How to operate my-d™ devices in NFC Forum™ Type 2 Tag infrastructures

SLE 66R01P
SLE 66R01PN
SLE 66R16P
SLE 66R32P
SLE 66R01L

Application Note

2011-11-14

Chip Card & Security

Edition 2011-11-14

Published by
Infineon Technologies AG
81726 Munich, Germany

Legal Disclaimer

The information given in this document shall in no event be regarded as a guarantee of conditions or characteristics. With respect to any examples or hints given herein, any typical values stated herein and/or any information regarding the application of the device, Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind, including without limitation, warranties of non-infringement of intellectual property rights of any third party.

Information

For further information on technology, delivery terms and conditions and prices, please contact the nearest Infineon Technologies Office (**www.infineon.com**).

Warnings

Due to technical requirements, components may contain dangerous substances. For information on the types in question, please contact the nearest Infineon Technologies Office.

Infineon Technologies components may be used in life-support devices or systems only with the express written approval of Infineon Technologies, if a failure of such components can reasonably be expected to cause the failure of that life-support device or system or to affect the safety or effectiveness of that device or system. Life support devices or systems are intended to be implanted in the human body or to support and/or maintain and sustain and/or protect human life. If they fail, it is reasonable to assume that the health of the user or other persons may be endangered.

my-d™ Type 2 Tag Operation - Application Note
The information in this document is subject to change without notice.

Revision History: Current Version 2011-11-14

Previous Release: 2011-10-06

Page	Subjects (major changes since last revision)
All	Added information about SLE 66R01PN (my-d™ move NFC)

Table of Contents

List of Tables

1 Introduction

This document gives details on

- the available my-d™ devices offering NFC Forum™ Type 2 Tag functionality
- default settings of my-d™ devices at delivery
- state transitions
- command sets of my-d™ products

Furthermore, some examples are given describing how to use the NFC memory block within the my-d™ devices.

This application note shall be used together with:

- ISO/IEC 14443-3 Type A Standard
- NFC Forum™ Type 2 Tag Specification
- my-d™ NFC Data Book
- my-d™ move Data Book

Please contact Chip Card & Security security.chipcard.ics@infineon.com to request the appropriate Data Books.

1.1 my-d™ Variants

Following my-d™ variants are available supporting NFC Forum™ Type 2 Tag functionality.

NFC commands allow to address data space listed in column ‘NFC memory size’, whereas ‘NFC User Data size’ indicates the amount of memory available for user purpose. The difference is reserved for administrative data such as Internal bytes and Capability Container.

Table 1 Chip variants of my-d™

Device name	Type	NFC memory size [bytes]	NFC User Data size [bytes]
my-d™ move lean	SLE 66R01L	64	48
my-d™ move	SLE 66R01P	152	128
my-d™ move NFC[1)]	SLE 66R01PN	152	128
my-d™ NFC	SLE 66R16P	1024	1008
my-d™ NFC	SLE 66R32P	2048[2)]	2032

1) Delivered in NFC INITIALIZED state
2) Two NFC memory sectors of 1 kByte

1.2 NFC Forum™ Type 2 Tag Characteristics

1.2.1 Memory Structure

Table 2 illustrates a 1024 byte memory area allocated to a Type 2 Tag. These 1024 bytes are organized in 256 blocks of 4 bytes each.

Table 2 NFC Forum™ Type 2 Tag memory organization (based on memory size 1024 bytes)

	NFC Block address	Byte Number within a block				Description
		0	1	2	3	
Type 2 Tag address range	00_H	internal0	internal1	internal2	internal3	Internal
	01_H	internal4	internal5	internal6	internal7	Internal
	02_H	internal8	internal9	LOCK0	LOCK1	Internal / Lock
	03_H	CC0	CC1	CC2	CC3	Capability Container
	04_H	D0	D1	D2	D3	Data
	05_H	D4	D5	D6	D7	Data
	...	...				Data
	...	...				Data
	FE_H	D1000	D1001	D1002	D1003	Data
	FF_H	D1004	D1005	D1006	D1007	Data

The given address range (from 00_H to FF_H) represents a NFC Forum™ Type 2 Tag with 1 kByte of memory.

