Introduction

ST25TVxxx products are EEPROM devices designed to be accessed via a standard contactless ISO 15693 RFID interface. ST25TVxxx devices can act as an NFC Type 5 Tag supporting NDEF messages. This document shows how to configure an ST25TVxxx device so that it can be detected as an NFC tag. It also lists the commands and operations compliant with the NFC Forum environment. This application note applies to the products listed below.

Table 1. Applicable products

<table>
<thead>
<tr>
<th>Reference</th>
<th>Part number</th>
</tr>
</thead>
<tbody>
<tr>
<td>ST25TVxxx</td>
<td>ST25TV512</td>
</tr>
<tr>
<td></td>
<td>ST25TV02K</td>
</tr>
</tbody>
</table>
1 Acronyms and notational conventions

### Table 2. List of acronyms

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>CC</td>
<td>Capability container (see Section 3.1 Capability container for details)</td>
</tr>
<tr>
<td>EEPROM</td>
<td>Electrically-erasable programmable read-only memory</td>
</tr>
<tr>
<td>ISO</td>
<td>International organization for standardization</td>
</tr>
<tr>
<td>MLEN</td>
<td>Encoded memory length (see Section 3.3 T5T_Area and MLEN for details)</td>
</tr>
<tr>
<td>NDEF</td>
<td>NFC data exchange (see Section A Appendix for details)</td>
</tr>
<tr>
<td>R</td>
<td>Read</td>
</tr>
<tr>
<td>RF</td>
<td>Radio frequency</td>
</tr>
<tr>
<td>RFID</td>
<td>Radio frequency identification</td>
</tr>
<tr>
<td>RFU</td>
<td>Reserved for future use</td>
</tr>
<tr>
<td>R/W</td>
<td>Read / Write</td>
</tr>
<tr>
<td>T5T</td>
<td>Type 5 tag (see Section A Appendix for details)</td>
</tr>
<tr>
<td>T5T_Area</td>
<td>Type 5 tag area</td>
</tr>
<tr>
<td>W</td>
<td>Write</td>
</tr>
</tbody>
</table>

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
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
- Select
- SLPV_Req

This reduced command set allows an NFC device to manage NDEF (NFC data exchange) implemented in a ST25TVxxx series device.

Please, for details refer to the NFC Forum Type 5 Tag Technical Specification.
3 configuration to support a NDEF message

The products in the ST25TVxxx series 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 block.

3.1 Capability Container
The CC manages the information of an NFC Forum T5T. The CC begins at the first byte of the memory area and contains four bytes. The CC is stored on contiguous bytes. A four-byte CC limits the maximum data area size of a ST25TVxxx series product to 2040 bytes (the number of blocks is coded on one byte).

The CC contains information the 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:

\[ \text{MLEN} = \frac{\text{T5T}_\text{Area}}{8} \]

For products in the ST25TVxxx series, the CC is coded on four bytes.

Table 3. Four-byte capability container details the structure of the four-byte CC that must be used.

<table>
<thead>
<tr>
<th>Byte 0</th>
<th>Byte 1</th>
<th>Byte 2</th>
<th>Byte 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Magic number</td>
<td>Version and access condition</td>
<td>MLEN</td>
<td>Additional feature information</td>
</tr>
</tbody>
</table>

3.2 Magic number
The magic number allows to select between one-byte address mode and two-byte address mode. ST25TVxxx supports only one-byte address mode. The T5T shall set the magic number (byte 0) in CC to E1h.

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 shall be equal to T5T_Area size, in bytes, divided by 8, as defined in NFC Forum T5T specification.

Achieving NFC Forum certification requires to obey to this MLEN encoding rule.

Examples for a 256 bytes (2kbits) memory size and 4 bytes capability container (CC):

- If the entire user memory is used to store NDEF, T5T_Area=256-4
  - MLEN = (256 – 4) / 8 = 31 (1Fh)
- If only part of user memory is used to store NDEF, T5T_Area = 128 bytes (1kbits) for example
  - MLEN = 128 / 8 = 16 (10h)

Note: 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 shall 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 is equal to the entire user memory size, in bytes, divided by 8.

Examples for a 256 bytes (2kbits) memory size and 4 bytes capability container (CC):
- \( \text{MLEN} = 256 / 8 = 32 \) (20h)

*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 shall be set to 1b.

*Note:* Encodings defined Section 3.3.2 MLEN specific encoding rule to support Android™ up to version Oreo included are not compliant with NFC Forum certification, but are supported by iOS™ and any version of Android™.

