

---

## NDEF management with ST25DVxxx and ST25TVxxx products

### Introduction

The ST25DVxxx are dual EEPROM devices designed to be accessed via two different interfaces: a wired I<sup>2</sup>C interface and a standard contactless ISO 15693 RFID interface.

ST25TVxxx devices are NFC and RFID tag offering up to 64 Kbits of electrically erasable programmable memory (EEPROM).

ST25DVxxx and ST25TVxxx can act as an NFC Type 5 Tag supporting NDEF message.

The purpose of this document is to present the way to configure ST25DVxxx and ST25TVxxx so that they can be detected as NFC tags. It also lists the commands and operations compliant with the NFC Forum environment.

This application note applies to the products listed in [Table 1. Applicable products.](#)

**Table 1. Applicable products**

Reference	Part number
ST25DVxxx	ST25DV04K, ST25DV16K, ST25DV64K
ST25TVxxx	ST25TV16K, ST25TV64K

# 1 Acronyms and notational conventions

**Table 2. List of acronyms**

Acronym	Definition
CC	Capability container (see <a href="#">Section 3.1 Capability container (CC)</a> for details)
EEPROM	Electrically-erasable programmable read-only memory
I <sup>2</sup> C	Inter-integrated circuit
ISO	International organization for standardization
IT	Interrupt
MLEN	Encoded memory length (see <a href="#">Section 3.3 T5T_Area and MLEN</a> for details)
NDEF	NFC data exchange (see <a href="#">Section A Appendix</a> for details)
R	Read
RF	Radio frequency
RFID	Radio frequency identification
RFU	Reserved for future use
R/W	Read / Write
T5T	Type 5 tag (see <a href="#">Section A Appendix</a> for details)
T5T_Area	Type 5 tag area
W	Write

The following conventions and notations apply in this document unless otherwise stated.

## 1.1 Binary number representation

Binary numbers are represented by strings of 0 and 1 digits, with the most significant bit on the left, the least significant bit on the right, and a 'b' suffix added at the end.

Example: 11110101b

## 1.2 Hexadecimal number representation

Hexadecimal numbers are represented by strings of numbers from 0 to 9 and letters from A to F, and an 'h' suffix added at the end. The most significant byte is shown on the left and the least significant byte on the right.

Example: F5h

## 1.3 Decimal number representation

Decimal numbers are represented without any trailing character.

Example: 245

## 2 NFC Forum

The NFC Forum specification reduces the amount of mandatory resources to activate ISO/IEC 15693 device as a T5T (type 5 tag).

RF exchanges are performed as follows:

- For the up link: 100 % amplitude modulation and 1 over 4 data coding with a data rate of 26 Kbit/s
- For the down link: Single sub-carrier load modulation with Manchester data coding at a high data rate of 26 Kbit/s

The command set specified by the NFC Forum is composed of the following commands:

- Read\_Single\_Block
- Write\_Single\_Block
- Lock\_Single\_Block
- Read\_Multiple\_Block
- Extended\_Read\_Single\_Block
- Extended\_Write\_Single\_Block
- Extended\_Lock\_Single\_Block
- Extended\_Read\_Multiple\_Block
- Select
- SLPV\_Req

This reduced command set allows a NFC device to manage NDEF (NFC data exchange) implemented in a ST25DVxxx Series or a ST25TVxxx devices.

Please refer to the NFC Forum Type 5 Tag technical Specification for details.

### 3 ST25DVxxx and ST25TVxxx configuration to support a NDEF message

The products in the ST25DVxxx and ST25TVxxx devices must be configured as T5T so that they can handle NDEF messages.

Such a configuration consists in including a valid CC (capability container) in the first memory blocks.

**Caution:**

For ST25DV16K, ST25DV64K and ST25TVxxx, reading NDEF messages is not natively supported on Android™ devices embedding older Android™ version than version O.

#### 3.1 Capability container (CC)

The CC manages the information of a NFC Forum T5T.

The CC begins at the first byte of the memory area and contains four or eight bytes.

The CC is stored on contiguous bytes.

A four-byte CC limits the maximum data area size of a ST25DVxxx and ST25TVxxx devices to 2040 bytes (the number of blocks is coded on one byte), 504 bytes maximum for ST25DV04K.

An eight-byte CC allows a larger data area (the number of blocks is coded on two bytes).

The CC contains information (MLEN, encoded memory length) that represents the part of the memory that is allocated for storing the NDEF Message T5T\_Area. It is calculated from the T5T\_Area in byte as follows:

$$\text{MLEN} = \text{T5T\_Area} / 8$$

For products in the ST25DVxxx and ST25TVxxx devices, the CC can be coded either on four bytes or on eight bytes depending on the number of memory blocks that are dedicated to the message.

[Table 3. Four-byte capability container](#) details the structure of a four-byte CC to be used for ST25DV04K product.