NFC block addresses 00_H to 03_H hold

- Internal Bytes: internal0 - internal9
  - these bytes are reserved for manufacturing use and shall not be used to store information data.
- Static Lock Bytes: LOCK0 - LOCK1
  - read-only locking bits (Note: does not imply physical locking!)
- Capability Containers: CC0 - CC3
  - manage the information about the Type 2 Tag

Block addresses higher than 03_H belong to the NFC Forum™ Type 2 Tag Data Area (D0 to D1007) and are available to the user. The Type 2 Tag memory structure required by an application can be arranged within this memory area.

The memory structure depends on the available memory on the Type 2 Tag platform.

1.2.1.1 Static Memory Structure

The Static Memory Structure may be applied to Type 2 Tag platforms with a physical memory size equal to 64 bytes or if the application requires small amounts of user memory (up to 48 bytes).

1.2.1.2 Dynamic Memory Structure

The Dynamic Memory Structure is used for Type 2 Tag platforms with memories bigger than 64 bytes. This memory layout contains optional configuration information (Lock Control TLV and Memory Control TLV) to identify reserved memory areas within the data area.

1.2.2 TLV Blocks

Encoding of data on a NFC Forum™ Type 2 Tag is realized with TLV blocks:

- T (tag field) identifies the type of TLV block
- L (length field) provides the size of the value field
- V (value field) indicates the value field

Table 3 lists defined TLV Blocks used to encode data. The NFC Forum™ device shall use these TLV Blocks to write data to the data area of the NFC Forum™ Type 2 Tag.

Table 3 TLV Blocks

TLV Block Name	Tag Field Value	Description
NULL TLV	00_H	used for padding memory areas
Lock Control TLV	01_H	defines details of the lock bits
Memory Control TLV	02_H	identifies reserved memory areas
NDEF Message TLV	03_H	contains an NDEF message
Proprietary TLV	FD_H	Proprietary Information
Terminator TLV	FE_H	Last TLV Block in the data area

The order of the TLV Blocks is specified as listed:

- if present: Lock Control TLV(s)
- if present: Memory Control TLV(s)
- NDEF Message TLV(s) and Proprietary TLV(s)
- Terminator TLV

Note: For more detailed information about the NFC Forum™ Type 2 Tag memory structures and TLV Blocks please refer to the NFC Forum™ Type 2 Tag Specification.

1.2.3 Life Cycle

A NFC Forum™ Type 2 Tag device may be in different states. The current state is identified by the content of the Type 2 Tag memory.

