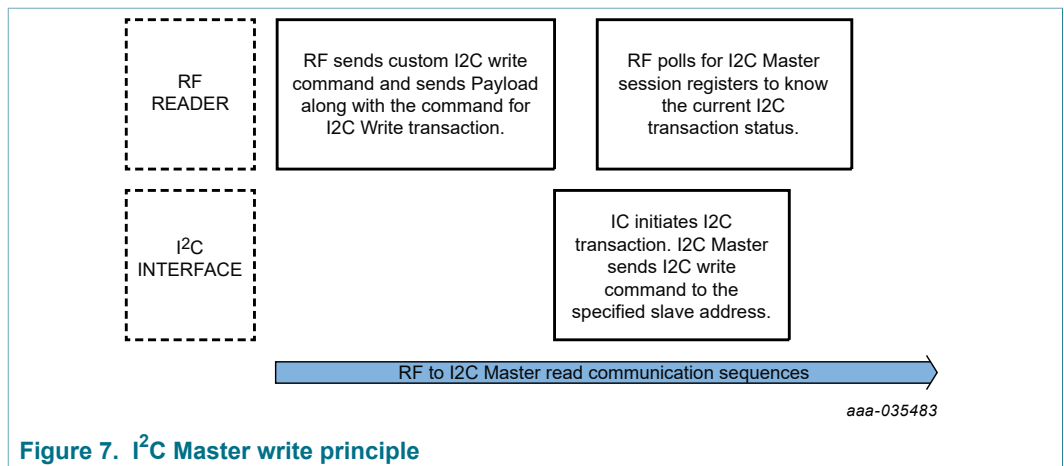
I<sup>2</sup>C master communication can do maximum 256 bytes to read from, and write to the connected slave.

I<sup>2</sup>C master clock speed needs to be configured with I2C\_MASTER\_SCL\_LOW and I2C\_MASTER\_SCL\_HIGH (see [Table 61](#)). I<sup>2</sup>C slave address will be selected directly within WRITE I<sup>2</sup>C and READ I<sup>2</sup>C command.

SRAM needs to be enabled by setting SRAM\_ENABLE bit (see [Table 37](#)) to 1b.

Basic principle for triggering an I<sup>2</sup>C write to the connected slave is illustrated in the figure below. NFC reader will get the response immediately, and then polls for the status of the I<sup>2</sup>C transaction (see [Table 112](#)).

Details of the NFC command WRITE I<sup>2</sup>C can be found in [Section 8.2.4.8.1](#).



Basic principle for triggering an I<sup>2</sup>C read and getting the response of the connected slave is illustrated in the figure below. Again the NFC reader will get a response immediately, and then polls for the status of the I<sup>2</sup>C transaction (see [Table 112](#)). Finally the result can be read from SRAM.

Details of the NFC command READ I<sup>2</sup>C can be found in [Section 8.2.4.8.2](#)

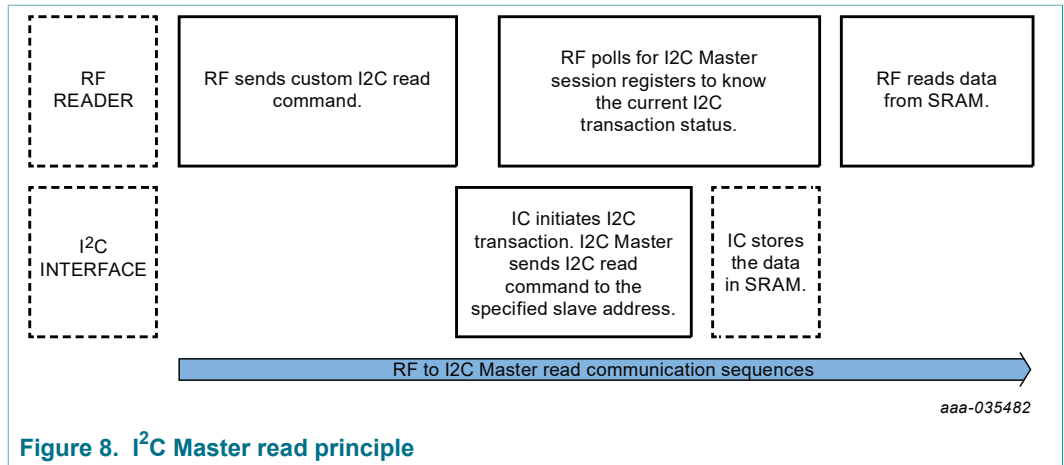


Figure 8. I<sup>2</sup>C Master read principle

8.2.4.8.1 WRITE I<sup>2</sup>C

Command code = D4h

WRITE I<sup>2</sup>C command is used to trigger an I<sup>2</sup>C master write command (R/W bit is 0b) on the I<sup>2</sup>C bus.

Command parameters:

- I<sup>2</sup>C param contains I<sup>2</sup>C slave address and STOP condition option.
- Data length N byte codes the length of bytes to be sent to the slave. N+1 bytes need to be put in the Data field and will be sent to the slave. E.g., to send one byte, 00h needs to be coded.
- Data field contains the data to be sent to the I<sup>2</sup>C slave. Minimum number of bytes is 1 byte, maximum is 256 bytes.

If SRAM is read or write protected a valid authentication needs to be preceded.

Response to this command will follow immediately.

To check the status and result of the WRITE I<sup>2</sup>C command, I<sup>2</sup>C Master Status Registers ADh should be checked (see Table 112).

Only Option\_flag = 0b is supported.

Table 197. WRITE I<sup>2</sup>C request format

Flags	WRITE I <sup>2</sup> C	Manuf. code	UID	I <sup>2</sup> C param	Data length N	Data	CRC16
8 bits	8 bits	8 bits	64 bits (optional)	8 bits	8 bits	(N+1) x 8 bits	16 bits

Table 198. I<sup>2</sup>C param byte

I <sup>2</sup> C param bit	Status	Value	Description
7	Disable STOP condition	0b	STOP condition will be generated at the end of transaction
		1b	STOP condition will be omitted at the end of transaction
6 to 0	I <sup>2</sup> C Address	xxh	7-bit I <sup>2</sup> C slave address of connected slave



**Table 199. WRITE I<sup>2</sup>C response format when Error\_flag is NOT set**

Flags	CRC16
8 bits	16 bits

**Table 200. WRITE I<sup>2</sup>C response format when Error\_flag is set**

Flags	Error Code	CRC16
8 bits	8 bits	16 bits

### 8.2.4.8.2 READ I<sup>2</sup>C

Command code = D5h

READ I<sup>2</sup>C command is used to trigger an I<sup>2</sup>C master read command (R/W bit is 1b) on the I<sup>2</sup>C bus.

Command parameters:

- I<sup>2</sup>C param contains I<sup>2</sup>C address and STOP condition option
- Data length N byte codes the length of bytes to be read from the slave. N+1 bytes will be read from the slave. E.g., to read one byte 00h needs to be coded. Maximum is FFh, which means 256 bytes will be read.

Response to this command will follow immediately.

The status register [Table 112](#) indicates when the I<sup>2</sup>C read command is completed.

To get the response of the addressed I<sup>2</sup>C slave device, the READ SRAM (see [Section 8.2.4.6.1](#)) command is used.

Only Option\_flag = 0b is supported.

**Table 201. READ I<sup>2</sup>C request format**

Flags	READ I <sup>2</sup> C	IC Mfg code	UID	I <sup>2</sup> C param	Data length	CRC16
8 bits	8 bits	8 bits	64 bits optional	8 bits	8 bits	16 bits

**Table 202. I<sup>2</sup>C param byte**

Bit	Status	Value	Description
7	Disable STOP condition	0b	STOP condition will be generated at the end of transaction
		1b	STOP condition will be omitted at the end of transaction
6 to 0	I <sup>2</sup> C Address	xxh	7-bit I <sup>2</sup> C slave address of connected slave

**Table 203. READ I<sup>2</sup>C response format when Error\_flag is NOT set**

Flags	CRC16
8 bits	16 bits

Table 204. READ I<sup>2</sup>C response format when Error\_flag is set

Flags	Error Code	CRC16
8 bits	8 bits	16 bits

### 8.2.4.9 Other

#### 8.2.4.9.1 WRITE AFI

As defined in [ISO/IEC 15693](#).

#### 8.2.4.9.2 LOCK AFI

As defined in [ISO/IEC 15693](#).

#### 8.2.4.9.3 WRITE DSFID

As defined in [ISO/IEC 15693](#).

#### 8.2.4.9.4 LOCK DSFID

As defined in [ISO/IEC 15693](#).

#### 8.2.4.9.5 SET EAS

Command code = A2h

The SET EAS command enables the EAS mode if the EAS mode is not locked.

If the EAS mode is password protected the EAS password has to be first transmitted with the SET PASSWORD command.

If AES authentication scheme is enabled and EAS mode is protected, a valid mutual authentication with a key with the EAS/AFI privilege set has to be executed before.

The timing of the command is write alike.

Option\_flag = 0b and Option\_flag = 1b is supported and is in accordance with ISO/IEC 15693 write-alike commands.

Table 205. SET EAS request format

Flags	SET EAS	Manuf. code	UID	CRC16
8 bits	8 bits	8 bits	64 bits (optional)	16 bits

Table 206. SET EAS response format when Error\_flag is NOT set

Flags	CRC16
8 bits	16 bits

Table 207. SET EAS response format when Error\_flag is set

Flags	Error Code	CRC16
8 bits	8 bits	16 bits

## 8.2.4.9.6 RESET EAS

Command code = A3h

The RESET EAS command disables the EAS mode if the EAS mode is not locked.

If the EAS mode is password protected the EAS password has to be first transmitted with the SET PASSWORD command.

If AES authentication scheme is enabled and EAS mode is protected a valid mutual authentication with a key with the EAS/AFI privilege set has to be executed before.

The timing of the command is write alike.

Option\_flag = 0b and Option\_flag = 1b is supported and is in accordance with ISO/IEC 15693 write-alike commands.

**Table 208. RESET EAS request format**

Flags	RESET EAS	Manuf. code	UID	CRC16
8 bits	8 bits	8 bits	64 bits optional	16 bits

**Table 209. RESET EAS response format when Error\_flag is NOT set**

Flags	CRC16
8 bits	16 bits

**Table 210. RESET EAS response format when Error\_flag is set**

Flags	Error Code	CRC16
8 bits	8 bits	16 bits

## 8.2.4.9.7 LOCK EAS

Command code = A4h

The LOCK EAS command locks the current state of the EAS mode and the EAS ID.

If the EAS mode is password protected the EAS password has to be first transmitted with the SET PASSWORD command.

If AES authentication scheme is enabled and EAS mode is protected a valid mutual authentication with a key with the EAS/AFI privilege set has to be executed before.

The timing of the command is write alike.

Option\_flag = 0b and Option\_flag = 1b is supported and is in accordance with ISO/IEC 15693 write-alike commands.

**Table 211. LOCK EAS request format**

Flags	LOCK EAS	Manuf. code	UID	CRC16
8 bits	8 bits	8 bits	64 bits (optional)	16 bits

**Table 212. LOCK EAS response format when Error\_flag is NOT set**

Flags	CRC16
8 bits	16 bits

**Table 213. LOCK EAS response format when Error\_flag is set**

Flags	Error Code	CRC16
8 bits	8 bits	16 bits

8.2.4.9.8 EAS ALARM

Command code = A5h

The EAS ALARM command can be used in the following configurations:

- Option\_flag is set to 0b:  
EAS ID mask length and EAS ID value shall not be transmitted.  
If the EAS mode is enabled, the EAS response is returned from the IC.
- Option\_flag is set to 1b:  
Within the command, the EAS ID mask length has to be transmitted to identify how many bits of the following EAS ID value are valid (multiple of 8-bits). Only those ICs will respond with the EAS sequence which have stored the corresponding data in the EAS ID configuration (selective EAS) and if the EAS Mode is set.  
If the EAS ID mask length is set to 00h, the IC will answer with its EAS ID.

**Table 214. EAS ALARM Request format**

Flags	EAS ALARM	Manuf. code	UID	EAS ID mask length	EAS ID value	CRC16
8 bits	8 bits	8 bits	64 bits (optional)	8 bits (optional)	0, 8 or 16 bits (optional)	16 bits

If an error is detected the IC remains silent.

Option\_flag is set to 0b or Option\_flag is set to logic 1b and the EAS ID mask length is not equal to 00h:

**Table 215. EAS ALARM Response format (Option flag logic 0)**

Flags	EAS sequence	CRC16
8 bits	256 bits	16 bits

EAS sequence (starting with the least significant bit, which is transmitted first; read from left to right):

```
11110100 11001101 01000110 00001110 10101011 11100101 00001001 11111110
00010111 10001101 00000001 00011100 01001011 10000001 10010010 01101110
01000001 01011011 01011001 01100001 11110110 11110101 11010001 00001101
10001111 00111001 10001011 01001000 10100101 01001110 11101100 11110111
```

Option\_flag is set to 1b and the EAS ID mask length is equal to 00h:

Table 216. EAS ALARM Response format(Option flag logic 1)

Flags	EAS ID value	CRC16
8 bits	16 bits	16 bits

Table 217. EAS ALAMR response format when Error\_flag is set

Flags	Error Code	CRC16
8 bits	8 bits	16 bits

If the EAS mode is disabled, the IC remains silent.

**Remark:** NTAG 5 boost in the QUIET state will not respond to an EAS ALARM command except the addressed flag is set.

#### 8.2.4.9.9 PROTECT EAS/AFI

##### Command code = A6h

In plain password mode the PROTECT EAS/AFI command enables the password protection for EAS and/or AFI if the EAS/AFI password is first transmitted with the SET PASSWORD command.

In AES mode, the PROTECT EAS/AFI command enables the protection for EAS and/or AFI if a valid mutual authentication with the EAS/AFI privilege has been executed before.

Option\_flag set to 0b: EAS will be protected.

Option\_flag set to 1b: AFI will be protected.

Both protections (AFI and EAS) can be enabled separately.

Once the EAS/AFI protection is enabled, it is not possible to change back to unprotected EAS and/or AFI.

The timing of the command is write-alike as of write commands with Option\_flag set to 0b.

**Note:** Option\_flag is only related to the parameter to be locked, and NOT to the response behavior.

Table 218. PROTECT EAS/AFI request format

Flags	PROTECT EAS/AFI	Manuf. code	UID	CRC16
8 bits	8 bits	8 bits	64 bits (optional)	16 bits

Table 219. PROTECT EAS/AFI response format when Error\_flag is NOT set

Flags	CRC16
8 bits	16 bits

Table 220. PROTECT EAS/AFI response format when Error\_flag is set

Flags	Error Code	CRC16
8 bits	8 bits	16 bits

## 8.2.4.9.10 WRITE EAS ID

**Command code = A7h**

The command WRITE EAS ID enables a new EAS Identifier to be stored in the corresponding configuration memory.

If EAS is password protected (for Set and Reset EAS) the EAS password has to be first transmitted with the SET PASSWORD command.

If AES mode is enabled and the EAS is protected a valid mutual authentication with a key with the EAS/AFI privilege set has to be executed before.

The timing of the command is write alike.