**Table 3. Four-byte capability container**

Byte 0	Byte 1	Byte 2	Byte 3
Magic number	Version and access condition	MLEN	Additional feature information

[Table 4. Eight-byte capability container](#) details the structure of an eight-byte CC to be used for ST25DV16K, ST25DV64K and ST25TVxxx products.

**Table 4. Eight-byte capability container**

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Magic number	Version and access condition	00h	Additional feature information	RFU	RFU	MLEN	

#### 3.2 Magic number

The magic number allows the selection between one-byte address mode and two-byte address mode:

- The T5T sets the magic number (byte 0) in CC to E1h when one-byte address mode is supported
- The T5T sets the magic number in CC to E2h, if two-byte address mode is supported

ST25DVxxx and ST25TVxxx devices support one- or two-byte address for each block. Consequently, E1h or E2h can be used for memory sizes below 16 Kbits. For memories larger than or equal to 16 Kbits, E2h must be chosen.

*Note:* Two bytes CC file with magic number E2h is not natively supported by Android™ up to version Oreo included. iOS and Android™ after version Oreo natively support 2 bytes CC file with magic number E2h.

### 3.3 T5T\_Area and MLEN

T5T\_Area is part of the user memory space available to contain a NDEF message TLV. The NDEF message is stored inside the V-field of the NDEF Message TLV. The Terminator TLV is the last TLV structure in the T5T\_Area.

*Note:* Depending on the size of the NDEF message, some memory space can remain unused in the T5T\_area. The T5T\_Area size measured in bytes is equal to 8 x MLEN. The T5T\_Area directly follows the CC in the memory of the Type 5 Tag.

#### 3.3.1 MLEN encoding rule

MLEN must be equal to T5T\_Area size, in bytes, divided by 8, as defined in NFC Forum T5T specification. Achieving NFC Forum certification requires obeying to this MLEN encoding rule.

Example 1: MLEN encoding for a ST25DV04K (512 bytes memory size) and 4 bytes capability container (CC):

- If the entire user memoryfull user memory is used to store NDEF, T5T\_Area=512-4
  - $MLEN = (512 - 4) / 8 = 63$  (3Fh)
- If only part of user memory is used to store NDEF, T5T\_Area = 128 bytes (1 Kbits) for example
  - $MLEN = 128 / 8 = 16$  (10h)

Example 2: MLEN encoding for a ST25DV64K (8192 bytes memory size) and 8 bytes capability container (CC):

- If the entire user memoryfull user memory is used to store NDEF, T5T\_Area=8192-8
  - $MLEN = (8192 - 8) / 8 = 1023$  (03FFh)
- If only part of user memory is used to store NDEF, T5T\_Area = 512 bytes (4 Kbits) for example
  - $MLEN = 512 / 8 = 64$  (40h)

**Caution:** iOS and Android™ after version Oreo fully support MLEN encoded according to this rule. But for older version of Android™ (up to version Oreo included), a specific MLEN encoding rule must be used, as described in next chapter.

#### 3.3.2 MLEN specific encoding rule to support Android™ up to version Oreo included

For smartphones supporting Android™ up to version Oreo, MLEN must be equal to the entire user memory Full\_Tag\_Memory size, in bytes, divided by 8.

Example 1: MLEN encoding for a ST25DV04K (512 bytes memory size) and 4 bytes capability container (CC):

- $MLEN = 512 / 8 = 64$  (40h)

Example 2: MLEN encoding for a ST25DV64K (8192 bytes memory size) and 8 bytes capability container (CC):

- $MLEN = 8192 / 8 = 1024$  (400h)

*Note:* Smartphones supporting Android™ up to version Oreo also support MLEN encoded according to [Section 3.3.1 MLEN encoding rule](#), but in this case CC-byte3-bit2 must be set to 1b.

**Caution:** Encodings defined above are not compliant with NFC Forum certification, but are supported by iOS and any version of Android™.

### 3.4 Version and access condition (byte 1)

[Table 5. Version and access condition byte description](#) details the structure of the version and access condition byte:

**Table 5. Version and access condition byte description**

b7	b6	b5	b4	b3	b2	b1	b0
Major version		Minor version		Read Access		Write Access	
01b: Version 1.x		00b: Version y.0		00b: Always		00b: Always	
				01b: RFU		01b: RFU	
				10b: Proprietary		10b: Proprietary	
				11b: RFU		11b: Never	

For version 1.0 with all accesses granted, the byte value is 40h.

### 3.5 Additional feature information (byte 3)

**Table 6. Additional feature information byte description**

b7	b6	b5	b4	b3	b2	b1	b0
RFU		Special frame		Lock block	RFU <sup>(1)(2)</sup>		MBREAD

1. RFU bits must be set to 0.

2. Refer to [Section 3.3.2 MLEN specific encoding rule to support Android™ up to version Oreo included for CC-byte3-bit2 specific usage.](#)

CC byte 3 indicates which additional features are supported by the T5 tag.