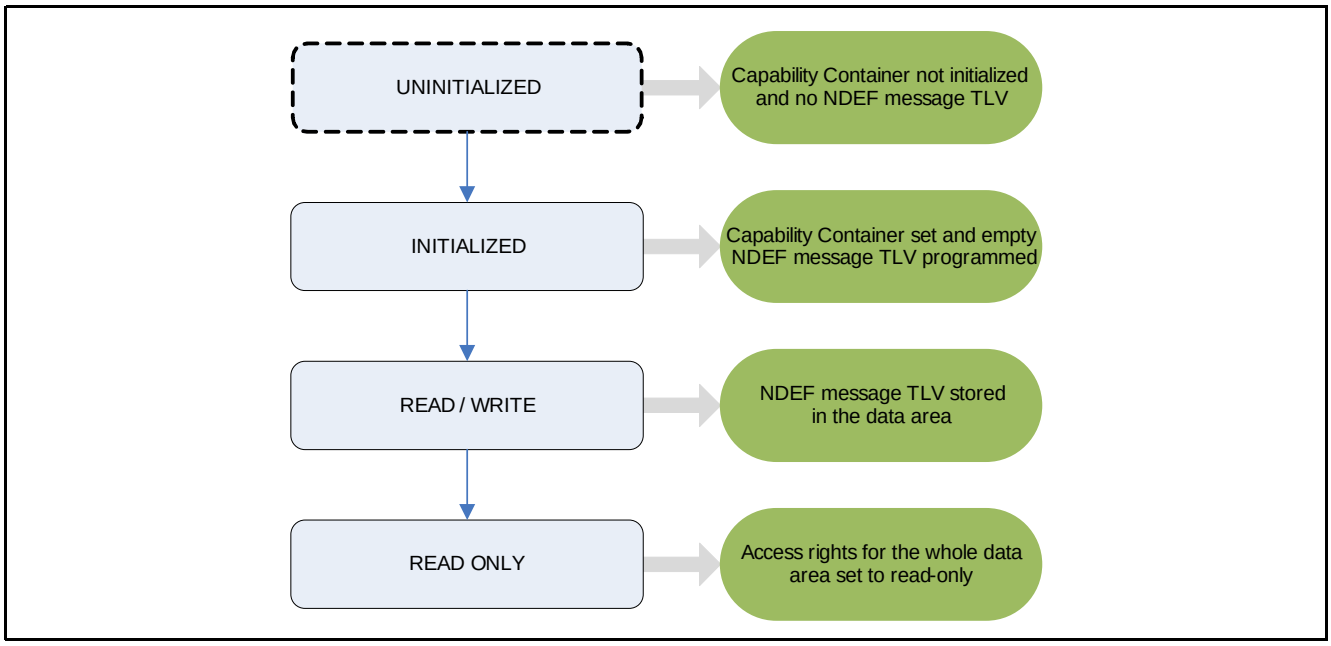

Figure 1 Type 2 Tag Life Cycle

In UNINITIALIZED state the memory is not prepared to be immediately used in a NFC Forum™ Type 2 Tag infrastructure. The chip supports the NFC Forum™ Type 2 Tag command set but the NFC Forum™ device may ignore the Type 2 Tag platform as it is not in a valid state.

The valid states within a NFC Forum™ Type 2 Tag infrastructure are

- INITIALIZED
  - the NFC Forum™ Type 2 Tag Capability Container bytes are set and allow read/write access (CC3 = 00_H)
  - the data area contains an NDEF message TLV with the length field set to 00_H (empty NDEF)
  - Terminator TLV
  - the NFC Forum™ device may then modify the data stored in the NFC Forum™ Type 2 Tag

Table 4 NFC Forum™ Type 2 Tag INITIALIZED state (based on memory size of 1024 bytes)

	NFC Block address	Byte Number within a block				Description
		0	1	2	3	
Type 2 Tag address range	00_H	internal0	internal1	internal2	internal3	Internal
	01_H	internal4	internal5	internal6	internal7	Internal
	02_H	internal8	internal9	LOCK0 = 00_H	LOCK1 = 00_H	Internal / Lock
	03_H	CC0 = $E1_H$	CC1 = 10_H	CC2 = $7E_H$	CC3 = 00_H	Capability Container
	04_H	03_H	00_H	FE_H	D3	Empty NDEF
	05_H	D4	D5	D6	D7	Data
	...	...				Data
	...	...				Data
	FE_H	D1000	D1001	D1002	D1003	Data
	FF_H	D1004	D1005	D1006	D1007	Data

- WRITE / READ
  - the NFC Forum™ Type 2 Tag Capability Container bytes are set and allow read/write access (CC3 = 00_H)
  - the data area contains an NDEF message TLV with the length value equal to the length of the NDEF message stored in the value field

Table 5 NFC Forum™ Type 2 Tag READ / WRITE state (based on memory size of 1024 bytes)

	NFC Block address	Byte Number within a block				Description
		0	1	2	3	
Type 2 Tag address range	00_H	internal0	internal1	internal2	internal3	Internal
	01_H	internal4	internal5	internal6	internal7	Internal
	02_H	internal8	internal9	LOCK0 = 00_H	LOCK1 = 00_H	Internal / Lock
	03_H	CC0 = $E1_H$	CC1 = 10_H	CC2 = $7E_H$	CC3 = 00_H	Capability Container
	04_H	03_H	$0B_H$	$D1_H$	01_H	NDEF message
	05_H	07_H	54_H	02_H	65_H	NDEF message
	06_H	$6E_H$	$4E_H$	44_H	45_H	NDEF message
	07_H	46_H	FE_H	D14	D15	NDEF message
	...	...				Data
	FE_H	D1000	D1001	D1002	D1003	Data
	FF_H	D1004	D1005	D1006	D1007	Data

- READ ONLY
  - the NFC Forum™ Type 2 Tag Capability Container bytes are set and allow read access only (CC3 = $0F_H$)
  - the data area contains an NDEF message TLV with the length value equal to the length of the NDEF message stored in the value field.