Option\_flag = 0b and Option\_flag = 1b is supported and is in accordance with ISO/IEC 15693 write-alike commands.

Table 221. WRITE EAS ID request format

Flags	WRITE EAS ID	Manuf. code	UID	EAS ID value	CRC16
8 bits	8 bits	8 bits	64 bits (optional)	16 bits	16 bits

Table 222. WRITE EAS ID response format when Error\_flag is NOT set

Flags	CRC16
8 bits	16 bits

Table 223. WRITE EAS ID response format when Error\_flag is set

Flags	Error Code	CRC16
8 bits	8 bits	16 bits

## 8.2.4.9.11 GET MULTIPLE BLOCK SECURITY STATUS

As defined in [ISO/IEC 15693](#).

## 8.2.4.9.12 GET SYSTEM INFORMATION

As defined in [ISO/IEC 15693](#).

The TAG type of NTAG 5 boost is "01h".

## 8.2.4.9.13 EXTENDED GET SYSTEM INFORMATION

As defined in [ISO/IEC 15693](#) and ISO/IEC 29167-10.

Command code = 3Bh

## 8.2.4.9.14 GET NXP SYSTEM INFORMATION

**Command code = ABh**

The GET NXP SYSTEM INFORMATION command provides information about the IC access conditions and supported features.

Table 224. GET NXP SYSTEM INFORMATION request format

Flags	Get NXP System Info	Manuf. code	UID	CRC16
8 bits	8 bits	8 bits	64 bits (optional)	16 bits

Table 225. GET NXP SYSTEM INFORMATION response format when Error\_flag is NOT set

Flags	PP pointer	PP condition	Lock bits	Feature flag	CRC16
8 bits	8 bits	8 bits	8 bits	32 bits	16 bits

On a valid received command the IC responds with detailed information:

PP pointer byte contains the block address of the protection pointer.

PP condition byte contains information about the access condition to Page H and Page L.

Table 226. Protection Pointer condition byte

Bit	Name	Value	Description
7	RFU	0b	
6	RFU	0b	
5	WH	0b	Page 0-H is not write protected
		1b	Page 0-H is write protected
4	RH	0b	Page 0-H is not read protected
		1b	Page 0-H is read protected
3	RFU	0b	
2	RFU	0b	
1	WL	0b	Page 0-L is not write protected
		1b	Page 0-L is write protected
0	RL	0b	Page 0-L is not read protected
		1b	Page 0-L is read protected

Lock bits byte contains information about permanently locked features.

Table 227. Lock bits byte

Bit	Name	Value	Description
7 to 4	RFU	0b	
3	NFC_PP_AREA_0H and NFC_PPC	0b	NFC_PP_AREA_0H and NFC_PPC is NOT locked
		1b	NFC_PP_AREA_0H and NFC_PPC is locked
2	DSFID	0b	DSFID is NOT locked
		1b	DSFID is locked
1	EAS	0b	EAS is NOT locked
		1b	EAS is locked
0	AFI	0b	AFI is NOT locked
		1b	AFI is locked

Feature flag byte contains information about supported features (related bit is 1b) of NTAG 5 boost. With this response, it is possible to distinguish the different NTAG 5 family members.

Table 228. Feature flags byte 0

Bit	Name	Description	NTAG 5
7	CID	Customer ID supported (see <a href="#">Section 8.1.3.3</a> )	1b
6	EAS IR	EAS selection supported by extended mode in INVENTORY READ command (see <a href="#">Section 8.2.4.5.1</a> )	1b
5	INVENTORY READ EXT	Extended mode supported by INVENTORY READ command (see <a href="#">Section 8.2.4.5.1</a> )	1b
4	AFI PROT	AFI protection supported (see <a href="#">Section 8.2.4.9.9</a> )	1b
3	EAS PROT	EAS protection supported (see <a href="#">Section 8.2.4.9.9</a> )	1b
2	EAS ID	EAS ID supported by EAS ALARM command (see <a href="#">Section 8.2.4.9.10</a> )	1b
1	COUNTER	NFC Counter supported (see <a href="#">Section 8.1.2.1</a> )	1b
0	UM PROT	User memory protection supported (see <a href="#">Section 8.2.4.3.6</a> )	1b

Table 229. Feature flags byte 1

Bit	Name	Description	NTAG 5	
			NTP5210	NTP5312 NTP5332 NTA5332
7	HIGH BITRATES	high bitrates supported (see <a href="#">Section 8.2</a> )	0b	1b
6	WRITE CID	Write and Lock CID enabled (see <a href="#">Section 8.1.3.3</a> )	1b	
5	DESTROY	DESTROY feature supported (see <a href="#">Section 8.2.4.3.8</a> )	1b	
4	NFC PRIVACY	NFC Privacy mode supported (see <a href="#">Section 8.2.4.3.9</a> )	1b	
3	RFU		0b	
2	PERS QUIET	PERSISTENT QUIET feature supported	0b	
1	RFU		0b	
0	ORIG SIG	Originality signature supported (see <a href="#">Section 8.1.3.1</a> )	1b	

Table 230. Feature flags byte 2

Bit	Name	Description	NTAG 5	
			NTP5210 NTP5312	NTP5332 NTA5332
7 to 3	RFU		all 0b	
2	KEY PRIV	Key privileges supported (see <a href="#">Section 8.1.3.8</a> )	0b	1b
1	MUTUAL AUTH	Mutual Authentication feature supported (see <a href="#">Section 8.6.4</a> )	0b	1b



Bit	Name	Description	NTAG 5	
			NTP5210 NTP5312	NPT5332 NTA5332
0	TAG AUTH	Tag Authentication feature supported (see <a href="#">Section 8.6.4</a> )	0b	1b

Table 231. Feature flags byte 3

Bit	Name	Description	NTAG 5		
			NTP5210	NTP5312	NPT5332 NTA5332
7	EXT FLAG	Additional 32 bits feature flags are transmitted	0b		
6-5	Interface	00b only NFC interface available	01b	11b	
		01b GPIO/ED host interface			
		10b RFU			
		11b GPIO and I <sup>2</sup> C host interface			
4	RFU		0b		
3 to 0	NUM KEYS	Number of Keys	0h		4h

Table 232. GET NXP SYSTEM INFORMATION response format when Error\_flag is set

Flags	Error Code	CRC16
8 bits	8 bits	16 bits

8.2.4.9.15 PICK RANDOM ID

**Command code = C2h**

In AES mode the PICK RANDOM ID command instructs NTAG 5 boost in NFC PRIVACY Mode to generate a random ID. After a valid PICK RANDOM ID command, the IC will respond with that random ID on following INVENTORY commands or GET SYSTEM INFORMATION command until an RF reset to allow an anti-collision procedure. The random ID will include the CID to identify the group-password or group-key to disable the privacy mode.

Only Option\_flag = 0b is supported.

Table 233. PICK RANDOM ID request format

Flags	Pick Random ID	Manuf. code	UID	CRC16
8 bits	8 bits	8 bits	64 bits (optional)	16 bits

Table 234. PICK RANDOM ID response format when Error\_flag is NOT set

Flags	CRC16
8 bits	16 bits

**Table 235. PICK\_RANDOM\_ID response format when Error\_flag is set**

Flags	Error Code	CRC16
8 bits	8 bits	16 bits

After a successful PICK\_RANDOM\_ID the NTAG 5 boost will respond on an INVENTORY command with a random ID as defined in the table below.

**Table 236. Random ID**

MSB						LSB
63:56	55:48	47:40	39:32	31:24	23:16	15:0
E0h	04h	00h	00h	CID_1	CID_0	16-bit random ID

### 8.2.5 Data integrity

Following mechanisms are implemented in the contactless communication link between reader and NTAG 5 boost to ensure very reliable data transmission:

- 16-bit CRC per block
- Bit count checking
- Bit coding to distinguish between logic 1, logic 0, and no information
- Channel monitoring (protocol sequence and bit stream analysis)

### 8.2.6 Error Handling

#### 8.2.6.1 Transmission Errors

According to ISO/IEC 15693 NTAG 5 boost will not respond if a transmission error (CRC, bit coding, bit count, wrong framing) is detected and will silently wait for the next correct received command.

#### 8.2.6.2 Not supported commands or options

If the received command or option is not supported, the behavior depends on the addressing mechanism.

- Non-Addressed Mode  
NTAG 5 boost remains silent
- Addressed or selected Mode  
NTAG 5 boost responds with error code 0Fh (no information given, or error code not supported).  
If the Inventory flag or the Protocol Extension flag is set, the IC will not respond if the command or option is not supported.
- Parameter out of range
  - Read alike commands  
If the sum of the first block number and the number of blocks exceeds the total available number of user blocks, the number of transmitted blocks is less than the requested number of blocks. This means that the last returned block is the highest available user block, followed by the 16-bit CRC and the EOF.
  - Write alike commands  
If the address of a block to be written does not exist or a block to be written is locked, the behavior of the IC depends on the addressing mechanism.

- Non-Addressed Mode  
NTAG 5 boost remains silent.
- Addressed or SELECTED Mode  
NTAG 5 boost responds with error code 0Fh (no information given, or error code not supported).

**8.3 Wired Interface**

NTAG 5 boost has not only an NFC interface, but also a wired interface. Details are described in following clauses.

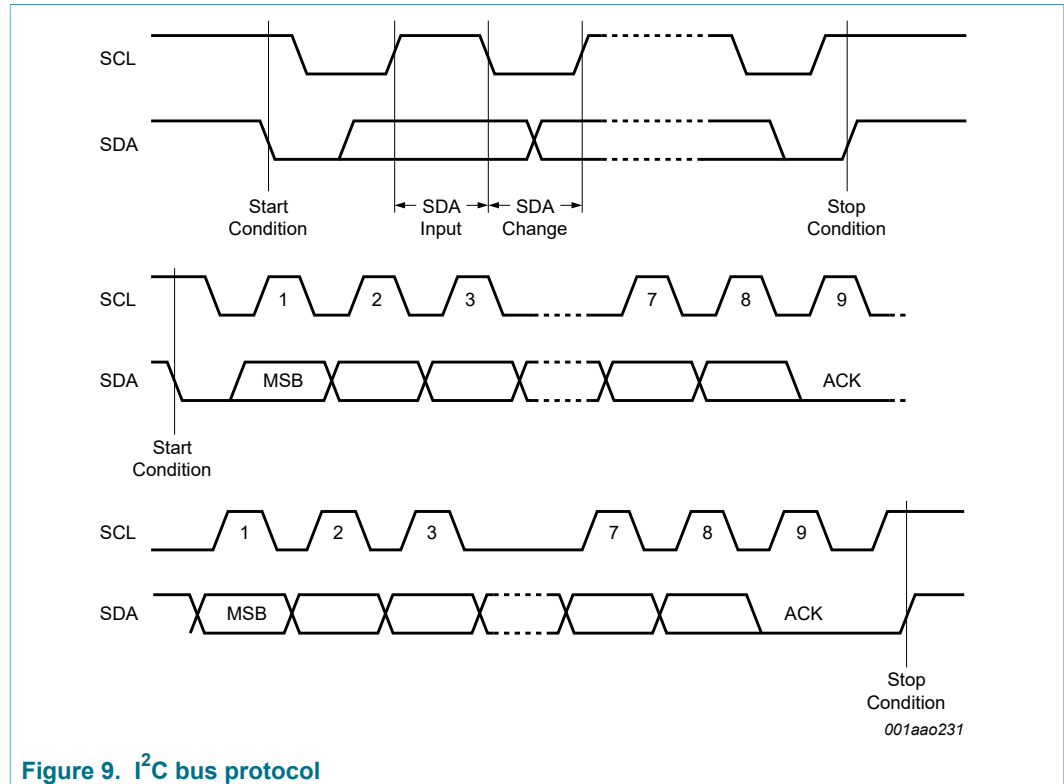
**8.3.1 I<sup>2</sup>C interface**

The definition of the I<sup>2</sup>C interface is according to the [UM10204](#). The details of slave and master mode are described in [Section 8.3.1.1](#) and [Section 8.3.1.2](#).

**8.3.1.1 Slave mode**

For details about I<sup>2</sup>C interface, refer to [UM10204](#).

The I<sup>2</sup>C slave interface supports both standard (up to 100 kHz) and fast mode (up to 400 kHz) communication speeds for both read and write. Implementation will be a so-called asynchronous interface which uses the SCL clock for the I<sup>2</sup>C protocol handling after which the data is synchronized to the system clock for memory access. NTAG 5 boost can be used in multi-master/multi-slave applications.



**Figure 9. I<sup>2</sup>C bus protocol**

NTAG 5 boost supports the I<sup>2</sup>C protocol defined in [UM10204](#). Any device that sends data onto the bus is defined as a transmitter, and any device that reads the data from the bus

is defined as a receiver. The device that controls the data transfer is known as the "bus master", and the other as the "slave" device. A data transfer can only be initiated by the bus master, which will also provide the serial clock for synchronization.

8.3.1.1.1 Start condition

Start is identified by a falling edge of Serial Data (SDA), while Serial Clock (SCL) is stable in the high state. A Start condition must precede any data transfer command. NTAG 5 boost continuously monitors SDA (except during a Write cycle) and SCL for a Start condition, and will not respond unless one is given.

8.3.1.1.2 Stop condition

Stop is identified by a rising edge of SDA while SCL is stable and driven high. A Stop condition terminates communication between NTAG 5 boost and the bus master. A Stop condition at the end of a Write command triggers the internal write cycle.

8.3.1.1.3 Acknowledge bit (ACK)

The acknowledge bit is used to indicate a successful byte transfer. The bus transmitter, whether it is the bus master or slave device, releases Serial Data (SDA) after sending 8 bits of data. During the ninth clock pulse period, the receiver pulls Serial Data (SDA) low to acknowledge the receipt of the 9th data bits.

8.3.1.1.4 Data input

During data input, the IC samples SDA on the rising edge of SCL. For correct device operation, SDA must be stable during the rising edge of SCL, and the SDA signal must change only when SCL is driven low.

8.3.1.1.5 Addressing

To start communication between a bus master and NTAG 5 boost, the bus master must initiate a Start condition. Following this initiation, the bus master sends the device address. The IC address from I<sup>2</sup>C consists of a 7-bit device identifier (see [Table 237](#) for default value).

As long as I<sup>2</sup>C address is 7 bit long, the 8th bit (least significant bit) is used as the Read/Write bit (R/W). This bit is set to 1b for Read and 0b for Write operations.

If a match occurs on the device address, the IC gives an acknowledgment on SDA during the 9th bit time. If the IC does not match the device select code, it deselects itself from the bus and clears the register I2C\_IF\_LOCKED (see [Table 94](#)).

**Table 237. Default NTAG 5 I<sup>2</sup>C address from I<sup>2</sup>C**

	Device address						
	b6	b5	b4	b3	b2	b1	b0
Value	1 <sup>[1]</sup>	0 <sup>[1]</sup>	1 <sup>[1]</sup>	0 <sup>[1]</sup>	1 <sup>[1]</sup>	0 <sup>[1]</sup>	0 <sup>[1]</sup>

[1] Initial values - can be changed.

The I<sup>2</sup>C address of NTAG 5 boost (Configuration Byte) can be modified by the NFC and I<sup>2</sup>C interface.