Special frame:

- ST25DVxxx and ST25TVxxx support special frame.

Lock block:

- ST25DVxxx and ST25TVxxx support (Ext) Lock Block command, but only blocks 0 and 1 (CC) can be locked using this command. In order to write protect other blocks of the user memory, Areas protection mechanism should be used (see ST25DVxxx and ST25TVxxx datasheets for user memory description).
- It is recommended to set this bit to 0.

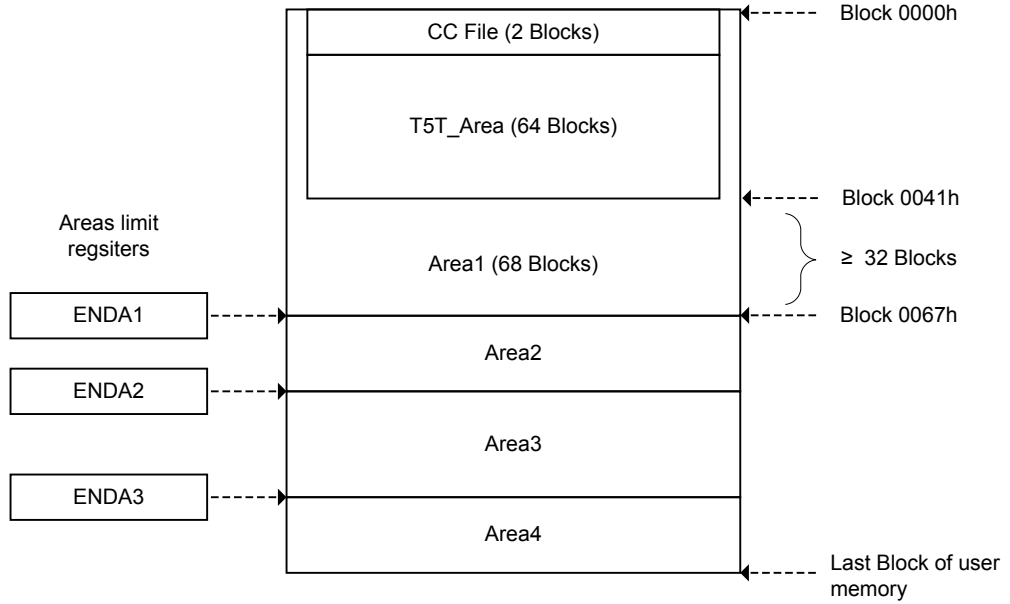
MBREAD:

- ST25DVxxx and ST25TVxxx support (Ext) Read Multiple Block command.

**Caution:** in ST25DVxxx and ST25TVxxx, a (Ext) Read Multiple Block command receives an error if crossing a user area border (see ST25DVxxx and ST25TVxxx datasheets). In Android™, if MBREAD is set to 1, Read Multiple Blocks of 32 blocks (128 bytes) are performed. Therefore, in order to avoid any read across user area borders if multiple block read is preferred (MBREAD=1), it is recommended to set user Area1 end, at least 32 blocks after the end of the T5T\_Area, or at the end of user memory (single user area).

Example of user Area 1 setting with MBREAD=1 and MLEN=0x20 (T5T\_Area=256 bytes) and CC File coded on 8 bytes:

- CC File size is 2 blocks (E2 40 00 01 00 00 00 20h)
- T5T\_Area size is (MLEN\*8)/4 = 64 blocks
  - T5T\_Area ends at block address 65 (41h)
- User Area1 must end at least at block 65+32=97 (61h)
  - ENDA1 = int(97/8) = 12 (0Ch). Last block of user Area1 is block address 103 (67h) (see ST25DVxxx and ST25TVxxx datasheets for Area end calculation description)

**Figure 1. ST25DVxxx/ST25TVxxx user memory**


## 4 Example of capability container for ST25DVxxx

This section provides two examples of programming values for the CC field of ST25DVxxx as a T5T. Refer to [Section 3.1 Capability container \(CC\)](#), [Section 3.2 Magic number](#), [Section 3.3 T5T\\_Area and MLEN](#), [Section 3.4 Version and access condition \(byte 1\)](#) and [Section 3.5 Additional feature information \(byte 3\)](#) for details.

### 4.1 ST25DV04K

Full user memory dedicated to store NDEF message:

- E1 40 40 00h to work properly with smartphone.
- E1 40 3F 00h for NFC Forum certification.