Table 6 NFC Forum™ Type 2 Tag READ ONLY state (based on memory size of 1024 bytes)

	NFC Block address	Byte Number within a block				Description
		0	1	2	3	
Type 2 Tag address range	00_H	internal0	internal1	internal2	internal3	Internal
	01_H	internal4	internal5	internal6	internal7	Internal
	02_H	internal8	internal9	LOCK0 = 00_H	LOCK1 = 00_H	Internal / Lock
	03_H	CC0 = $E1_H$	CC1 = 10_H	CC2 = $7E_H$	CC3 = $0F_H$	Capability Container
	04_H	03_H	$0B_H$	$D1_H$	01_H	NDEF message
	05_H	07_H	54_H	02_H	65_H	NDEF message
	06_H	$6E_H$	$4E_H$	44_H	45_H	NDEF message
	07_H	46_H	FE_H	D14	D15	NDEF message
	...	...				Data
	FE_H	D1000	D1001	D1002	D1003	Data
	FF_H	D1004	D1005	D1006	D1007	Data

Note: Please note that CC3 = $0F_H$ indicates a "soft-lock" only and does not physically lock the memory.

1.2.4 Command set

Table 7 lists the NFC Forum™ Type 2 Tag commands available with the my-d™ devices.

Table 7 NFC Forum™ Type 2 Tag command set

Command[1)]	Op-Code	Comment	Address range
Sector Select[2)]	$C2_H$	Sector selection	-
Read	30_H	Reads 16 bytes of data	00_H - FF_H
Write	$A2_H$	Writes 4 byte of data	00_H - FF_H

1) Please refer to the NFC Forum™ Type 2 Tag specification for detailed command descriptions.
2) For variants with more than 1 kByte of Type 2 Tag memory.

1.2.4.1 Sector Select Command

The Sector Select command supports the access to NFC Forum™ Type 2 Tag platforms with physical memories bigger than 1 kByte. The sector size is 1 kByte.

The Sector Select command is divided into two command packets.

- packet 1 contains the command code and a single byte parameter (set to FF_H). Valid responses are:
  - ACK for Type 2 Tag platforms with bigger memories than 1 kByte
  - NACK for Type 2 Tag platforms with smaller memories than 1 kByte

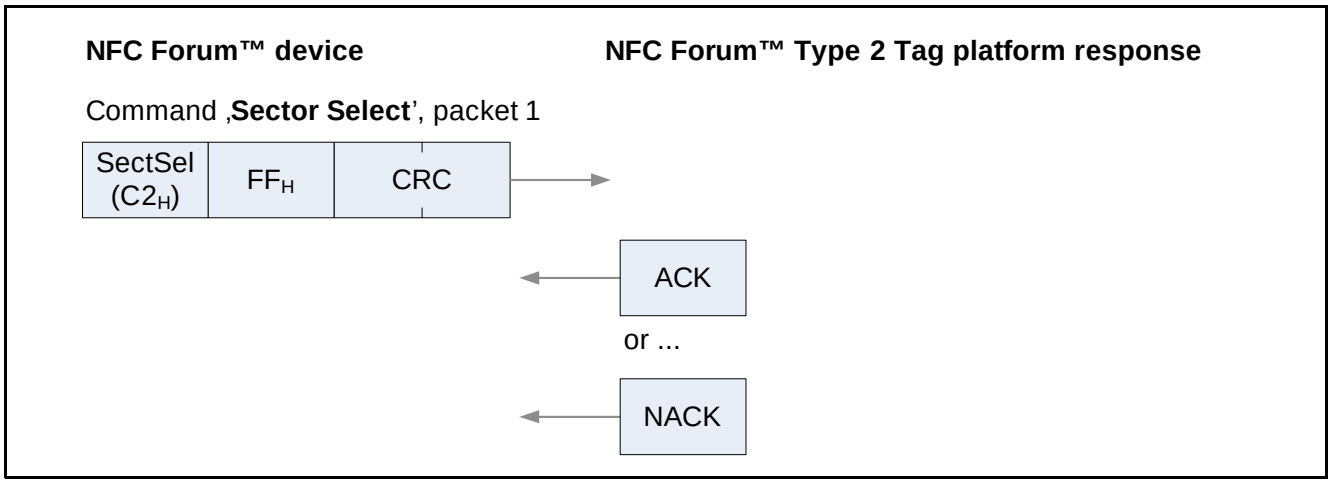

Figure 2 Sector Select Command, packet 1

Table 8 'Sector Select' command parameters (packet 1)