#### 8.3.1.1.6 Disable I<sup>2</sup>C Interface

NTAG 5 boost offers the option to disable the I<sup>2</sup>C interface temporarily using the session register bit DISABLE\_I<sup>2</sup>C (see [Table 106](#)). With this feature, the NFC Device can easily get exclusive access to EEPROM.

This feature can be enabled via the NFC interface during the session by setting related session bit.

#### 8.3.1.2 Master mode

NTAG 5 boost can be configured in I<sup>2</sup>C master mode. Using I<sup>2</sup>C Master interface, I<sup>2</sup>C slave device like sensors or memories can be connected to NTAG 5 boost without an external microcontroller. Using energy harvesting capability, I<sup>2</sup>C device can be powered by NTAG 5 boost .

When using I<sup>2</sup>C master mode, it must be ensured, that there is no other active I<sup>2</sup>C master on the same bus. The USE\_CASE\_CONF needs to be set to I<sup>2</sup>C master (01b) and SRAM needs to be enabled in CONFIG\_1 byte (see [Table 37](#)).

The used clock speed can be configured by setting I2C\_MASTER\_SCL\_LOW and I2C\_MASTER\_SCL\_HIGH (see [Section 8.1.3.21](#)).

To communicate with the connected I<sup>2</sup>C slave device two custom commands are implemented.

- WRITE I2C (see [Section 8.2.4.8.1](#))
- READ I2C (see [Section 8.2.4.8.2](#))

The response from the READ I<sup>2</sup>C command will be stored in the SRAM and can be read (see [Section 8.2.4.6.1](#)) afterwards from NFC perspective. Due to the 256 byte SRAM, only 256 bytes can be written / read at once to / from the I<sup>2</sup>C interface.

Of course, all other NFC commands are working in master mode and the user memory as well as configuration memory is accessible from NFC perspective.

WARNING: When enabling I<sup>2</sup>C master mode and disabling NFC interface in parallel, NTAG 5 boost gets disabled for current session.

Implementation details can be found in [AN12368](#).

#### 8.3.1.3 Watch Dog Timer

A programmable watchdog timer is implemented to unlock the I<sup>2</sup>C host from NTAG 5 boost latest after a defined maximum time period. The host itself will not be notified of this event directly but the NFC status register is updated accordingly.

On default Watch Dog Timer is enabled with a value of 0848h (~20 ms) but the watchdog timer can be freely set with WDT\_CONFIG from 0000h (9.434 μs) up to (FFFFh+1) \* 9.434 μs (~618 ms). It is recommended to keep this time as short as possible, by setting the value above, but close to the maximum needed I<sup>2</sup>C transaction time.

The timer is only active, when WDT\_ENABLE is set to 1b and the IC is V<sub>CC</sub> powered. The timer starts ticking when the I<sup>2</sup>C communication starts. The Watch Dog Timer ensures, that I<sup>2</sup>C interface gets released after the configured time period in any case.

In the case where the I<sup>2</sup>C communication has completed before the end of the timer and the status register I2C\_IF\_LOCKED was not cleared by the host, it will be cleared when defined watchdog time elapses.

NOTE: If WDT\_CONFIG configured time elapses before ongoing I<sup>2</sup>C communication is finished, WDT will release SDA line in between of ongoing I<sup>2</sup>C communication.

The timer is reset automatically, when I2C\_IF\_LOCKED gets cleared, or the IC is not V<sub>CC</sub> powered.

In I<sup>2</sup>C master use case, watchdog timer is always enabled independently of WDT\_ENABLE. It is important to set WDT\_CONFIG in accordance with maximum execution time.

8.3.1.4 Command Set

NTAG 5 boost offers an easy to use I<sup>2</sup>C command set.

- WRITE MEMORY and READ MEMORY to access user and configuration memory
- WRITE MEMORY to present the related password, when password authentication from I<sup>2</sup>C perspective is enabled
- WRITE REGISTER and READ REGISTER to access session registers

In Figure 10 the access to EEPROM with READ MEMORY and WRITE MEMORY is illustrated and following symbols are used:

- START: I<sup>2</sup>C Start condition as defined in Section 8.3.1.1.1.
- SL\_AD: 7-bit slave address (msb aligned) plus (lsb) R/W bit as defined in Section 8.3.1.1.5
- BL\_AD1 (MSB) / BL\_AD0 (LSB): 16-bit block address
- A/N: Acknowledge / NAK as defined in Section 8.3.1.1.3
- DATA 0, DATA 1, ... , DATA N: Data bytes to be read or written.  
N shall be 3 for writing to EEPROM  
N shall be multiple of 4 reduced by 1; maximum 255 for writing to SRAM  
N is any number for reading data. NTAG 5 boost will respond until host's NACK.
- Stop: Stop condition as defined in Section 8.3.1.1.2

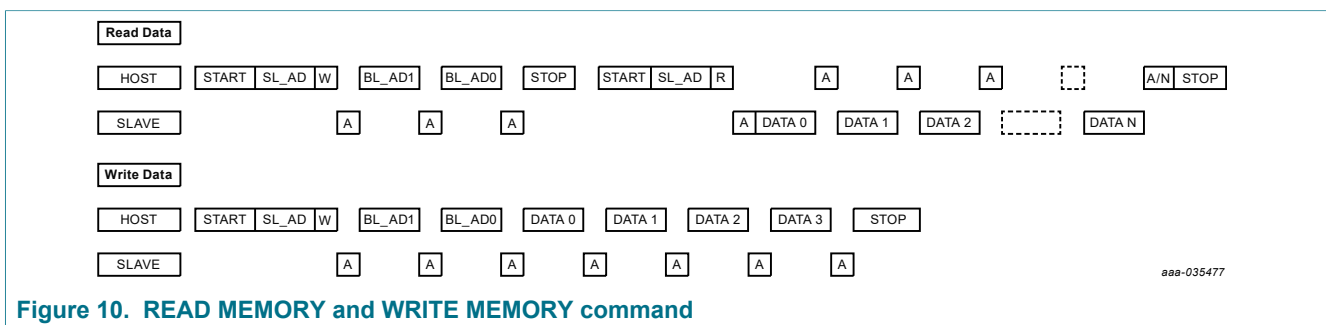
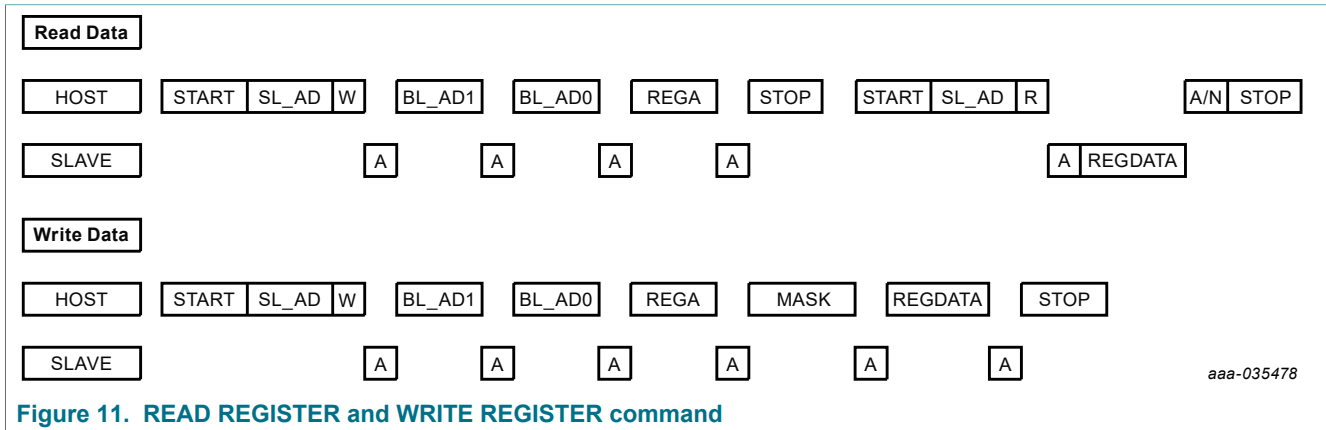


Figure 10. READ MEMORY and WRITE MEMORY command

In Figure 11 the access to Registers with READ REGISTER and WRITE REGISTER is illustrated and following symbols are used:

- START: I<sup>2</sup>C Start conditions as defined in Section 8.3.1.1.1.
- SL\_AD: 7-bit slave address (msb aligned) plus (lsb) R/W bit as defined in Section 8.3.1.1.5
- BL\_AD1 (MSB) | BL\_AD0 (LSB): 16-bit register block address
- REGA: 8-bit register address
- MASK: 8-bit control register bit mask. Only if corresponding control bit is set to 1b, the register bit will be overwritten.
- A/N: Acknowledge / Not Acknowledge as defined in Section 8.3.1.1.3

- REGDAT: 8-bit register data to read/write
- STOP: Stop condition as defined in [Section 8.3.1.1.2](#)



### 8.3.1.5 Error Handling

In case of any detected error, NTAG 5 boost in slave mode responds with a NACK:

- Memory Write
  - EEPROM
    - generated on the fourth data byte if the block is not writable
  - SRAM
    - generated on the first byte if the block is not writable
  - Arbiter locked to NFC interface, or
  - EEP cycle ongoing, or
  - I<sup>2</sup>C interface disabled
    - generate on block address BL\_AD0
- Memory Read
  - EEPROM/SRAM
    - returned data will be FFh if the access is to restricted region
  - Arbiter locked to NFC interface, or
  - EEP cycle ongoing, or
  - I<sup>2</sup>C interface disabled
    - generate on block address BL\_AD0
- Register Access
 

Registers are always accessible. NACK will only be generated:

  - DATA NACK due to register write command to trigger system reset
  - NACK for register read/write command on BL\_AD0 if I<sup>2</sup>C interface is disabled

### 8.3.2 Event detection

The event detection feature provides the capability to trigger an external device (e.g.,  $\mu$ Controller) or switch on the connected circuitry by an external power management unit depending on activities on the NFC interface. On top this active low pin can be used as one of the two possible PWM channels to offer I<sup>2</sup>C and PWM functionality.

As the event detection pin functionality is operated via NFC field power, V<sub>CC</sub> supply for the IC itself is only required when ED pin is used as PWM channel.

NOTE: In some cases V<sub>OUT</sub> pin might be used as field detection trigger.

The configurable events indicated at event detection pin are:

- The presence/absence of the NFC field
- Data read/written in pass-through mode
- Arbiter locked/unlocked EEPROM to NFC interface
- NDEF Message TLV length field is ZERO/non-ZERO
- IC is/is not in standby mode
- Dedicated config bit is ZERO
- Write/Read command ongoing

Event detection pin is an active LOW signal. Due to open-drain implementation an external pull-up resistor shall be used on this pin.

How to use the event detection pin in applications is described in [AN11203](#).

### 8.3.3 GPIO

I<sup>2</sup>C pins (SCL/SDA) are multiplexed and can be used as general-purpose input/output pins linked to configuration/session bits. When configured as GPIO pins, I<sup>2</sup>C communication is not possible anymore.

At POR, the GPIO are set to high-impedance state. When configuration is read, the pins are controlled to behave as per the configuration.

GPIOs can be configured to be either input or output (see [Section 8.1.3.15](#)). In input mode, the status of the pad will be available in one of the session register bits. In output mode status depends on the session register/config bits content.

How to use the GPIO pins in applications is described in [AN11203](#).

### 8.3.4 PWM

I<sup>2</sup>C pins (SDA/SCL) and ED pin are multiplexed and can be used as a pulse width modulation output. I<sup>2</sup>C pins have push-pull architecture, ED pin is an open-drain implementation, which means the PWM signal gets inverted.

PWM resolution, pre-scaler factor (see [Section 8.1.3.15](#)) as well as duty cycle can be configured using configuration bytes (see [Section 8.1.3.16](#)).

The pulse width modulation resolution (PWMx\_RESOLUTION\_CONF) defines the maximum number of pulses that are available in the given PWM period. PWM resolution can be set independently for both outputs to either 6, 8, 10 or 12 bits.

The 2-bit PWMx\_PRESCALE value divides the PWM input frequency (1695 kHz) by a factor of 1, 2, 4 or 8.

**Table 238. Pulse Width Modulation Frequency**

Resolution	Pre-scaler			
	00b	01b	10b	11b
12 bit	413 Hz	206 Hz	103 Hz	52 Hz
10 bit	1.7 kHz	825.0 Hz	412.6 Hz	206.2 Hz
8 bit	6.6 kHz	3.3 kHz	1.7 kHz	825.0 Hz
6 bit	26.4 kHz	13.2 kHz	6.6 kHz	3.3 kHz



PWMx\_ON and PWMx\_OFF defines the starting point and end point of the PWMx output is asserted to HIGH.

To calculate proper PWMx\_ON (start of HIGH level) and PWMx\_OFF (end of HIGH level) values, PWMx\_RESOLUTION\_CONF value and PWM\_PRESCALE values need to be set to achieve desired PWM frequency. As an example 12-bit resolution is chosen. Duty cycle shall be set to 20 % and start time shall be 10 % offset.

Start Time 10 %:  $2^{12} * 10/100 = 4096 * 10/100 = \sim 410 \rightarrow \text{PWMx\_ON} = 19\text{Ah}$

PWM Duty Cycle 20 %:  $2^{12} * 20/100 = 4096 * 20/100 = \sim 819 \rightarrow \text{PWMx\_OFF} = 410 + 819 = 1229 = 4\text{CDh}$

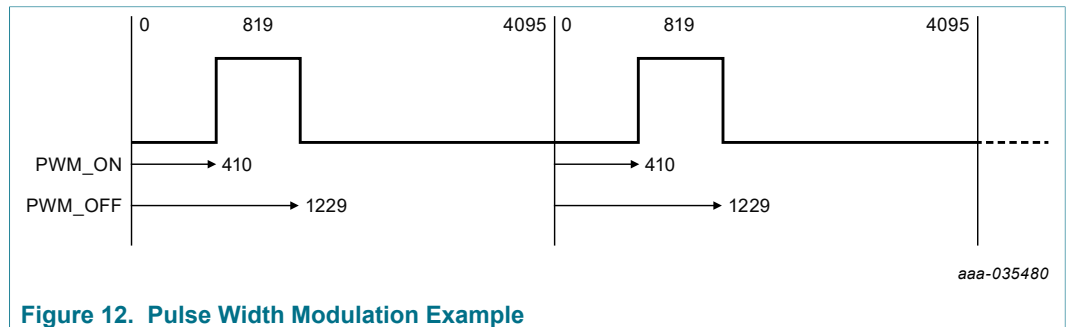


Figure 12. Pulse Width Modulation Example

How to use PWM in applications is described in [AN11203](#).

### 8.3.5 Standby mode

To minimize overall current consumption, when the IC is supplied via V<sub>CC</sub> NTAG 5 boost can be set to standby mode by writing related session bit from NFC or I<sup>2</sup>C perspective. The IC will leave standby mode according to configuration when NFC field is detected, automatically, or HPD pin gets pulled to HIGH for at least 20 μs and released again. In standby mode the current is typically less than 10 μA.

Worst case standby current consumption values can be found in [Section 10.1](#) table.