### 4.2 ST25DV64K and ST25TV64K

Part of the user memory (2 Kbits) dedicated to store NDEF message:

- E2 40 00 00 00 00 40 00h to work properly with smartphone
- E2 40 00 00 00 00 00 20h for NFC Forum certification

Full user memory dedicated to store NDEF message, with multiple read enabled (assuming user Area1 is set to full memory size):

- E2 40 00 01 00 00 40 00h to work properly with smartphone
- E2 40 00 01 00 00 03 FFh for NFC Forum certification



## 5 NDEF : NFC data exchange

This section illustrates NDEF with ST25DVxxx devices.

### 5.1 Reference

Refer to the dedicated NFC Forum specification (List available in [Section A.1 Reference NFC Forum documents](#)).

### 5.2 Dedicated PC applicative SW for CR95HF

Refer to the *CR95HF PC SW Manual: UM1084 user manual* available at [www.st.com](http://www.st.com).

### 5.3 PC SW application example with ST25DV04K

This section demonstrates the storage of an NDEF message when the user memory is fully dedicated to NDEF.

#### 5.3.1 CC content

**Figure 2. CC content example screen**

*NFC Type 5 - NDEF Message user interface*

READ WRITE CC FILE
READ NDEF MESSAGE
PREPARE NDEF MESSAGE
WRITE NDEF MESSAGE

**BLOCK 0**

Byte0	Byte1	Byte2	Byte3
E1	40	40	00

use 8 bytes Capacity Container

Additional informations

bit4: Special Frame format  
 bit3: support Lock Block commands  
 bit0: support Multiple Block commands

Magic Number

E1  
 E2 (Extended cmd supported)

MLEN \ 8

512bit = 0x08  
 1Kbit = 0x10  
 2Kbit = 0x20  
 4Kbit = 0x40  
 Size >= 16kbit : 0x00 and use 8 bytes Capacity Container

Additional informations

bit4: Special Frame format  
 bit3: support Lock Block commands  
 bit0: support Multiple Block commands