Name	Length[1)]	Remark
SectSel	1	command opcode: $C2_H$
FF_H	1	fixed value for packet 1
CRC	2	CRC

1) length of parameter field in [byte(s)]

Table 9 'Sector Select' command response parameters (packet 1)

Name	Length	Remark
ACK	1	Replied without CRC
NACK	1	Error on Sector Select command; replied without CRC

Packet 2 will be skipped if the response to packet 1 was NACK.

- packet 2 contains the sector number and three additional bytes which are reserved for future use and set to 00_H
  - passive ACK (= no response) if the sector number is valid
  - NACK if the addressable data memory space is exceeded

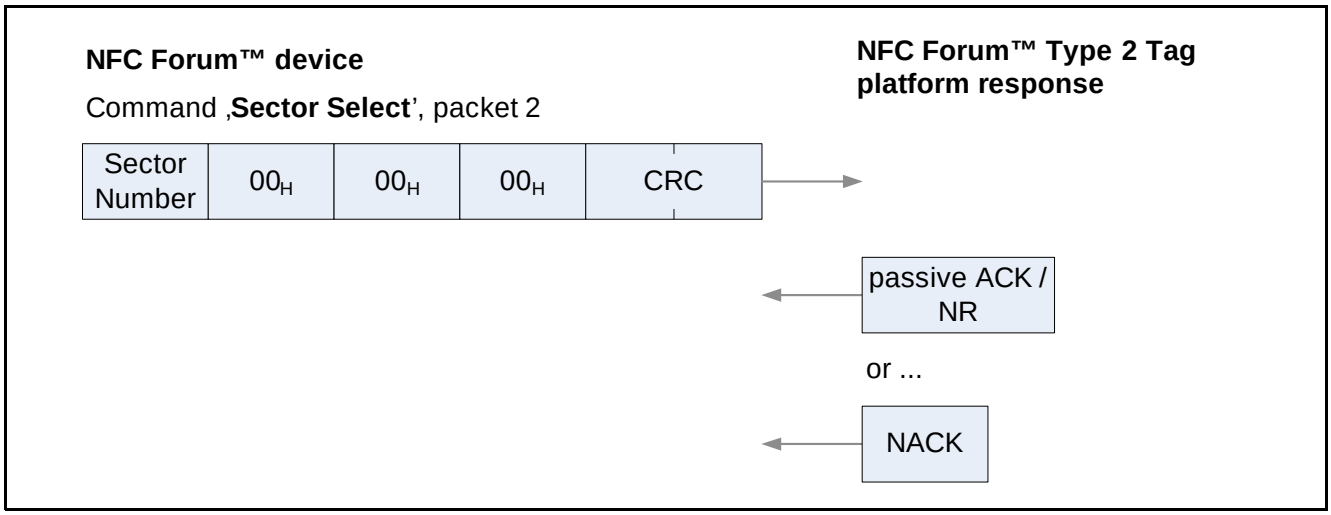

Figure 3 Sector Select Command, packet 2

Table 10 'Sector Select' command parameters (packet 2)

Name	Length[1)]	Remark
Sector Number	1	Valid range for Sector Number: 00_H to FE_H FF_H is RFU
00_H	3	RFU
CRC	2	CRC

1) length of parameter field in [byte(s)]

Table 11 'Sector Select' command response parameters (packet 2)

Name	Length	Remark
passive ACK - NR	0	no reply
NACK	1	Error on Sector Select command; replied without CRC

1.2.4.2 Read Command

The Read command reads 16 bytes data out of the memory starting from the specified address.

The valid address range is 00_H to FF_H and depends on the available memory. If an address outside the chip-specific address range is specified the my-d™ chip responds with a NACK.