In case SDA/SCL/GPIO/PWM pins are not used the pins can be left floating. However, to ensure lowest standby current, following settings are needed:

CONFIG bytes USE\_CASE\_CONF shall be set in any case to GPIO/PWM, both pins SDA\_GPIO1 and SCL\_GPIO0 shall be set as input using weak pull-up (GPIOx\_IN in).

Block 37h: CONFIG\_1, USE\_CASE\_CONF shall be set to GPIO/PWM (10b) and CONFIG, GPIOx\_IN, both shall be set to plain input with weak pullup (01b).

Block 39h: PWM\_GPIO\_CONFIG\_0, SDA\_GPIO1 and SCL\_GPIO0 shall both be set to 1b to define them as general-purpose input.

### 8.3.6 Hard power-down mode

In hard power-down mode NTAG 5 boost is switched off using hard power down pin. When pulled to HIGH, the hard power down current is typically less than 0.25 μA. This mode can only be left by connecting HPD pin to ground.

Worst case hard power down current consumption values can be found in [Section 10.1](#) table.

## 8.4 Arbitration between NFC and I<sup>2</sup>C interface

There are different modes implement to ensure access to the EEPROM and described in detail hereafter. Two status bits (I2C\_IF\_LOCKED and NFC\_IF\_LOCKED) are provided to show the status of arbiter.

Details about the different arbitrations modes can be found in [AN12364](#).

### 8.4.1 NFC Mode

If NTAG 5 boost is only powered by NFC, arbiter needs only to lock to the NFC interface if the IC receives a valid NFC command. After completion of the NFC command, NFC\_IF\_LOCKED will be cleared automatically.

### 8.4.2 I<sup>2</sup>C Mode

If NTAG 5 boost is only powered by V<sub>CC</sub>, arbiter needs only to lock to the I<sup>2</sup>C interface if the IC is correctly addressed for the memory access. The host needs to clear I2C\_IF\_LOCKED. Otherwise, the bit will be cleared automatically if the watchdog timer expires.

In I<sup>2</sup>C mode, availability of the SRAM as part of the memory depends on SRAM\_ENABLE bit.

### 8.4.3 Normal Mode

If NTAG 5 boost is powered by NFC and VCC, arbiter locks interface on a first come first serve principle.

When receiving a valid NFC command and access is not locked to I<sup>2</sup>C, then the arbiter locks to the NFC interface. After completion of the NFC command, the lock will be released automatically. The host can access the registers at any time. Only access to EEPROM is locked.

When NTAG 5 boost is correctly addressed by its I<sup>2</sup>C address for the memory access and access is not locked to NFC, then the arbiter locks to the I<sup>2</sup>C interface. The host needs to clear the lock actively. If not, the lock will be released automatically as soon as the watchdog timer expires. NFC reader can access the registers at any time. Only access to EEPROM is locked.

In this mode, availability of the SRAM depends on SRAM\_ENABLE bit.

How to exchange data based on NDEF messages is defined in [NFC Forum Tag NDEF Exchange Protocol \(TNEP\) Specification](#).

### 8.4.4 SRAM Mirror Mode

In this mode arbiter works like in normal mode with the exception, that SRAM is used instead of EEPROM.

### 8.4.5 SRAM Pass-Through Mode

In this mode, the NTAG 5 boost transfers data from NFC to I<sup>2</sup>C and vice versa using the SRAM. The arbiter switches automatically between the two interfaces when accessing the terminator block (last block of SRAM).

Details can be found in [AN12364](#).

#### 8.4.6 SRAM PHDC Mode

This mode is similar to SRAM mirror mode. This mode needs to be enabled, when PHDC communication scheme as defined in [NFC Forum PHDC specification](#) shall be used. NFC will always get a response when accessing SRAM.

### 8.5 Energy harvesting

NTAG 5 boost in passive mode provides the capability to supply external low-power devices with energy harvested from the NFC field of an NFC device.

When DISABLE\_POWER\_CHECK bit is set to 0b, minimum provided output power can be configured by setting desired voltage and minimum required output current in the related configuration bytes (see [Section 8.1.3.18](#)).

WARNING: Sufficient RF field is required when DISABLE\_POWER\_CHECK is set to 0b to have access to EEPROM. As long as NTAG 5 boost detects too less energy to be harvested from the field only INVENTORY command and READ/WRITE CONFIGURATION to access session registers will be handled. This feature ensures a stable system, as the host will only be supplied if there is sufficient energy available. However, during design phase we recommend disabling this power check.

The provided output power in general of course depends on many parameters like the strength of the NFC field, the antenna size, or the distance from the NFC device. The design ensures with the right settings, that V<sub>OUT</sub> is only enabled, when sufficient energy can be harvested from the NFC field.

1.8 V, 2.4 V or 3 V output voltage can be selected by coding EH\_VOUT\_V\_SEL accordingly.

Minimum required load current can be coded in EH\_VOUT\_I\_SEL configuration field.

V<sub>OUT</sub> and VCC need to be connected as soon as energy harvesting is used. Otherwise there is no EEPROM access possible from NFC perspective and status registers may contain invalid information.

Appropriate capacitor dependent on load needs to be placed between V<sub>OUT</sub> and ground to close energy gaps during miller pauses. An example circuit is illustrated in the figure below.

V<sub>OUT</sub> pin shall be kept floating (not connected) in case energy harvesting feature is not used. If energy harvesting is disabled, pin will be connected to GND internally.

With EH\_ENABLE configuration bit set to 1b, energy harvesting will be enabled after boot, automatically and all energy harvesting-related session register bits are meaningless.

When enabling energy harvesting via session registers, EH\_MODE, EH\_VOUT\_SEL and EH\_IOUT\_SEL needs to be configured properly in the related configuration bytes. EH\_ENABLE configuration bit need to be 0b in this case.

After boot, session registers can be used to first trigger current detection by setting EH\_TRIGGER to 1b, then poll for EH\_LOAD\_OK that gets 1b and finally set EH\_TRIGGER and EH\_ENABLE to 1b, or directly enable energy harvesting by setting EH\_TRIGGER and EH\_ENABLE bit to 1b (see [Table 103](#)).

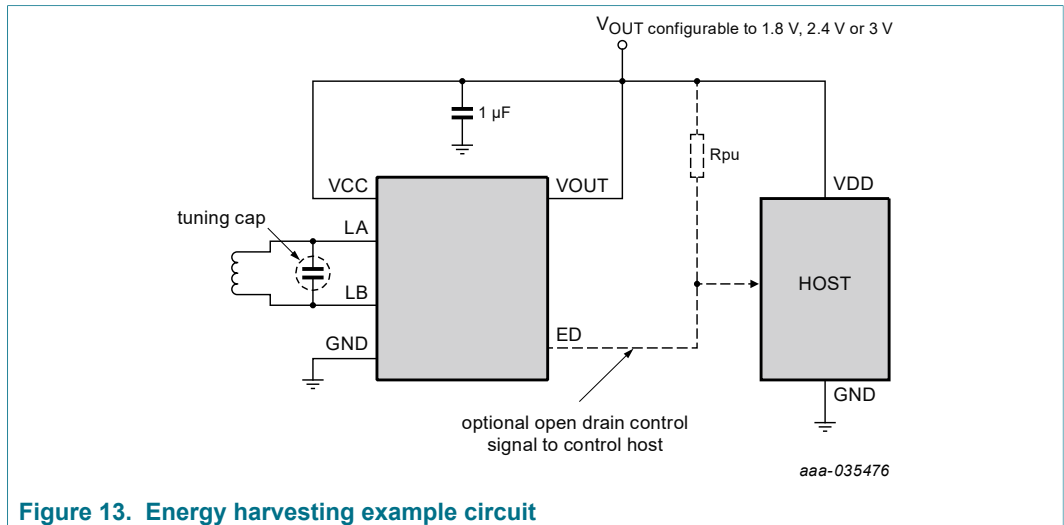


Figure 13. Energy harvesting example circuit

How to use energy harvesting in applications is described in [AN12365](#).

## 8.6 Security

NTAG 5 boost implements different levels to protect data. The easiest, but efficient method is to lock EEPROM to read only.

With the plain password authentication scheme, the memory can be split in three different parts with different access conditions.

With mutual AES authentication the memory is split in three different parts again, but with the advantage, that the password will never be transmitted in plain text.

Configuration area and SRAM can be protected from both interfaces as well.

Further implementation details can be found in [AN12366](#).

### 8.6.1 Locking EEPROM to read only

Independent on the split of the memory, the user memory may be locked to read-only. If the user EEPROM shall stay in read/write state, the LOCK BLOCK command can be disabled (see [Table 38](#)) and lock block sections can be locked (see [Table 86](#)). With these features, it can be ensured, NTAG 5 boost stays in read/write state.

Locking the complete EEPROM to read-only as defined in NFC Forum Type 5 Tag specification is quite time consuming. Every single block needs to be addressed by a LOCK BLOCK command (see [Section 8.2.4.5](#)). To accelerate this locking, NTAG 5 boost stores the information in the configuration area. With this feature, locking the EEPROM can be accelerated by a factor of 16. Note, that these bits are one time programmable (see [Section 8.1.3.31](#)) and blocks are indicated as locked in the Get Multiple Block Security Status response.

As long as I<sup>2</sup>C Lock Block Configuration bytes are not set to 1b, the user EEPROM may still be modified from I<sup>2</sup>C perspective.

Table 239. NFC Lock Block Configuration location

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
6Ah	106Ah	NFC_LOCK_BL0	NFC_LOCK_BL1	RFU	RFU
...	...	...	...		
89h	1089h	NFC_LOCK_BL62	NFC_LOCK_BL63		

Table 240. I<sup>2</sup>C Lock Block Configuration location

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
8Ah	108Ah	I2C_LOCK_BL0	I2C_LOCK_BL1	RFU	RFU
...	...	...	...		
91h	1091h	I2C_LOCK_BL62	I2C_LOCK_BL63		

8.6.2 Memory Areas

The memory may be split into three different configurable areas with different access conditions.

Highest priority has the 16-bit Protection Pointer PP\_AREA\_1. It splits the memory into an AREA\_0 and an AREA\_1 at the address configured with the PP\_AREA\_1.

Restricted area AREA\_1, starting from block address PP\_AREA\_1 is automatically protected by the AREA\_1 read and AREA\_1 write password in plain password mode.

In AES mode, a key with read and write privilege is needed to be able to access the restricted area.

To enable password protection to AREA\_1 from I<sup>2</sup>C perspective, I<sup>2</sup>C passwords need to be enabled by setting I<sup>2</sup>C key header (I2C\_KH) to active. In that case AREA\_1 read and write passwords need to be presented to NTAG 5 boost.

The split configured with the 16-bit Protection Pointer is the same for both, NFC and I<sup>2</sup>C perspective.

The area below this address can be split into two more areas with the 8-bit NFC\_PP\_AREA\_0-H (see Section 8.1.3.29) and the 8-bit I2C\_PP\_AREA\_0-H (see Section 8.1.3.11), independently of the NFC and I<sup>2</sup>C perspective.

NFC AREA\_0-L, usually used to store NDEF messages, starts from block 0. NFC AREA\_0-H, usually used as password protected area to store private data, starts from block address configured by the 8-bit NFC\_PP\_AREA\_0H and ends just before the block addressed with the PP\_AREA\_1 configuration byte. If PP\_AREA\_1 points outside the addressable memory space, only AREA\_0-L and AREA\_0-H are available.

I<sup>2</sup>C AREA\_0-L starts from block 0. I<sup>2</sup>C AREA\_0-H, starts from the block address configured by the 8-bit I2C\_PP\_AREA\_0H.

The concept is illustrated in the Figure below and further details can be found in AN12366.

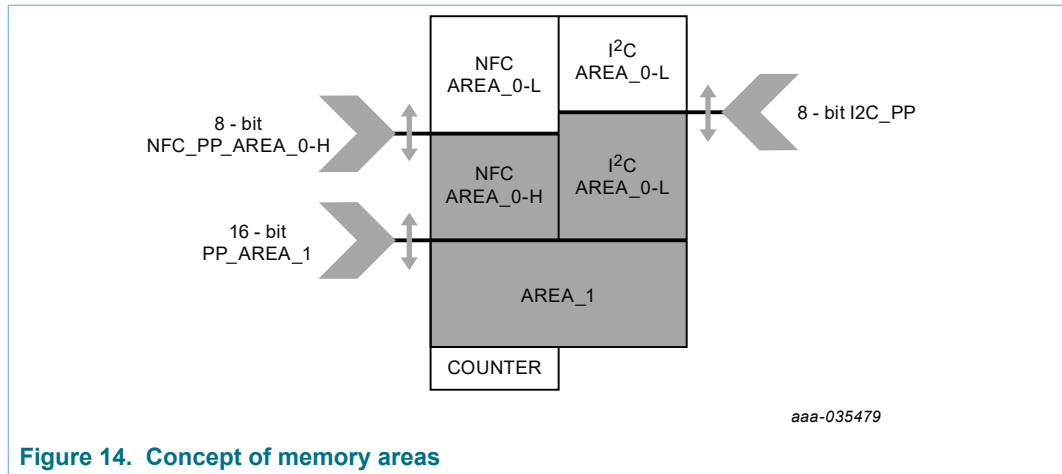


Figure 14. Concept of memory areas

### 8.6.3 Plain password authentication

NTAG 5 boost implements plain password authentication scheme from NFC perspective.

In summary, seven 32-bit passwords are available from NFC perspective.

- Read
- Write
- Restricted AREA\_1 Read
- Restricted AREA\_1 Write
- Destroy
- NFC Privacy password (is used to come out of NFC PRIVACY mode)
- EAS/AFI protection

64-bit password protection can be enabled for read and write operations.

A 32-bit password is used to authenticate, before doing memory operations. The mechanism is easy to use. After setting and locking the password, and setting right access conditions in initialization phase, the NFC Device needs to fetch a random number from the ICs. XORing the plain password and this random number results in used password to authenticate.

From I<sup>2</sup>C perspective, plain password authentication can be enabled with two 32-bit passwords for the restricted AREA\_1 and two for the rest of user EEPROM.

To resist brute force attacks, a negative authentication counter can be enabled.

How to use plain password authentication in applications is described in [AN12366](#).

### 8.6.4 AES authentication

NTAG 5 boost implements AES authentication from NFC perspective.

Highest security level of NTAG 5 boost is AES mutual authentication based on the Crypto Suite AES128 as defined in [ISO/IEC 29167-10](#). A 128-bit password is used to (mutual) authenticate, before doing memory operations.

From I<sup>2</sup>C perspective, only plain password authentication can be enabled.

How to use AES authentication in applications is described in [AN12366](#).

### 8.7 NFC privacy mode

In the privacy mode, the NTAG 5 boost is not traceable by its UID neither by data stored in the user memory. All NTAG 5 boost in the NFC PRIVACY mode will respond to an Inventory command with the UID E0 04 00 00 00 00 00 00, consequently also the user memory is NOT accessible.

NTOE: An anti-collision procedure is not possible in plain password mode.

In plain password mode ENABLE NFC PRIVACY Mode command (see [Section 8.2.4.3.9](#)) with a valid privacy password is used to set NTAG 5 boost to this mode and DISABLE NFC PRIVACY (see [Section 8.2.4.3.10](#)) is used to disable it again.