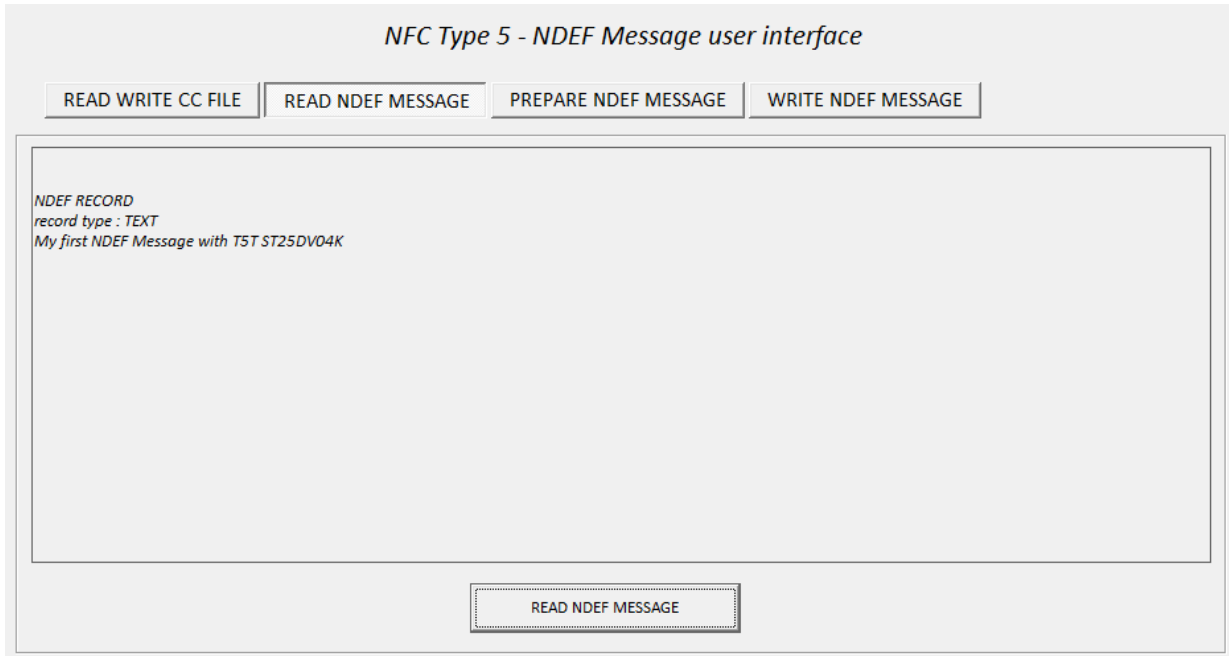
READ CC FILE

WRITE CC FILE \*

\* Write CCfile will modify your tag CC file

### 5.3.2 NDEF message

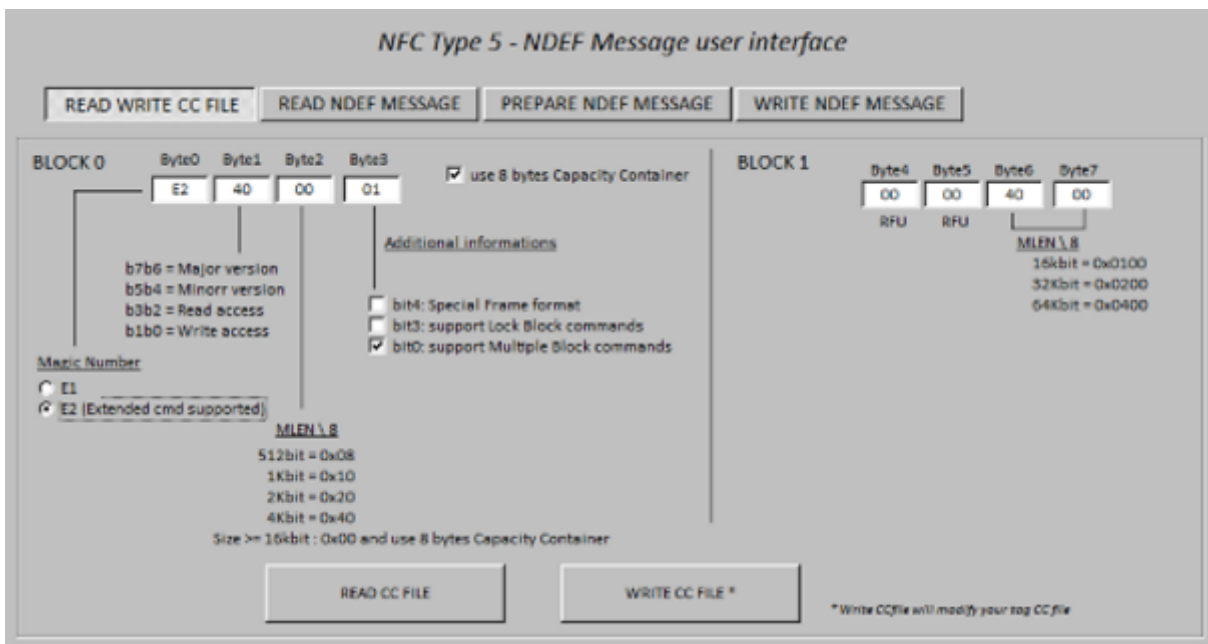
Figure 3. NDEF message example screen



### 5.3.3 User memory content (Read single block)

- Block 00h: Capability Container
- Blocks 01h to 0Dh: NDEF message

Figure 4. User memory content example screen



## 5.4 PC SW application example with ST25DV64K AND ST25TV64K

This section demonstrates the storage of an NDEF message when 18 Kbits in the user memory are dedicated to the NDEF message with multiple read enabled (assuming user Area1 is set to full memory size) and to be compatible with a smartphone.

### 5.4.1 CC content

Figure 2. CC content example screen

*NFC Type 5 - NDEF Message user interface*

READ WRITE CC FILE
READ NDEF MESSAGE
PREPARE NDEF MESSAGE
WRITE NDEF MESSAGE

**BLOCK 0**

Byte0	Byte1	Byte2	Byte3
E1	40	40	00

b7b6 = Major version  
 b5b4 = Minorr version  
 b3b2 = Read access  
 b1b0 = Write access

**Magic Number**

E1  
 E2 (Extended cmd supported)

**MLEN \ 8**

512bit = 0x08  
 1Kbit = 0x10  
 2Kbit = 0x20  
 4Kbit = 0x40  
 Size >= 16kbit : 0x00 and use 8 bytes Capacity Container

use 8 bytes Capacity Container

Additional informations

bit4: Special Frame format  
 bit3: support Lock Block commands  
 bit0: support Multiple Block commands

READ CC FILE

WRITE CC FILE \*

\* Write CCfile will modify your tag CC file

### 5.4.2 NDEF message

Figure 3. NDEF message example screen

*NFC Type 5 - NDEF Message user interface*

READ WRITE CC FILE
READ NDEF MESSAGE
PREPARE NDEF MESSAGE
WRITE NDEF MESSAGE

*NDEF RECORD*

*record type : TEXT*

*My first NDEF Message with T5T ST25DV04K*

READ NDEF MESSAGE

### 5.4.3 User memory content (Read single block)

- Block 00h: Capability Container
- Blocks 01h to 0Dh: NDEF message

Figure 5. User memory content example screen

The screenshot displays the 'EEPROM' tab of the application. The interface is divided into several functional areas:

- Control Panel:** Contains buttons for 'Extended Read Single Block', 'Extended Write Single Block', 'Extended Get Multiple Blocks Security Status', 'Extended Lock Block', and 'send EOF'. It also includes input fields for 'Address Block' (from 0000, Blocks Number 1) and 'RF block' (from 000F, Blocks Number 4).
- Memory Content Table:** A table showing memory addresses and their corresponding data:
 