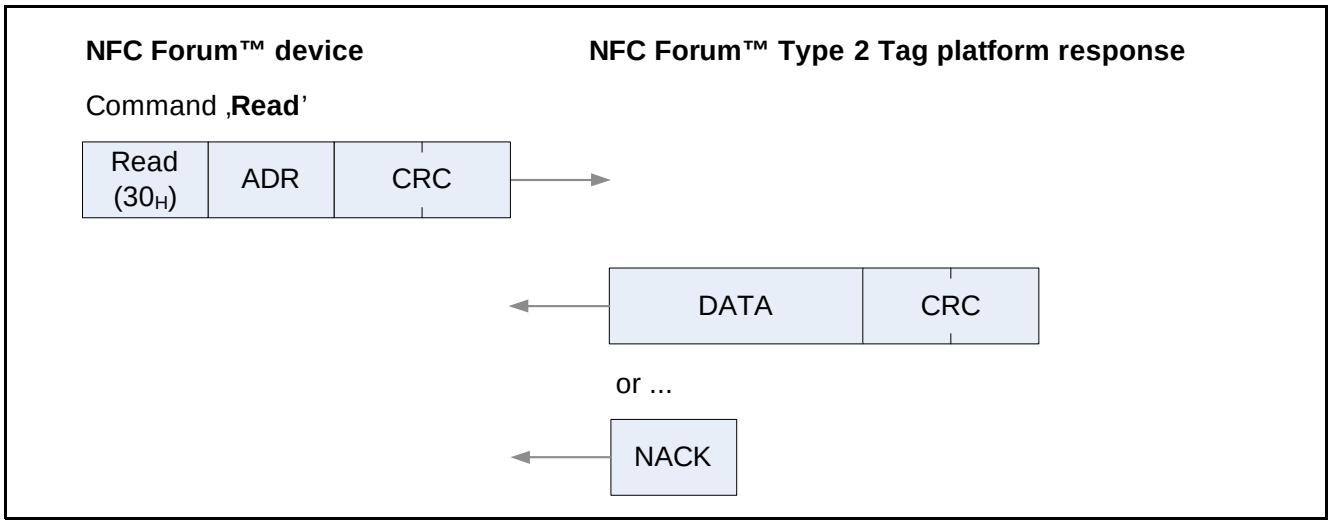

Figure 4 Read Command

Table 12 'Read' command parameters

Name	Length[1)]	Remark
Read	1	command opcode: 30_H
ADR	1	Block address to be read The valid address range is from 00_H to FF_H (depending on the available memory)
CRC	2	CRC

1) length of parameter field in [byte(s)]

Table 13 'Read' command response parameters

Name	Length	Remark
DATA	16	Content of four blocks
CRC	2	CRC
or		
NACK	1	Error on Read command; replied without CRC

1.2.4.3 Write Command

If the write access is granted the Write command is used to program 4 bytes of data to the specified address in the memory. This command should be used to program the OTP block and LOCKx bytes as well.

The valid address range is from 00_H to FF_H and depends on the available memory. If an address outside the chip-specific address range is specified the my-d™ chip responds with a NACK.