In AES mode, a valid mutual authentication (see [Section 8.2.4.4.4](#)) with the vendor-specific Purpose\_MAM2[3:0] (see [Table 172](#)) and a key with the privacy privilege set to 1b is needed to enable using Purpose\_MAM2 = 1001b and disable the NFC PRIVACY mode using Purpose\_MAM2 = 1000b until next NFC field reset or 1010b permanently.

NTAG 5 boost in NFC PRIVACY mode only support following commands:

- INVENTORY
- SELECT
- STAY QUIET
- RESET TO READY
- PICK RANDOM ID (in AES mode to allow an anti-collision procedure)
- GET RANDOM NUMBER
- DISABLE NFC PRIVACY in plain password mode
- AUTHENTICATE

### 8.8 Programmable Originality signature

NTAG 5 boost original signature is based on standard Elliptic Curve Cryptography (curve name secp128r1), according to the ECDSA algorithm. The use of a standard algorithm and curve ensures easy software integration of the originality check procedure in NFC devices without specific hardware requirements.

The UID is signed with an NXP private key and the resulting 32 byte signature is stored in the configuration memory during IC production.

The originality signature is stored in the configuration memory block 00h to block 07h.

**Table 241. 32 Byte Originality Signature**

Block Address		Byte 0	Byte 1	Byte 2	Byte 3
NFC	I <sup>2</sup> C				
00h	1000h	SIG0 (LSB)	SIG1	SIG2	SIG3
01h	1001h	SIG4	SIG5	SIG6	SIG7
02h	1002h	SIG8	SIG9	SIG10	SIG11
03h	1003h	SIG12	SIG13	SIG14	SIG15
04h	1004h	SIG16	SIG17	SIG18	SIG19
05h	1005h	SIG20	SIG21	SIG22	SIG23
06h	1006h	SIG24	SIG25	SIG26	SIG27
07h	1007h	SIG28	SIG29	SIG30	SIG31 (MSB)

This signature can be retrieved using the READ\_SIGNATURE command or with the READ CONFIG command and can be verified in the NFC device by using the corresponding ECC public key provided by NXP. In case the NXP public key is stored in the reader device, the complete signature verification procedure can be performed offline.

To verify the signature (for example with the use of the public domain crypto library OpenSSL) the tool domain parameters shall be set to secp128r1, defined within the standards for elliptic curve cryptography SEC.

NTAG 5 boost provides the possibility to customize the originality signature to personalize the IC individually for specific application. At delivery, the NTAG 5 boost is pre-programmed with the NXP originality signature described above. This signature is unlocked in the dedicated memory. If needed, the signature can be reprogrammed with a custom-specific signature using the WRITE CONFIG command during the personalization process by the customer. The signature can be permanently locked afterwards by setting the Config Header to "locked" with the WRITE CONFIG command to avoid further modifications.

In any case, it is recommended to permanently lock the originality signature during the initialization process by setting the Config Header to lock with the WRITE CONFIG command.

How to use and verify Originality Signature in applications is described in [AN11350](#).

How to generate Originality Signature is described in [AN11859](#).



## 9 Limiting values

Table 242. Limiting values In accordance with the Absolute Maximum Rating System (IEC 60134).

Symbol	Parameter	Conditions	Min	Max	Unit
T <sub>stg</sub>	storage temperature	all packages	-65	+150	°C
T <sub>j</sub>	junction temperature	EEPROM write operation	-	+95	°C
T <sub>j</sub>	junction temperature	EEPROM read, SRAM and register operation	-	+115	°C
V <sub>ESD</sub>	electrostatic discharge voltage	charged device model (CDM) <sup>[1]</sup>	-2	2	kV
		human body model (HBM) <sup>[2]</sup>	-2	2	kV
V <sub>CC</sub>	supply voltage	on pin V <sub>CC</sub>	-0.5	7.15	V
V <sub>CC</sub>	supply voltage	on pin V <sub>CC_TX</sub>	-0.5	7.15	V
V <sub>i</sub>	input voltage	on pin SDA, SCL, ED, HPD	-0.5	7.15	V
V <sub>i (RF)</sub>	RF input voltage	on pin LA/LB	-0.5	5.2	V <sub>p</sub>
V <sub>i</sub>	input voltage	on pin LA; LB is 0 V; sine wave of 13.56 MHz	-0.5	5.2	V <sub>p</sub>
		on pin LB; LA is 0 V; sine wave of 13.56 MHz	-0.5	5.2	V <sub>p</sub>
V <sub>i</sub>	input voltage	on pin LA_TX, LB_TX	-0.5	4.7	V
I <sub>i(max)</sub>	maximum input current	La/Lb; peak	-168	168	mA

[1] According to ANSI/ESDA/JEDEC JS-002.

[2] According to ANSI/ESDA/JEDEC JS-001.

## 10 Characteristics

### 10.1 Static Characteristics

Table 243. Characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
General						
f <sub>i</sub>	input frequency	ISO/IEC 15693	13.553	13.56	13.567	MHz
C <sub>i</sub>	input capacitance	LA-LB, Pin capacitance, VLA-LB @ 1.8Vp, Network Analyzer (13.56 MHz) @Room temp	-	15	-	pF
R <sub>i</sub>	Impedance from LA to LB	V <sub>LALB</sub> =1.8Vpp; passive mode	30	-	-	kΩ
R <sub>i</sub>	Impedance from LA to LB	V <sub>LALB</sub> =1.8Vpp; active mode	246	-	-	Ω
Operating conditions						
T <sub>amb</sub>	ambient temperature	T <sub>j</sub> <T <sub>j,max</sub> ; for EEPROM write operation	-40	25	85	°C
T <sub>amb</sub>	ambient temperature	T <sub>j</sub> <T <sub>j,max</sub> ; for EEPROM read, SRAM and register operation	-40	25	105	°C
R <sub>TH_JA</sub>	thermal resistance	JEDEC 2s2p board and SO8 package	-	82	-	K/W
R <sub>TH_JA</sub>	thermal resistance	JEDEC 2s2p board and TSSOP16 package	-	126	-	K/W
R <sub>TH_JA</sub>	thermal resistance	JEDEC 2s2p board and XQFN16 package	-	75	-	K/W
V <sub>CC</sub>	supply voltage	on pin V <sub>CC</sub>	1.62	-	5.5	V
V <sub>CC_TX</sub>	supply voltage	functional on pin V <sub>CC_TX</sub>	1.62	-	5.5	V
I <sub>i</sub>	input current	La/Lb; 12 A/m; RMS	-	-	43.75	mA
		La/Lb; 12 A/m; peak	-	-	61.87	mA
V <sub>sens</sub>	minimum receiver sensitivity	on LA/LB, V <sub>DD</sub> = 3.3 V	-	40	60	mVp
Active transmitter parameters						
R <sub>ON</sub>	low side output resistance TX1/TX2		1.66	-	5	Ω
R <sub>ON</sub>	high side output resistance TX1/TX2	configReg=0Fh, V <sub>TX</sub> = V <sub>VCC_TX</sub> - 300 mV	13	-	20	Ω
R <sub>ON</sub>	high side output resistance TX1/TX2	configReg=00h, V <sub>TX</sub> = V <sub>VCC_TX</sub> - 300 mV	1170	-	1970	Ω
V <sub>OH</sub>	High-level output voltage on TX1/TX2	V <sub>VCC_TX</sub> = 3.75 V	V <sub>VCC_TX</sub> - 0.20	V <sub>VCC_TX</sub> - 0.15	V <sub>VCC_TX</sub> - 0.10	V
V <sub>OHmax</sub>	High-level output voltage on TX1/TX2	V <sub>VCC_TX</sub> = 3.75 V	-	-	3.65	V

NTAG 5 boost - NFC Forum-compliant I<sup>2</sup>C bridge for tiny devices

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
Current consumption						
I <sub>VCC_TX</sub>	V <sub>CC_TX</sub> supply current	V <sub>CC_TX</sub> = 3.3 V; active communication TX	-	-	60.0	mA
I <sub>VCC_TX</sub>	V <sub>CC_TX</sub> supply current	V <sub>CC_TX</sub> = 5.5 V; battery assisted active communication TX	-	-	68.0	mA
I <sub>VCC_TX</sub>	V <sub>CC_TX</sub> supply current	V <sub>CC_TX</sub> = 3.3 V; active communication RX	-	-	18.0	mA
I <sub>VCC_TX</sub>	V <sub>CC_TX</sub> supply current	V <sub>CC_TX</sub> = 5.5 V; active communication	-	-	22.0	mA
I <sub>VCC</sub> +I <sub>VCC_TX</sub>	V <sub>CC_TX</sub> supply current	V <sub>CC_TX</sub> = 5.5 V; battery supplied no NFC communication ALM mode	-	450	750	μA
I <sub>VCC</sub> +I <sub>VCC_TX</sub>	V <sub>CC_TX</sub> supply current	V <sub>CC_TX</sub> = 5.5 V; battery supplied no NFC communication PLM mode	-	140	200	μA
I <sub>VCC</sub>	V <sub>CC</sub> supply current	V <sub>CC</sub> = 5.5 V; NFC passive communication no host activity	-	120	150	μA
I <sub>VCC</sub>	V <sub>CC</sub> supply current	V <sub>CC</sub> = 5.5 V, IDLE Mode. No NFC or Host activity	-	-	120	μA
I <sub>VCC</sub>	V <sub>CC</sub> supply current	V <sub>CC</sub> = 5.5 V, PWM/GPIO use case	-	128	175	μA
I <sub>VCC</sub>	V <sub>CC</sub> supply current	V <sub>CC</sub> = 1.8 V, 400 kHz, I <sup>2</sup> C read/write operation	-	115	163	μA
I <sub>VCC</sub>	V <sub>CC</sub> supply current	V <sub>CC</sub> = 3.3 V, 400 kHz, I <sup>2</sup> C read/write operation	-	115	163	μA
I <sub>VCC</sub>	V <sub>CC</sub> supply current	V <sub>CC</sub> = 5.5 V, 400 kHz, I <sup>2</sup> C read/write operation.	-	138	168	μA
I <sub>standby Vcc and Vcc_tx</sub>	standby current	V <sub>CC</sub> = 1.8 V wake-up via Active NFC level detector and host interface	-	9.0	40	μA
I <sub>standby Vcc and Vcc_tx</sub>	standby current	V <sub>CC</sub> = 3.3 V wake-up via Active NFC level detector and host interface	-	9.5	40	μA
I <sub>standby Vcc and Vcc_tx</sub>	standby current	V <sub>CC</sub> = 5.5 V wake-up via Active NFC level detector and I <sup>2</sup> C	-	10.0	40	μA
I <sub>standby Vcc and Vcc_tx</sub>	standby current	V <sub>CC</sub> = 1.8 V wake-up via I <sup>2</sup> C	-	5.5	16	μA
I <sub>standby Vcc and Vcc_tx</sub>	standby current	V <sub>CC</sub> = 3.3 V wake-up via I <sup>2</sup> C	-	5.9	18	μA
I <sub>standby Vcc and Vcc_tx</sub>	standby current	V <sub>CC</sub> = 5.5 V wake-up via I <sup>2</sup> C	-	6.9	21	μA

NTAG 5 boost - NFC Forum-compliant I<sup>2</sup>C bridge for tiny devices

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
I <sub>hrd_pwr_dwn</sub> V <sub>CC</sub> and V <sub>CC_tx</sub>	hard power down current	V <sub>CC</sub> = 1.8 V; XQFN16 package only	-	0.23	2.3	μA
I <sub>hrd_pwr_dwn</sub> V <sub>CC</sub> and V <sub>CC_tx</sub>	hard power down current	V <sub>CC</sub> = 3.3 V; XQFN16 package only	-	0.25	3.44	μA
I <sub>hrd_pwr_dwn</sub> V <sub>CC</sub> and V <sub>CC_tx</sub>	hard power down current	V <sub>CC</sub> = 5.5 V; XQFN16 package only	-	0.31	5.72	μA
<b>Energy harvesting VOUT pad</b>						
V <sub>out</sub>	output voltage	configured to 1.8 V; load current ≤ configured output current	1.62	-	1.98	V
		configured to 2.4 V; load current ≤ configured output current	2.16	-	2.64	V
		configured to 3.0 V; load current ≤ configured output current	2.7	-	3.3	V
I <sub>out</sub>	min. output current	at different regulated output voltages when current detection is enabled and dependent on selected output current value	0.4	-	12.5	mA
<b>ED pin characteristics</b>						
V <sub>OL</sub>	LOW-level output voltage	I <sub>OL</sub> = 3 mA	-	-	0.4	V
I <sub>IED</sub>	leakage current	V <sub>IN</sub> = 0 V to 5.5 V	10	-	1000	nA
<b>HPD pin characteristics for XQFN16 package</b>						
V <sub>IL</sub>	LOW-level input voltage		0	-	0.3*V <sub>CC</sub>	V
V <sub>IH</sub>	HIGH-level input voltage		0.7*V <sub>CC</sub>	-	V <sub>CC</sub>	V
I <sub>IL</sub>	LOW-level input current	V <sub>IN</sub> = 0 V	-1	-	-	μA
I <sub>IH</sub>	HIGH-level input current	V <sub>IN</sub> = 5.5 V	-	-	1	μA
C <sub>i</sub>	input capacitance		-	-	1.2	pF
<b>GPIO pad pin characteristics in I<sup>2</sup>C mode</b>						
V <sub>IH</sub>	HIGH-level input voltage		0.7*V <sub>CC</sub>	-	-	V
V <sub>IL</sub>	LOW-level input voltage		-	-	0.3*V <sub>CC</sub>	V
I <sub>IL</sub>	LOW-level input current	V <sub>IN</sub> = 0 V	-1	-	-	μA
I <sub>IH</sub>	HIGH-level input current	V <sub>IN</sub> = 5.5 V	-	-	1	μA
V <sub>OH</sub>	HIGH-level output voltage	I <sub>OH</sub> < 3 mA	0.7*V <sub>CC</sub>	-	V <sub>CC</sub>	V
V <sub>OL</sub>	LOW-level output voltage	I <sub>OL</sub> < 3 mA	0	-	0.4	V
C <sub>i</sub>	input capacitance		-	-	3.5	pF
C <sub>L</sub>	load capacitance		-	400	-	pF
<b>GPIO pad pin characteristics in GPIO mode</b>						
V <sub>IH</sub>	HIGH-level input voltage		0.7*V <sub>CC</sub>	-	-	V

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
V <sub>IL</sub>	LOW-level input voltage		-	-	0.3*V <sub>CC</sub>	V
I <sub>OL</sub>	static output low current	at V <sub>OL</sub> = 0.4 V	4	-	-	μA
I <sub>OH</sub>	static output high current	at V <sub>OH</sub> = V <sub>CC</sub> - 0.4 V	4	-	1	μA
I <sub>IL</sub>	LOW-level input current		-1	-	-	μA
I <sub>OH</sub>	HIGH-level output current		-	-	1	μA
C <sub>i</sub>	input capacitance		-	-	3.5	pF
C <sub>L</sub>	load capacitance		-	400	-	pF

## 10.2 Dynamic characteristics

Table 244.