0000	E2 40 00 01
0001	00 00 40 00
0002	03 2F D1 01
0003	2B 54 02 65
0004	6E 4D 79 20
0005	66 69 72 73
0006	74 20 4E 44
0007	45 46 20 4D
0008	65 73 73 61
0009	67 65 20 77
000A	69 74 68 20
000B	54 35 54 20
000C	53 54 32 35
000D	44 56 36 34
000E	4B 7E 00 00
000F	00 00 00 00
- Request/Answer Section:** Shows a request of '02300F00' and an answer of '800800000000077CF00' with 'Data = 00 00 00 00'.
- Log Window:** Displays communication logs:
 

```

            >>> CR95HFDLL_SENDRECEIVE, 02300D00
            <<< 80080044563634C36800
            EXTENDED READ SINGLE BLOCK
            >>> CR95HFDLL_SENDRECEIVE, 02300E00
            <<< 80080048FE0000FA8500
            EXTENDED READ SINGLE BLOCK
            >>> CR95HFDLL_SENDRECEIVE, 02300F00
            <<< 800800000000077CF00
            
```
- Tag Information:** Shows 'Detected Tag: TRANSPONDER UID' and options for 'Select Flag', 'Address Flag', and 'Option flag'.
- Data Rate:** Includes radio buttons for 'Low', 'High', '10%', and '100%', and a checkbox for 'Fast'.
- RF Controls:** Features 'Launch RF POR', 'RF OFF', and 'RF ON' buttons.
- Log Management:** Includes 'Save Log' and 'Clear Log' buttons.

## 6 ST25DVxxx as Type 5 Tag : NFC state transition

An NFC Forum device can detect ST25DVxxx as T5T in different states. The state is encoded in the byte 1 of the CC.

An ST25DVxxx acting as NFC Forum T5T platform can be issued in any valid state.

As a result, an ST25DVxxx T5T platform can be issued in the INITIALIZED state, the READ/WRITE state or even in the READ-ONLY state with a predefined NDEF message stored in it.

1. In the INITIALIZED State, the CC and the T5T\_Area are accessible for reading and writing data
  - The CC area is encoded properly with the bits b3 to b0 of byte 1 equal to 0000b (read/write access granted)
  - The T5T\_Area contains an NDEF Message TLV The L-field of NDEF Message TLV is equal to 00h.
2. In the READ/WRITE State, the CC and the T5T\_Area are accessible for reading and writing data.
  - The CC area is encoded properly with b3 to b0 of byte 1 equal to 0000b (read/write access granted)
  - The T5T\_Area contains an NDEF Message TLV The L-field of NDEF Message TLV is different than zero.
3. In the READ/ONLY state, the CC and the T5T\_Area are set to read-only
  - The CC area is encoded properly with b3 to b0 of byte 1 equal to 0001b, 0010b or 0011b (only read access granted)
  - The T5T\_Area contains an NDEF Message TLV The L-field of NDEF Message TLV is different than zero.
  - All blocks of the T5T\_Area shall be locked

When ST25DVxxx devices act as a Type 5 Tag , area 1 is dedicated to T5T\_Area,

The first two blocks are reserved for CC usage and can be locked individually if requested, using ISO 15693 RF Lock Block or Extended Lock Block command.

Area 1 of ST25DVxxx devices can be locked globally setting the corresponding RF access protection modifying RFA1SS register value located in system area (Refer to [Section A.2 Data Protection Management : RFA1SS](#)).

*Note: Area 1 is always readable so that access to CC content is always possible. Refer to ST25DVxxx datasheet "Dynamic NFC/RFID tag IC with 4-Kbit, 16-Kbit or 64-Kbit EEPROM and Fast Transfer Mode capability" or to the ST25TVxxx datasheet "64 Kbit EEPROM tag IC with passwords based on ISO/IEC 15693 and NFC type 5 tag", for more details on data protection management.*