3.4 **Version & access condition (byte 1)**

Table 4. Version and access condition byte description details the structure of the version and access condition byte:

<table>
<thead>
<tr>
<th>b7</th>
<th>b6</th>
<th>b5</th>
<th>b4</th>
<th>b3</th>
<th>b2</th>
<th>b1</th>
<th>b0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Major version</td>
<td>Minor version</td>
<td>Read Access</td>
<td>Write Access</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>01b: Version 1.x</td>
<td>00b: Version y.0</td>
<td>00b: Always</td>
<td>00b: Always</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>01b: RFU</td>
<td>01b: RFU</td>
<td>10b: Proprietary</td>
<td>10b: Proprietary</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11b: RFU</td>
<td>11b: RFU</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

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

3.5 **Additional feature information (byte 3)**

ST25TVxxx supports Special Frame and Multiple Byte read. All blocks can be locked. The additional feature information byte is usually set to 01h.

<table>
<thead>
<tr>
<th>b7</th>
<th>b6</th>
<th>b5</th>
<th>b4</th>
<th>b3</th>
<th>b2</th>
<th>b1</th>
<th>b0</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFU</td>
<td>Special frame</td>
<td>Lock block</td>
<td>RFU</td>
<td>MBREAD</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
4 Example of capability container for ST25TVxxx

This section provides two examples of programming values for the CC field of ST25TVxxx as a T5T. Refer to Section 3.1 Capability container, Section 3.2 Magic number, Section 3.3 T5T_Area and MLEN, Section 3.4 Version & access condition (byte 1) and Section 3.5 Additional feature information (byte 3) for details.

4.1 ST25TV02K
• E1 40 20 01h to work properly with smartphone
• E1 40 1F 01h for NFC Forum certification

4.2 ST25TV512
• E1 40 08 01h to work properly with smartphone
• E1 40 07 01h for NFC Forum certification
5 NDEF : NFC Data Exchange

This section illustrates NDEF with ST25TVxxx.

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 development software user manual (UM1084) available at www.st.com.

5.3 PC SW application example with ST25TVxxx
This section demonstrates the storage of an NDEF message when the user memory is fully dedicated to NDEF.

5.3.1 CC content

Figure 1. CC content example screen
5.3.2 NDEF message

Figure 2. NDEF message example screen

5.3.3 User memory content (Read single block)

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

Figure 3. User memory content example screen
ST25TVxxx as Type 5 Tag: NFC State Transition

An NFC Forum Device can detect ST25TVxxx as T5T in different states. The state is encoded in the byte 1 of the CC.

An ST25TVxxx acting as NFC Forum T5T platform can be issued in any valid state. As a result, an ST25TVxxx 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 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.
   - The CC and all blocks of the T5T_Area shall be locked

When ST25TVxxx acts as a Type 5 Tag, area 1 is dedicated to T5T_Area, the first block is reserved for CC usage and can be locked individually if requested, using ISO 15693 RF Lock Block.

Each block of Area 1 of ST25TVxxx can be locked individually with the Lock_Single_Block command.

Note: Area 0 of ST25TV512 and ST25TV02K is always readable so that the CC content is always accessible.
Appendix

A.1 Reference NFC Forum documents

- **Type 5 Tag Technical Specification**
  
  Type 5 Tag defines the behaviour of NFC Forum Type 5 Tag, which means a tag that supports 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
## Revision history

**Table 6. Document revision history**

<table>
<thead>
<tr>
<th>Date</th>
<th>Revision</th>
<th>Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>30-Oct-2017</td>
<td>1</td>
<td>Initial release.</td>
</tr>
<tr>
<td>12-Jun-2018</td>
<td>2</td>
<td>Updated:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Section 3.1 Capability container</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Section 3.3 T5T_Area and MLEN</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Section 5.2 Dedicated PC applicative SW for CR95HF</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Added:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Section 3.3.1 MLEN encoding rule</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Section 3.3.2 MLEN specific encoding rule to support Android™ up to version Oreo included</td>
</tr>
</tbody>
</table>
List of tables

Table 1. Applicable products .......................................................... 1
Table 2. List of acronyms ................................................................. 2
Table 3. Four-byte capability container ............................................ 4
Table 4. Version and access condition byte description ................. 5
Table 5. Additional feature information byte description ............... 5
Table 6. Document revision history ................................................ 12
List of figures

Figure 1. CC content example screen ........................................................... 7
Figure 2. NDEF message example screen..................................................... 8
Figure 3. User memory content example screen.......................................... 8