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
<b>I<sup>2</sup>C master SCL/SDA pin characteristics</b>						
f <sub>SCL</sub>	SCL clock frequency	fast mode; C <sub>b</sub> < 400 pF	-	-	400	kHz
t <sub>SU,STA</sub>	set-up time for a (repeated) START condition	fast mode; C <sub>b</sub> < 400 pF	600	-	-	ns
t <sub>HD,STA</sub>	hold time (repeated) START condition	fast mode; C <sub>b</sub> < 400 pF	600	-	-	ns
t <sub>LOW</sub>	low period of the SCL clock	fast mode; C <sub>b</sub> < 400 pF	1.3	-	-	us
t <sub>HIGH</sub>	high period of the SCL clock	fast mode; C <sub>b</sub> < 400 pF	600	-	-	ns
t <sub>SU,DAT</sub>	data set-up time	fast mode; C <sub>b</sub> < 400 pF	100	-	-	ns
t <sub>HD,DAT</sub>	data hold time	fast mode; C <sub>b</sub> < 400 pF	0	-	900	ns
t <sub>rSDA</sub>	SDA rise time	CL = 100 pF, R <sub>pull-up</sub> = 2 K, Standard and fast mode	30	-	250	ns
t <sub>fSDA</sub>	SDA fall time	CL = 100 pF, R <sub>pull-up</sub> = 2 K, Standard and fast mode	30	-	250	ns
V <sub>hys</sub>	hysteresis of Schmitt trigger inputs	fast mode; C <sub>b</sub> < 400 pF	0.05 *V <sub>CC</sub>	-	-	V
<b>I<sup>2</sup>C slave SDA/SCL pin characteristics</b>						
t <sub>r</sub>	rise time	CL = 100 pF, R <sub>pull-up</sub> = 2 K, standard and fast mode	30	-	250	ns
t <sub>f</sub>	fall time	CL = 100 pF, R <sub>pull-up</sub> = 2 K, standard and fast mode	30	-	250	ns
<b>PWM AC timings</b>						
PWM <sub>freq</sub>	PWM output frequency		414	-	26400	Hz
Pulse Width	PWM signal pulse width		0.6	-	-	μs
PWM <sub>freq_tol</sub>	PWM output frequency tolerance		-	-	10	%

NTAG 5 boost - NFC Forum-compliant I<sup>2</sup>C bridge for tiny devices

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
PWM <sub>V_tol</sub>	PWM output voltage tolerance	I <sub>OH</sub> = 4 mA	V <sub>CC</sub> - 0.4	-	V <sub>CC</sub>	V
<b>GPIO pin characteristics</b>						
tr	rise time	CL = 20 pF; V <sub>CC</sub> = 1.8 V	-	-	20.9	ns
		CL = 20 pF; V <sub>CC</sub> = 3.3 V	-	-	10.92	ns
		CL = 20 pF; V <sub>CC</sub> = 5.5 V	-	-	8.22	ns
tf	fall time	CL = 20 pF; V <sub>CC</sub> = 1.8 V	-	-	129	ns
		CL = 20 pF; V <sub>CC</sub> = 3.3 V	-	-	77.9	ns
		CL = 20 pF; V <sub>CC</sub> = 5.5 V	-	-	66.9	ns
<b>Start Up time</b>						
t <sub>Start_VCC</sub>	V <sub>CC</sub> startup time from power OFF state. After this time, the IC is able to receive the command from I <sup>2</sup> C interface.		-	-	3	ms
t <sub>Start_RF</sub>	Startup time from NFC from Power OFF state. After this time, the IC is able to receive the command from NFC interface.		-	-	1	ms
<b>EEPROM characteristics</b>						
t <sub>ret</sub>	retention time	Ta < 85 °C	40	-	-	year
N <sub>endu(W)</sub>	write endurance	Ta < 85 °C	1000000	-	-	cycle

11 Package outline

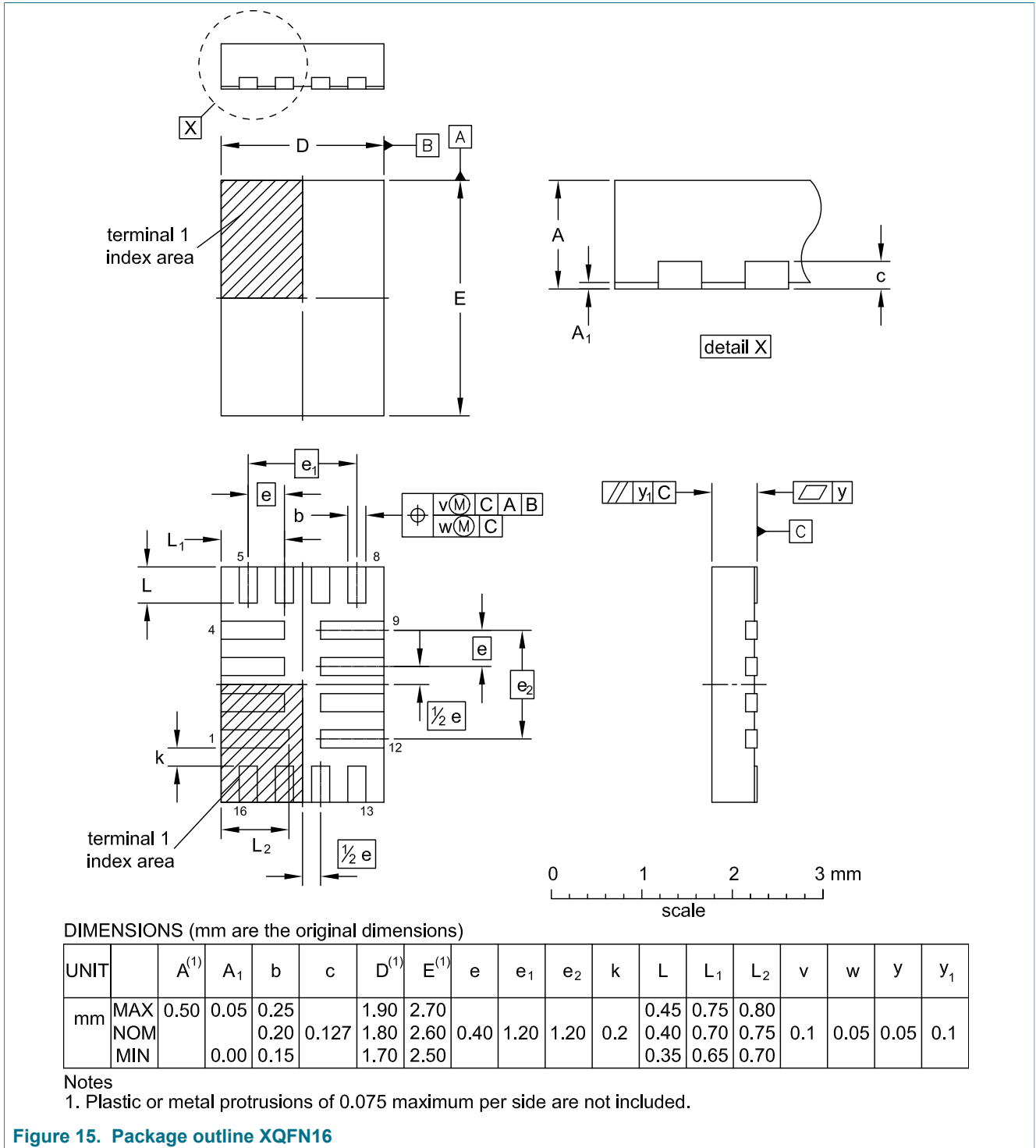
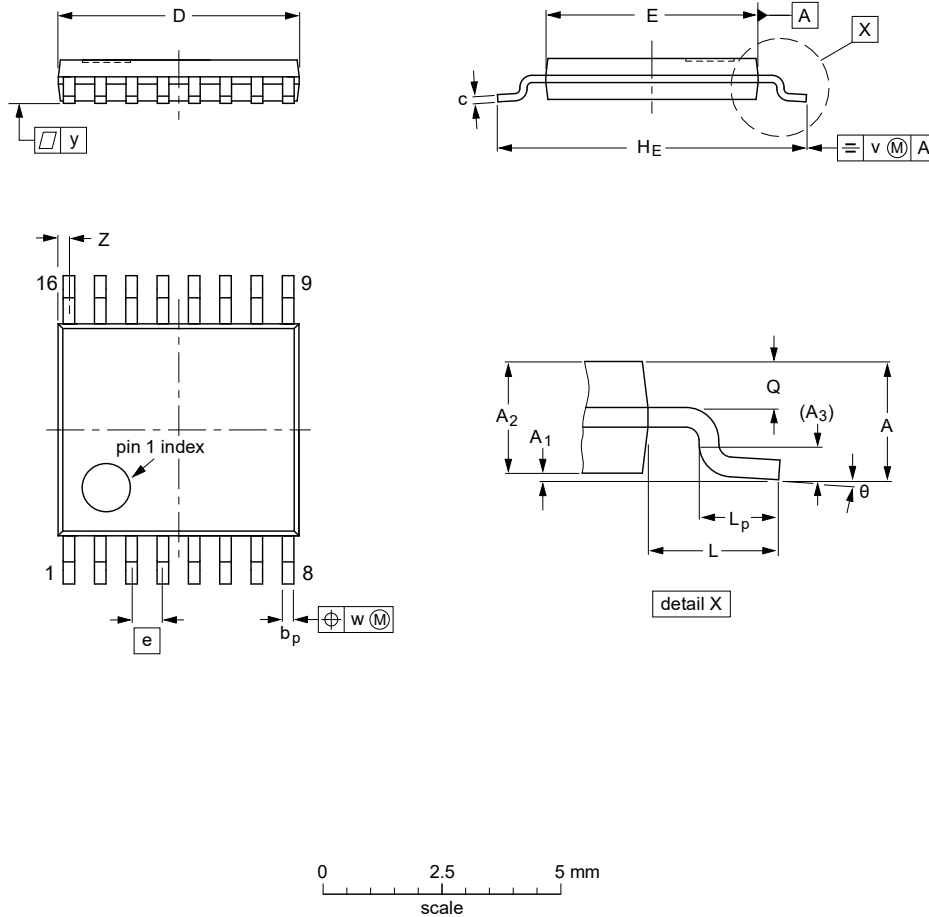


Figure 15. Package outline XQFN16

TSSOP16: plastic thin shrink small outline package; 16 leads; body width 4.4 mm

SOT403-1



**DIMENSIONS (mm are the original dimensions)**

UNIT	A max.	A <sub>1</sub>	A <sub>2</sub>	A <sub>3</sub>	b <sub>p</sub>	c	D <sup>(1)</sup>	E <sup>(2)</sup>	e	H <sub>E</sub>	L	L <sub>p</sub>	Q	v	w	y	Z <sup>(1)</sup>	θ
mm	1.1	0.15 0.05	0.95 0.80	0.25	0.30 0.19	0.2 0.1	5.1 4.9	4.5 4.3	0.65	6.6 6.2	1	0.75 0.50	0.4 0.3	0.2	0.13	0.1	0.40 0.06	8° 0°

**Notes**

1. Plastic or metal protrusions of 0.15 mm maximum per side are not included.
2. Plastic interlead protrusions of 0.25 mm maximum per side are not included.

OUTLINE VERSION	REFERENCES				EUROPEAN PROJECTION	ISSUE DATE
	IEC	JEDEC	JEITA			
SOT403-1		MO-153				99-12-27 03-02-18

Figure 16. Package outline TSSOP16



## 12 Handling information

**CAUTION**

This device is sensitive to ElectroStatic Discharge (ESD). Observe precautions for handling electrostatic sensitive devices. Such precautions are described in the *ANSI/ESD S20.20*, *IEC/ST 61340-5*, *JESD625-A* or equivalent standards.

## 13 Abbreviations

Table 245. Abbreviations

Acronym	Description
BoM	Bill of Material
CCH	Crypto Configuration Header
CH	Configuration Header
CID	Customer ID
ECC	Elliptic Curve Cryptography
EEPROM	Electrically Erasable Programmable Read-only Memory
GPIO	General Purpose Input Output
IC	Integrated Circuit
lsb	least significant bit
LSB	Least Significant Byte
Manuf. Code	IC Manufacturing Code of NXP is 04h.
msb	most significant bit
MSB	Most Significant Byte
NDEF	NFC Data Exchange Format
NFC	Near Field Communication
POR	Power On Reset
PWM	Pulse Width Modulation
RFU	Reserved for Future Use
TNEP	Tag NDEF Exchange Protocol
SRAM	Static Random-Access Memory

## 14 References

- [1] NFC Forum specification, Digital Protocol - Technical Specification Version 2.1  
2019-04-03 [T5T] NFC Forum™  
<https://nfc-forum.org/product-category/specification/>
- [2] NFC Forum specification, Type 5 Tag - Technical Specification Version 1.0  
2018-04-27 [T5T] NFC Forum™  
<https://nfc-forum.org/product-category/specification/>
- [3] NFC Forum specification, Tag NDEF Exchange Protocol - Technical Specification  
Version 1.0 2019-04-24 [TNEP] NFC Forum™  
<https://nfc-forum.org/our-work/specifications-and-application-documents/specifications/nfc-forum-candidate-technical-specifications/>
- [4] NFC Forum Personal Health Care Devices (PHDC) specification  
<https://nfc-forum.org/product-category/specification/>
- [5] ISO/IEC 15693  
<https://www.iso.org/ics/35.240.15/x/>
- [6] ISO/IEC 29167-10  
<https://www.iso.org/ics/35.240.15/x/>
- [7] AN11203 - NTAG 5 Use of PWM, GPIO and Event detection, doc.no. 5302xx  
<https://www.nxp.com/docs/en/application-note/AN11203.pdf>
- [8] AN12364 - NTAG 5 Bidirectional data exchange, doc.no. 5303xx  
<https://www.nxp.com/docs/en/application-note/AN12364.pdf>
- [9] AN11201 - NTAG 5 How to use energy harvesting, doc.no. 5304xx  
<https://www.nxp.com/docs/en/application-note/AN12365.pdf>
- [10] AN12366 - NTAG 5 Memory Configuration and Scalable Security, doc.no. 5305xx  
<https://www.nxp.com/docs/en/application-note/AN12366.pdf>
- [11] AN12368 - NTAG 5 Link I<sup>2</sup>C Master mode, doc.no. 5306xx  
<https://www.nxp.com/docs/en/application-note/AN12368.pdf>
- [12] AN12339 - Antenna Design Guide for NTAG 5  
<https://www.nxp.com/docs/en/application-note/AN12339.pdf>
- [13] AN11859 - MIFARE Ultralight and NTAG Generating Originality Signature  
<https://www.docstore.nxp.com/products>
- [14] AN11350 - NTAG Originality Signature Validation  
<https://www.nxp.com/confidential/AN11350>
- [15] UM10204 - I2C-bus specification and user manual  
<https://www.nxp.com/docs/en/user-guide/UM10204.pdf>

## 15 Revision history

Table 246. Revision history

Document ID	Release date	Data sheet status	Change notice	Supersedes
NTA5332 v.3.3	20200703	Product data sheet	-	v.3.2
Modifications:	<ul style="list-style-type: none"> <li>• ALM_MOD removed from (<a href="#">Table 10</a>)</li> <li>• Clarified, that ALM_STATUS_REG is only accessible from I<sup>2</sup>C perspective</li> <li>• Clarified, that I<sup>2</sup>C master registers are only accessible from NFC perspective</li> <li>• Clarified, that PICK RANDOM ID is only supported in AES mode</li> <li>• Information about Privacy mode was missing (see <a href="#">Section 8.7</a>)</li> <li>• Clarified that SRAM_CONFIG_PROT restricts access to blocks 37h to 54h (see <a href="#">Section 8.1.3.23</a>)</li> <li>• Type number in <a href="#">Table 2</a> corrected</li> <li>• Password identifier for access to restricted area added (see <a href="#">Table 133</a>)</li> <li>• Editorial updates</li> </ul>			
NTA5332 v.3.2	20200427	Product data sheet	-	v.3.1
NTA5332 v.3.1	20200324	Product data sheet	-	v.3.0
NTA5332 v.3.0	20200116	Product data sheet	-	v.2.0
NTA5332 v.2.0	20191002	Preliminary data sheet	-	v.1.0
NTA5332 v.1.0	20190528	Objective data sheet	-	-

## 16 Legal information

### 16.1 Data sheet status

Document status <sup>[1][2]</sup>	Product status <sup>[3]</sup>	Definition
Objective [short] data sheet	Development	This document contains data from the objective specification for product development.
Preliminary [short] data sheet	Qualification	This document contains data from the preliminary specification.
Product [short] data sheet	Production	This document contains the product specification.