## A Appendix

### A.1 Reference NFC Forum documents

- **Type 5 Tag Operation Specification**  
Type 5 Tag defines how an NFC device can interact with an NFC Forum Type 5 Tag, that is, an ISO/IEC 15693 tag that is configured to be able to store an NDEF message. As with other NFC Forum Tag operation specifications, the specification defines how to interact with, not build, a tag. The Type 5 Tag operation specification is focused on supporting poll-side, short-range communication with ISO/IEC 15693 tags containing an NDEF message.
- **Data Exchange Specifications**
- **NFC Data Exchange Format (NDEF) Technical Specification**  
Specifies a common data format for NFC Forum-compliant devices and NFC Forum-compliant tags.
- **Record Type Definition Technical Specifications**  
Technical specifications for Record Type Definitions (RTDs) and four specific RTDs: Text, URI, Smart Poster, and Generic Control.
- **NFC Record Type Definition (RTD) Technical Specification**  
Specifies the format and rules for building standard record types used by NFC Forum application definitions and third parties that are based on the NDEF data format. The RTD specification provides a way to efficiently define record formats for new applications and gives users the opportunity to create their own applications based on NFC Forum specifications.
- **Text RTD Technical Specification**  
Provides an efficient way to store text strings in multiple languages by using the RTD mechanism and NDEF format. An example of using this specification is included in the Smart Poster RTD.
- **URI RTD Technical Specification**  
Provides an efficient way to store Uniform Resource Identifiers (URI) by using the RTD mechanism and NDEF format. An example of using this specification is included in the Smart Poster RTD.
- **Verb RTD Technical Specification**  
Used to encode generic and carrier-specific supported services, which can then be used by implementations of the Connection Handover 1.4 candidate specification to offer an enhanced user experience. The Verb Record can, for example, encode the service to trigger the printing of a document or picture that will be transferred via the Bluetooth or WLAN connection.
- **Smart Poster RTD Technical Specification**  
Defines an NFC Forum Well Known Type to put URLs, SMSs or phone numbers on an NFC tag, or to transport them between devices. The Smart Poster RTD builds on the RTD mechanism and NDEF format and uses the URI RTD and Text RTD as building blocks.
- **Signature Record Type Definition Technical Specification**
- **Signature RTD Certificate Policy**  
Defines the procedural and operational requirements that the NFC Forum expects Certificate Authorities (CAs) to adhere to, when issuing and managing certificates to create signatures for NDEF messages. Provides users with the possibility of verifying the authenticity and integrity of data within the NDEF message, and specifies the format used when signing single or multiple NDEF records.
- **NFC Device Information RTD Technical Specification**  
Defines the Device Information record type, which conveys fundamental model and identity identification information. The purpose of the Device Information record is to convey host information in a record format that can be used across different carrier types or service types.
- **Reference Application Technical Specifications**
- **Connection Handover Technical Specification**
- **Personal Health Device Communication Technical Specification**

### A.2 Data Protection Management : RFA1SS

Table 7. RFA1SS register describes the structure and the programming of the RFA1SS register.

**Table 7. RFA1SS register**

RF	Command	Read configuration (cmd code A0h) @04h Write configuration (cmd code A1h) @04h	
	Type	R always, W if RF configuration security session is open and configuration not locked	
I <sup>2</sup> C	Address	E2 = 1, 0004h	
	Type	R always, W if I <sup>2</sup> C security session is open	
Field	Name	Function	Factory value
b1-b0	PWD_CTRL_A1	00: Area 1 RF user security session cannot be opened by password 01: Area 1 RF user security session opened by RF_PWD_1 10: Area 1 RF user security session opened by RF_PWD_2 11: Area 1 RF user security session opened by RF_PWD_3	00b
b3-b2	RW_PROTECTION_A1	00: Area 1 RF access: Read always allowed / Write always allowed 01: Area 1 RF access: Read always allowed / Write allowed if RF user security session is open 10: Area 1 RF access: Read always allowed / Write allowed if RF user security session is open 11: Area 1 RF access: Read always allowed / Write always forbidden	00b
b7-b4	RFU	-	0000b

## Revision history

**Table 8. Document revision history**

Date	Revision	Changes
01-Mar-2017	1	Initial release.
13-Dec-2017	2	<p>Updated:</p> <ul style="list-style-type: none"> <li>• Introduction in cover page</li> <li>• Section 3.1: Capability Container</li> <li>• Section 3.3: T5T_Area &amp; MLEN</li> <li>• Section 3.5: Additional feature information (byte 3)</li> <li>• Section 4.1: ST25DV04K</li> <li>• Section 4.2: ST25DV64K and ST25TV64K</li> <li>• Figure 1: CC content example screen,</li> <li>• Figure 2: NDEF message example screen</li> <li>• Figure 3: User memory content example screen</li> <li>• Figure 4: CC content example screen,</li> <li>• Figure 5: NDEF message example screen</li> <li>• Figure 6: User memory content example screen.</li> <li>• Section 6: ST25DVxxx as Type 5 Tag : NFC State Transition</li> </ul> <p>Added:</p> <ul style="list-style-type: none"> <li>• in all document reference to ST25TV64K</li> <li>• note in Section 3.2: Magic number</li> </ul>
01-Feb-2018	3	<p>Updated:</p> <ul style="list-style-type: none"> <li>• Section 3.3: T5T_Area &amp; MLEN</li> <li>• Section 4.2: ST25DV64K and ST25TV64K</li> </ul>
31-Jul-2018	4	<p>Updated:</p> <ul style="list-style-type: none"> <li>• <a href="#">Note in Section 3.2 Magic number</a></li> <li>• <a href="#">Section 3.5 Additional feature information (byte 3)</a></li> <li>• <a href="#">Section 4.1 ST25DV04K</a></li> <li>• <a href="#">Section 4.2 ST25DV64K and ST25TV64K</a></li> <li>• <a href="#">Figure 4. User memory content example screen</a></li> <li>• <a href="#">Figure 5. User memory content example screen</a></li> </ul> <p>Added:</p> <ul style="list-style-type: none"> <li>• RPN ST25TV16K</li> <li>• <a href="#">Section 3.3.1 MLEN encoding rule</a></li> <li>• <a href="#">Section 3.3.2 MLEN specific encoding rule to support Android™ up to version Oreo included</a></li> <li>• <a href="#">Figure 1. ST25DVxxx/ST25TVxxx user memory</a></li> </ul>