- [1] Please consult the most recently issued document before initiating or completing a design.
- [2] The term 'short data sheet' is explained in section "Definitions".
- [3] The product status of device(s) described in this document may have changed since this document was published and may differ in case of multiple devices. The latest product status information is available on the Internet at URL <http://www.nxp.com>.

### 16.2 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.

**Short data sheet** — A short data sheet is an extract from a full data sheet with the same product type number(s) and title. A short data sheet is intended for quick reference only and should not be relied upon to contain detailed and full information. For detailed and full information see the relevant full data sheet, which is available on request via the local NXP Semiconductors sales office. In case of any inconsistency or conflict with the short data sheet, the full data sheet shall prevail.

**Product specification** — The information and data provided in a Product data sheet shall define the specification of the product as agreed between NXP Semiconductors and its customer, unless NXP Semiconductors and customer have explicitly agreed otherwise in writing. In no event however, shall an agreement be valid in which the NXP Semiconductors product is deemed to offer functions and qualities beyond those described in the Product data sheet.

### 16.3 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.

**Limiting values** — Stress above one or more limiting values (as defined in the Absolute Maximum Ratings System of IEC 60134) will cause permanent damage to the device. Limiting values are stress ratings only and (proper) operation of the device at these or any other conditions above those given in the Recommended operating conditions section (if present) or the Characteristics sections of this document is not warranted. Constant or repeated exposure to limiting values will permanently and irreversibly affect the quality and reliability of the device.

**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.

**No offer to sell or license** — Nothing in this document may be interpreted or construed as an offer to sell products that is open for acceptance or the grant, conveyance or implication of any license under any copyrights, patents or other industrial or intellectual property rights.

**Quick reference data** — The Quick reference data is an extract of the product data given in the Limiting values and Characteristics sections of this document, and as such is not complete, exhaustive or legally binding.

**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.

**Non-automotive qualified products** — Unless this data sheet expressly states that this specific NXP Semiconductors product is automotive qualified, the product is not suitable for automotive use. It is neither qualified nor tested in accordance with automotive testing or application requirements. NXP Semiconductors accepts no liability for inclusion and/or use of non-automotive qualified products in automotive equipment or applications. In the event that customer uses the product for design-in and use in automotive applications to automotive specifications and standards, customer (a) shall use the product without NXP Semiconductors' warranty of the product for such automotive applications, use and specifications, and (b) whenever customer uses the product for automotive applications beyond NXP Semiconductors' specifications such use shall be solely at customer's own risk, and (c) customer fully indemnifies NXP Semiconductors for any liability, damages or failed product claims resulting from customer design and use of the product for automotive applications beyond NXP Semiconductors' standard warranty and NXP Semiconductors' product specifications.

**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** — While NXP Semiconductors has implemented advanced security features, all products may be subject to unidentified vulnerabilities. Customers are responsible for the design and operation of their applications and products to reduce the effect of these vulnerabilities on customer's applications and products, and NXP Semiconductors accepts no liability for any vulnerability that is discovered. Customers should implement appropriate design and operating safeguards to minimize the risks associated with their applications and products.

## 16.4 Licenses

### Purchase of NXP ICs with NFC technology

Purchase of an NXP Semiconductors IC that complies with one of the Near Field Communication (NFC) standards ISO/IEC 18092 and ISO/IEC 21481 does not convey an implied license under any patent right infringed by implementation of any of those standards. Purchase of NXP Semiconductors IC does not include a license to any NXP patent (or other IP right) covering combinations of those products with other products, whether hardware or software.

## 16.5 Trademarks

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

**I<sup>2</sup>C-bus** — logo is a trademark of NXP B.V.

**NTAG** — is a trademark of NXP B.V.

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

Tables

Tab. 1.	Ordering information	4	Tab. 44.	Pulse Width Modulation Duty Cycle Configuration Location (PWMx_ON and PWMx_OFF)	32
Tab. 2.	Marking codes	5	Tab. 45.	Pulse Width Modulation Duty Cycle Session Register Location (PWMx_ON and PWMx_OFF)	32
Tab. 3.	Pin description for XQFN16	7	Tab. 46.	Pulse Width Modulation ON time Configuration Definition (PWMx_ON and PWMx_ON_REG)	32
Tab. 4.	Pin description for TSSOP16	8	Tab. 47.	Pulse Width Modulation OFF time Configuration Definition (PWMx_OFF and PWMx_OFF_REG)	32
Tab. 5.	User memory organization	10	Tab. 48.	Watch Dog Timer Configuration Location (WDT_CONFIG)	33
Tab. 6.	Memory content at delivery	10	Tab. 49.	Watch Dog Timer Configuration Register Location (WDT_CONFIG_REG)	33
Tab. 7.	COUNTER BLOCK data structure	11	Tab. 50.	Watch Dog Timer Enable Definition (WDT_ENABLE and WDT_EN_REG)	33
Tab. 8.	Preset counter data structure	11	Tab. 51.	Energy harvesting Configuration Location (EH_CONFIG)	33
Tab. 9.	Increment counter data structure	12	Tab. 52.	Energy harvesting Configuration Value Definition (EH_CONFIG)	33
Tab. 10.	Configuration Memory organization	12	Tab. 53.	Event Detection Configuration Location (ED_CONFIG)	34
Tab. 11.	32 Byte Originality Signature	17	Tab. 54.	Event Detection Configuration Register Location (ED_CONFIG_REG)	34
Tab. 12.	Configuration Header (CH) location	17	Tab. 55.	Event Detection Definition (ED_CONFIG and ED_CONFIG_REG)	34
Tab. 13.	Configuration Header Codes	17	Tab. 56.	Event Detection Clear Register Location (ED_INTR_CLEAR_REG)	36
Tab. 14.	Customer ID (CID) location	18	Tab. 57.	Event Detection Clear Register (ED_INTR_CLEAR_REG)	36
Tab. 15.	NFC Global Crypto Header (GCH) location	19	Tab. 58.	I2C Slave Configuration Location	36
Tab. 16.	Global Crypto Header Configuration in plain password mode	19	Tab. 59.	I2C Slave Configuration Definition (I2C_SLAVE_ADDR)	36
Tab. 17.	Global Crypto Header Configuration Value in AES mode	19	Tab. 60.	I2C Slave Configuration Definition (I2C_SLAVE_CONFIG)	36
Tab. 18.	Crypto Configuration Header (CCH) location	20	Tab. 61.	I2C Master Clock Settings Configuration Location (I2C_MASTER_CONFIG)	37
Tab. 19.	Crypto Configuration Header Values	20	Tab. 62.	I2C Master Clock Configuration Definition (I2C_MASTER_SCL_LOW)	37
Tab. 20.	NFC Authentication Limit Counter (NFC_AUTH_LIMIT) location	21	Tab. 63.	I2C Master Clock Configuration Definition (I2C_MASTER_SCL_HIGH)	37
Tab. 21.	NFC Key Header (KHx) location	22	Tab. 64.	Device Security Configuration Byte location	37
Tab. 22.	NFC Key Header Values	22	Tab. 65.	Device Security Byte Definition (DEV_SEC_CONFIG)	37
Tab. 23.	NFC Key Privileges (KPx) location	22	Tab. 66.	SRAM and Configuration Byte location	38
Tab. 24.	Definition of NFC Key Privileges bytes KPx	23	Tab. 67.	SRAM and Configuration Protection (SRAM_CONF_PROT)	38
Tab. 25.	Key location	23	Tab. 68.	Restricted AREA_1 Pointer location	39
Tab. 26.	Plain Password location	24	Tab. 69.	Memory organization example from NFC perspective	39
Tab. 27.	I2C Password location	25	Tab. 70.	Active NFC Configuration location	39
Tab. 28.	I2C Key Authenticate Password location	25	Tab. 71.	ALM Configuration 0 (ALM_CONF_00)	40
Tab. 29.	I2C Key Header (I2C_KH) location	26	Tab. 72.	ALM Configuration 1 (ALM_CONF_01)	40
Tab. 30.	I2C Key Header Values	26	Tab. 73.	ALM Configuration 2 (ALM_CONF_02)	40
Tab. 31.	I2C Protection Pointer and Configuration location	26			
Tab. 32.	I2C Memory organization example	26			
Tab. 33.	I2C Protection Pointer Configuration (I2C_PPC)	27			
Tab. 34.	I2C Authentication Limit Counter (I2C_AUTH_LIMIT) location	28			
Tab. 35.	Configuration Bytes Location (CONFIG)	28			
Tab. 36.	Configuration Definition (CONFIG_0)	28			
Tab. 37.	Configuration Definition (CONFIG_1)	29			
Tab. 38.	Configuration Definition (CONFIG_2)	29			
Tab. 39.	Synchronization Block Bytes Location (SYNCH_DATA_BLOCK)	30			
Tab. 40.	Synchronization Block Register Bytes Location (SYNCH_DATA_BLOCK_REG)	30			
Tab. 41.	PWM and GPIO Configuration Location (PWM_GPIO_CONFIG)	31			
Tab. 42.	PWM and GPIO Configuration Definition (PWM_GPIO_CONFIG_0)	31			
Tab. 43.	PWM and GPIO Configuration Definition (PWM_GPIO_CONFIG_1 and PWM_GPIO_CONFIG_1_REG)	31			

Tab. 74.	ALM Configuration 3 (ALM_CONF_03) .....	40	Tab. 113.	ALM Status Register Location (ALM_STATUS_REG) .....	57
Tab. 75.	ALM Look-up-table (ALM_LUT_dd) .....	40	Tab. 114.	ALM Status Register (ALM_STATUS_REG) ...	57
Tab. 76.	Active NFC Configuration Default Content .....	41	Tab. 115.	SRAM mirroring .....	58
Tab. 77.	Application Family Identifier (AFI) location .....	41	Tab. 116.	SRAM mirroring with default content .....	58
Tab. 78.	Data Storage Format Identifier (DSFID) location .....	42	Tab. 117.	SRAM COPY BYTES (SRAM_COPY_BYTES) .....	59
Tab. 79.	Electronic Article Surveillance ID (EASID) location .....	42	Tab. 118.	SRAM_COPY_BYTES Definition .....	59
Tab. 80.	NFC Protection Pointer (NFC PP) location .....	42	Tab. 119.	SRAM Default Content location .....	59
Tab. 81.	Memory organization example .....	42	Tab. 120.	Bit rates from reader to tag .....	59
Tab. 82.	NFC Protection Pointer Conditions (NFC_PPC) location .....	43	Tab. 121.	Bit rates from tag to reader .....	59
Tab. 83.	NFC Protection Pointer Configuration (NFC_PPC) .....	43	Tab. 122.	NFC command set supported by NTAG 5 boost .....	63
Tab. 84.	NFC Lock Block Configuration location .....	44	Tab. 123.	READ CONFIG request format .....	66
Tab. 85.	I2C Lock Block Configuration location .....	44	Tab. 124.	READ CONFIG response format when Error_flag is NOT set .....	66
Tab. 86.	Device configuration section lock bytes location .....	44	Tab. 125.	READ CONFIGURATION response format when Error_flag is set .....	66
Tab. 87.	NFC configuration section lock byte 0 definition (NFC_SECTION_LOCK_0) .....	45	Tab. 126.	WRITE CONFIG request format .....	67
Tab. 88.	NFC configuration section lock Byte 1 definition (NFC_SECTION_LOCK_1) .....	45	Tab. 127.	WRITE CONFIG response format when Error_flag is NOT set .....	67
Tab. 89.	I2C configuration section lock byte 0 definition (I2C_SECTION_LOCK_0) .....	46	Tab. 128.	WRITE CONFIG response format when Error_flag is set .....	67
Tab. 90.	I2C configuration section lock byte 1 definition (I2C_SECTION_LOCK_1) .....	46	Tab. 129.	GET RANDOM NUMBER request format .....	67
Tab. 91.	Session Register Location .....	47	Tab. 130.	GET RANDOM NUMBER response format when Error_flag is NOT set .....	67
Tab. 92.	Status Register Location .....	48	Tab. 131.	GET RANDOM NUMBER response format when Error_flag is set .....	68
Tab. 93.	Status 0 Register .....	49	Tab. 132.	SET PASSWORD request format .....	68
Tab. 94.	Status 1 Register .....	49	Tab. 133.	Password Identifier .....	68
Tab. 95.	Configuration Register Location (CONFIG_REG) .....	50	Tab. 134.	SET PASSWORD response format when Error_flag is NOT set .....	68
Tab. 96.	Configuration Definition (CONFIG_0_REG) ...	50	Tab. 135.	SET PASSWORD response format when Error_flag is set .....	69
Tab. 97.	Configuration Definition (CONFIG_1_REG) ...	51	Tab. 136.	WRITE PASSWORD request format .....	69
Tab. 98.	Configuration Definition (CONFIG_2_REG) ...	51	Tab. 137.	WRITE PASSWORD response format when Error_flag is NOT set .....	69
Tab. 99.	PWM and GPIO Configuration Register Location (PWM_GPIO_CONFIG_REG) .....	52	Tab. 138.	WRITE PASSWORD response format when Error_flag is set .....	69
Tab. 100.	PWM and GPIO Configuration Register Definition (PWM_GPIO_CONFIG_0_REG) .....	52	Tab. 139.	LOCK PASSWORD request format .....	70
Tab. 101.	PWM and GPIO Configuration Register Definition (PWM_GPIO_CONFIG_1_REG) .....	53	Tab. 140.	LOCK PASSWORD response format when Error_flag is NOT set .....	70
Tab. 102.	Energy Harvesting Configuration Register Location (EH_CONFIG_REG) .....	54	Tab. 141.	LOCK PASSWORD response format when Error_flag is set .....	70
Tab. 103.	Energy Harvesting Register Value Definition (EH_CONFIG_REG) .....	54	Tab. 142.	64 BIT PASSWORD PROTECTION request format .....	70
Tab. 104.	I2C Slave Configuration Register Location .....	54	Tab. 143.	64 BIT PASSWORD PROTECTION response format when Error_flag is NOT set ...	71
Tab. 105.	I2C Slave Configuration Definition (I2C_SLAVE_ADDR_REG) .....	55	Tab. 144.	64 BIT PASSWORD PROTECTION response format when Error_flag is NOT set ...	71
Tab. 106.	I2C Slave Configuration Definition (I2C_SLAVE_CONFIG_REG) .....	55	Tab. 145.	Memory organization .....	71
Tab. 107.	RESET_GEN_REG location .....	55	Tab. 146.	PROTECT PAGE request format .....	72
Tab. 108.	ED_INTR_CLEAR_REG location .....	56	Tab. 147.	Extended Protection status byte .....	72
Tab. 109.	I2C Master Configuration Register Location .....	56	Tab. 148.	Protection status bits definition in plain password mode .....	72
Tab. 110.	I2C Slave Address used in I2C master transaction (I2C_M_S_ADD_REG) .....	56	Tab. 149.	Protection status bits definition in AES mode ...	73
Tab. 111.	I2C Master Data Length Definition (I2C_M_LEN_REG) .....	56	Tab. 150.	PROTECT PAGE response format when Error_flag is NOT set .....	73
Tab. 112.	I2C Master Status Definition (I2C_M_STATUS_REG) .....	56			



Tab. 151. PROTECT PAGE response format when Error_flag is set .....	73	Tab. 190. READ SRAM response format when Error_flag is set .....	85
Tab. 152. LOCK PAGE PROTECTION CONDITION request format .....	73	Tab. 191. WRITE SRAM request format .....	86
Tab. 153. LOCK PAGE PROTECTION CONDITION response format when Error_flag is NOT set ...	74	Tab. 192. WRITE SRAM response format when Error_flag is NOT set .....	86
Tab. 154. LOCK PAGE PROTECTION CONDITION response format when Error_flag is set .....	74	Tab. 193. WRITE SRAM response format when Error_flag is set .....	86
Tab. 155. DESTROY request format .....	74	Tab. 194. READ SIGNATURE request format .....	86
Tab. 156. DESTROY response format when Error_flag is NOT set .....	74	Tab. 195. READ SIGNATURE response format when Error_flag is NOT set .....	86
Tab. 157. DESTROY response format when Error_flag is set .....	74	Tab. 196. READ SIGNATURE response format when Error_flag is set .....	87
Tab. 158. ENABLE NFC PRIVACY request format .....	75	Tab. 197. WRITE I2C request format .....	88
Tab. 159. ENABLE NFC PRIVACY response format when Error_flag is NOT set .....	75	Tab. 198. I2C param byte .....	88
Tab. 160. ENABLE NFC PRIVACY response format when Error_flag is set .....	75	Tab. 199. WRITE I2C response format when Error_flag is NOT set .....	89
Tab. 161. DISABLE NFC PRIVACY request format .....	76	Tab. 200. WRITE I2C response format when Error_flag is set .....	89
Tab. 162. DISABLE NFC PRIVACY response format when Error_flag is NOT set .....	76	Tab. 201. READ I2C request format .....	89
Tab. 163. DISABLE NFC PRIVACY response format when Error_flag is set .....	76	Tab. 202. I2C param byte .....	89
Tab. 164. AUTHENTICATE request format .....	77	Tab. 203. READ I2C response format when Error_flag is NOT set .....	89
Tab. 165. AUTHENTICATE response format when Error_flag is NOT set(in process reply) .....	77	Tab. 204. READ I2C response format when Error_flag is set .....	90
Tab. 166. AUTHENTICATE response format when Error_flag is set .....	77	Tab. 205. SET EAS request format .....	90
Tab. 167. Message format for TAM1 .....	77	Tab. 206. SET EAS response format when Error_flag is NOT set .....	90
Tab. 168. TResponse for TAM1 .....	77	Tab. 207. SET EAS response format when Error_flag is set .....	90
Tab. 169. Message format for MAM1 .....	78	Tab. 208. RESET EAS request format .....	91
Tab. 170. TResponse for MAM1 .....	78	Tab. 209. RESET EAS response format when Error_flag is NOT set .....	91
Tab. 171. Message format for MAM2 .....	78	Tab. 210. RESET EAS response format when Error_flag is set .....	91
Tab. 172. Definition of Purpose_MAM2 .....	78	Tab. 211. LOCK EAS request format .....	91
Tab. 173. TResponse for MAM2 .....	79	Tab. 212. LOCK EAS response format when Error_flag is NOT set .....	92
Tab. 174. CHALLENGE request format .....	79	Tab. 213. LOCK EAS response format when Error_flag is set .....	92
Tab. 175. Message format .....	79	Tab. 214. EAS ALARM Request format .....	92
Tab. 176. READBUFFER request format .....	80	Tab. 215. EAS ALARM Response format (Option flag logic 0) .....	92
Tab. 177. READBUFFER response format when Error_flag is NOT set .....	80	Tab. 216. EAS ALARM Response format(Option flag logic 1) .....	93
Tab. 178. TResponse .....	80	Tab. 217. EAS ALAMR response format when Error_flag is set .....	93
Tab. 179. READBUFFER response format when Error_flag is set .....	80	Tab. 218. PROTECT EAS/AFI request format .....	93
Tab. 180. INVENTORY READ request format .....	81	Tab. 219. PROTECT EAS/AFI response format when Error_flag is NOT set .....	93
Tab. 181. INVENTORY READ response format: Option flag logic 0b .....	82	Tab. 220. PROTECT EAS/AFI response format when Error_flag is set .....	93
Tab. 182. INVENTORY READ response format: Option flag logic 1b .....	82	Tab. 221. WRITE EAS ID request format .....	94
Tab. 183. Example: mask length = 30 .....	82	Tab. 222. WRITE EAS ID response format when Error_flag is NOT set .....	94
Tab. 184. Inventory Read (extended mode) request format .....	83	Tab. 223. WRITE EAS ID response format when Error_flag is set .....	94
Tab. 185. Extended options .....	83	Tab. 224. GET NXP SYSTEM INFORMATION request format .....	95
Tab. 186. Inventory Read (extended mode) response format: Option_flag 1b .....	84		
Tab. 187. Example .....	84		
Tab. 188. READ SRAM request format .....	85		
Tab. 189. READ SRAM response format when Error_flag is NOT set .....	85		

Tab. 225. GET NXP SYSTEM INFORMATION response format when Error_flag is NOT set ...	95	Tab. 236. Random ID .....	98
Tab. 226. Protection Pointer condition byte .....	95	Tab. 237. Default NTAG 5 I2C address from I2C .....	100
Tab. 227. Lock bits byte .....	95	Tab. 238. Pulse Width Modulation Frequency .....	104
Tab. 228. Feature flags byte 0 .....	96	Tab. 239. NFC Lock Block Configuration location .....	109
Tab. 229. Feature flags byte 1 .....	96	Tab. 240. I2C Lock Block Configuration location .....	109
Tab. 230. Feature flags byte 2 .....	96	Tab. 241. 32 Byte Originality Signature .....	111
Tab. 231. Feature flags byte 3 .....	97	Tab. 242. Limiting values In accordance with the Absolute Maximum Rating System (IEC 60134). .....	113
Tab. 232. GET NXP SYSTEM INFORMATION response format when Error_flag is set .....	97	Tab. 243. Characteristics .....	114
Tab. 233. PICK RANDOM ID request format .....	97	Tab. 244. ....	117
Tab. 234. PICK RANDOM ID response format when Error_flag is NOT set .....	97	Tab. 245. Abbreviations .....	122
Tab. 235. PICK RANDOM ID response format when Error_flag is set .....	98	Tab. 246. Revision history .....	124

## Figures

Fig. 1.	NTAG 5 boost overview .....	1	Fig. 10.	READ MEMORY and WRITE MEMORY command .....	102
Fig. 2.	Block diagram NTAG 5 boost .....	6	Fig. 11.	READ REGISTER and WRITE REGISTER command .....	103
Fig. 3.	Pin configuration for XQFN16 package .....	7	Fig. 12.	Pulse Width Modulation Example .....	105
Fig. 4.	Pin configuration for TSSOP16 package .....	8	Fig. 13.	Energy harvesting example circuit .....	108
Fig. 5.	NTAG 5 boost output driver .....	60	Fig. 14.	Concept of memory areas .....	110
Fig. 6.	State Machine and State Transitions .....	61	Fig. 15.	Package outline XQFN16 .....	119
Fig. 7.	I2C Master write principle .....	87	Fig. 16.	Package outline TSSOP16 .....	120
Fig. 8.	I2C Master read principle .....	88			
Fig. 9.	I2C bus protocol .....	99			

## Contents

<b>1</b>	<b>General description</b>	<b>1</b>	8.1.4.8	Event detection register	54
<b>2</b>	<b>Features and benefits</b>	<b>2</b>	8.1.4.9	I2C slave register settings	54
<b>3</b>	<b>Applications</b>	<b>3</b>	8.1.4.10	System reset generation	55
<b>4</b>	<b>Ordering information</b>	<b>4</b>	8.1.4.11	Clear event detection register	55
<b>5</b>	<b>Marking</b>	<b>5</b>	8.1.4.12	I2C master status registers	56
<b>6</b>	<b>Block diagram</b>	<b>6</b>	8.1.4.13	ALM status registers	57
<b>7</b>	<b>Pinning information</b>	<b>7</b>	8.1.5	SRAM	57
<b>8</b>	<b>Functional description</b>	<b>9</b>	8.2	NFC interface	59
8.1	Memory Organization	9	8.2.1	Passive communication mode	60
8.1.1	General	9	8.2.2	Active communication mode	60
8.1.2	User memory	9	8.2.3	State diagram and state transitions	60
8.1.2.1	16-bit counter	10	8.2.3.1	POWER-OFF state	61
8.1.3	Configuration memory	12	8.2.3.2	READY state	61
8.1.3.1	Originality Signature	17	8.2.3.3	SELECTED state	62
8.1.3.2	Configuration Header	17	8.2.3.4	SELECTED SECURE state	62
8.1.3.3	Customer ID (CID)	18	8.2.3.5	QUIET state	63
8.1.3.4	NFC Global Crypto Header	18	8.2.4	Command set	63
8.1.3.5	NFC Crypto Configuration Header	20	8.2.4.1	Commands for state transitions	65
8.1.3.6	NFC Authentication Limit Counter	20	8.2.4.2	Configuration operations	66
8.1.3.7	NFC Key Header	21	8.2.4.3	PWD Authentication	67
8.1.3.8	NFC Key Privileges	22	8.2.4.4	AES Authentication	76
8.1.3.9	Keys and passwords	23	8.2.4.5	Memory operations	80
8.1.3.10	I2C Key Header	25	8.2.4.6	SRAM operations	85
8.1.3.11	I2C Protection Pointer and Condition	26	8.2.4.7	Originality Signature	86
8.1.3.12	I2C Authentication Limit Counter	28	8.2.4.8	I2C Transparent Channel	87
8.1.3.13	Configuration	28	8.2.4.9	Other	90
8.1.3.14	Synchronization Block	30	8.2.5	Data integrity	98
8.1.3.15	Pulse Width Modulation and GPIO configuration	30	8.2.6	Error Handling	98
8.1.3.16	Pulse Width Modulation duty cycle settings	32	8.2.6.1	Transmission Errors	98
8.1.3.17	Watch Dog Timer settings	32	8.2.6.2	Not supported commands or options	98
8.1.3.18	Energy harvesting settings	33	8.3	Wired Interface	99
8.1.3.19	Event detection pin configuration settings	34	8.3.1	I2C interface	99
8.1.3.20	I2C slave configuration settings	36	8.3.1.1	Slave mode	99
8.1.3.21	I2C master clock configuration settings	36	8.3.1.2	Master mode	101
8.1.3.22	Device security configuration bytes	37	8.3.1.3	Watch Dog Timer	101
8.1.3.23	SRAM and Configuration protection	38	8.3.1.4	Command Set	102
8.1.3.24	Restricted AREA_1 pointer	38	8.3.1.5	Error Handling	103
8.1.3.25	Active NFC configuration	39	8.3.2	Event detection	103
8.1.3.26	Application Family Identifier	41	8.3.3	GPIO	104
8.1.3.27	Data Storage Format Identifier	42	8.3.4	PWM	104
8.1.3.28	Electronic Article Surveillance ID	42	8.3.5	Standby mode	105
8.1.3.29	NFC protection pointer	42	8.3.6	Hard power-down mode	105
8.1.3.30	NFC Protection Pointer Conditions	43	8.4	Arbitration between NFC and I2C interface	106
8.1.3.31	NFC lock bytes	43	8.4.1	NFC Mode	106
8.1.3.32	I2C lock bytes	44	8.4.2	I2C Mode	106
8.1.3.33	Device configuration section lock bytes	44	8.4.3	Normal Mode	106
8.1.4	Session registers	47	8.4.4	SRAM Mirror Mode	106
8.1.4.1	Status register	48	8.4.5	SRAM Pass-Through Mode	106
8.1.4.2	Configuration register	50	8.4.6	SRAM PHDC Mode	107
8.1.4.3	Synchronization block register	52	8.5	Energy harvesting	107
8.1.4.4	Pulse Width Modulation and GPIO configuration register	52	8.6	Security	108
8.1.4.5	Pulse Width Modulation duty cycle register	53	8.6.1	Locking EEPROM to read only	108
8.1.4.6	Watch Dog Timer register	53	8.6.2	Memory Areas	109
8.1.4.7	Energy harvesting register	53	8.6.3	Plain password authentication	110
			8.6.4	AES authentication	110
			8.7	NFC privacy mode	111

8.8	Programmable Originality signature .....	111
<b>9</b>	<b>Limiting values .....</b>	<b>113</b>
<b>10</b>	<b>Characteristics .....</b>	<b>114</b>
10.1	Static Characteristics .....	114
10.2	Dynamic characteristics .....	117
<b>11</b>	<b>Package outline .....</b>	<b>119</b>
<b>12</b>	<b>Handling information .....</b>	<b>121</b>
<b>13</b>	<b>Abbreviations .....</b>	<b>122</b>
<b>14</b>	<b>References .....</b>	<b>123</b>
<b>15</b>	<b>Revision history .....</b>	<b>124</b>
<b>16</b>	<b>Legal information .....</b>	<b>125</b>

---

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

---

© NXP B.V. 2020.

All rights reserved.

For more information, please visit: <http://www.nxp.com>

For sales office addresses, please send an email to: [salesaddresses@nxp.com](mailto:salesaddresses@nxp.com)

Date of release: 3 July 2020  
Document number: 544733

# Mouser Electronics

Authorized Distributor

Click to View Pricing, Inventory, Delivery & Lifecycle Information:

[NXP:](#)

[NTA53321G0FTTZ](#) [NTA53321G0FHKZ](#)