## Contents

<b>1</b>	<b>Acronyms and notational conventions</b>	<b>2</b>
1.1	Binary number representation	2
1.2	Hexadecimal number representation	2
1.3	Decimal number representation	2
<b>2</b>	<b>NFC Forum</b>	<b>3</b>
<b>3</b>	<b>ST25DVxxx Series and ST25TV64K configuration to support a NDEF message</b>	<b>4</b>
3.1	Capability Container	4
3.2	Magic number	4
3.3	T5T_Area & MLEN	4
3.3.1	MLEN encoding rule	5
3.3.2	MLEN specific encoding rule to support Android™ up to version Oreo included	5
3.4	Version & access condition (byte 1)	5
3.5	Additional feature information (byte 3)	6
<b>4</b>	<b>Example of capability container for ST25DVxxx</b>	<b>8</b>
4.1	ST25DV04K	8
4.2	ST25DV64K and ST25TV64K	8
<b>5</b>	<b>NDEF : NFC Data Exchange</b>	<b>9</b>
5.1	Reference	9
5.2	Dedicated PC applicative SW for CR95HF	9
5.3	PC SW application example with ST25DV04K	9
5.3.1	CC content	9
5.3.2	NDEF message	9
5.3.3	User memory content (Read single block)	10
5.4	PC SW application example with ST25DV64K AND ST25TV64K	10
5.4.1	CC content	11
5.4.2	NDEF message	11
5.4.3	User memory content (Read single block)	11
<b>6</b>	<b>ST25DVxxx as Type 5 Tag : NFC State Transition</b>	<b>13</b>
<b>A</b>	<b>Appendix</b>	<b>14</b>

<b>A.1</b>	Reference NFC forum documents .....	14
<b>A.2</b>	Data Protection Management : RFA1SS .....	14
<b>Revision history</b>	.....	<b>16</b>
<b>Contents</b>	.....	<b>17</b>
<b>List of tables</b>	.....	<b>19</b>
<b>List of figures.</b>	.....	<b>20</b>

## List of tables

<b>Table 1.</b>	Applicable products . . . . .	1
<b>Table 2.</b>	List of acronyms . . . . .	2
<b>Table 3.</b>	Four-byte capability container . . . . .	4
<b>Table 4.</b>	Eight-byte capability container . . . . .	4
<b>Table 5.</b>	Version and access condition byte description . . . . .	5
<b>Table 6.</b>	Additional feature information byte description . . . . .	6
<b>Table 7.</b>	RFA1SS register . . . . .	15
<b>Table 8.</b>	Document revision history . . . . .	16

## List of figures

Figure 1.	ST25DVxxx/ST25TVxxx user memory . . . . .	7
Figure 2.	CC content example screen . . . . .	9
Figure 3.	NDEF message example screen . . . . .	10
Figure 4.	User memory content example screen . . . . .	10
Figure 2.	CC content example screen . . . . .	11
Figure 3.	NDEF message example screen . . . . .	11
Figure 5.	User memory content example screen . . . . .	12

**IMPORTANT NOTICE – PLEASE READ CAREFULLY**

STMicroelectronics NV and its subsidiaries (“ST”) reserve the right to make changes, corrections, enhancements, modifications, and improvements to ST products and/or to this document at any time without notice. Purchasers should obtain the latest relevant information on ST products before placing orders. ST products are sold pursuant to ST’s terms and conditions of sale in place at the time of order acknowledgement.

Purchasers are solely responsible for the choice, selection, and use of ST products and ST assumes no liability for application assistance or the design of Purchasers’ products.

No license, express or implied, to any intellectual property right is granted by ST herein.

Resale of ST products with provisions different from the information set forth herein shall void any warranty granted by ST for such product.

ST and the ST logo are trademarks of ST. All other product or service names are the property of their respective owners.

Information in this document supersedes and replaces information previously supplied in any prior versions of this document.

© 2018 STMicroelectronics – All rights reserved