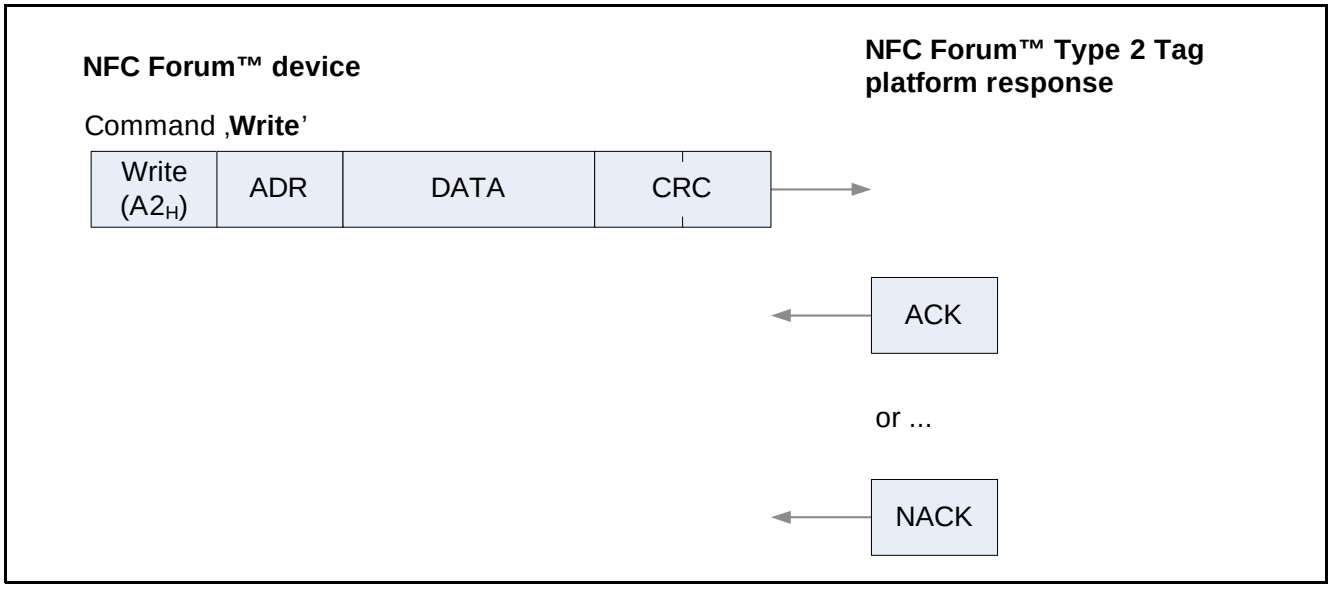

Figure 5 Write Command

Table 14 'Write' command parameters

Name	Length[1)]	Remark
Write	1	command opcode: $A2_H$
ADR	1	Block address to be written The valid address range is from 00_H to FF_H (depending on the available memory)
DATA	4	Data to be written
CRC	2	CRC

1) length of parameter field in [byte(s)]

Table 15 'Write' command response parameters

Name	Length	Remark
ACK	1	Write successful; replied without CRC
NACK	1	Error on Write command; replied without CRC

2 Setting up my-d™ to support NFC Forum™ Type 2 Tags

The following section describes the recommended procedure to set up a NFC Forum™ Type 2 Tag memory on a my-d™ device.

2.1 my-d™ Memory Configuration at Delivery

Depending on the application and order form my-d™ devices could be delivered in UNINITIALIZED or INITIALIZED state.

In UNINITIALIZED state the my-d™ devices are delivered with the memory set to 00_H. The NFC Forum™ Type 2 Tag platform needs to be initialized prior to be used in a NFC Forum™ Type 2 Tag infrastructure.

In INITIALIZED state the necessary memory content has already been programmed to the my-d™ devices and they can be used in NFC Forum™ Type 2 Tag infrastructures immediately.

Table 16 and **Table 20** show the default memory configuration of the my-d™ devices at delivery.

2.2 my-d™ move and my-d™ move lean

These chips come with

- a 7-byte UID including two BCC bytes programmed during manufacturing (uid0-uid6, BCC0 and BCC1); these bytes are internal bytes. This UID is used during the ISO/IEC 14443-3 Type A anti-collision.
- INT2 set to 00_H
- LOCK0 and LOCK1 set to 00_H[1)] (Lock bytes for Block 00_H - $0F_H$)
- LOCK2 to LOCK5 set to 00_H (my-d™ move only: Lock bytes for Block 10_H - 23_H)
- OTP memory in block 03_H set to 00_H (Capability Container)
- complete data area set to 00_H
- address range: my-d™ move lean: 00_H - $0F_H$; my-d™ move: 00_H - 25_H

Table 16 my-d™ move - SLE 66R01P in UNINITIALIZED state

	NFC Block address	Byte Number within a block				Description
		0	1	2	3	
Type 2 Tag address range	00_H	uid0	uid1	uid2	BCC0	Serial Number / BCC
	01_H	uid3	uid4	uid5	uid6	Serial Number
	02_H	BCC1	INT2 = 00_H	LOCK0 = 00_H	LOCK1 = 00_H	BCC1 / Internal / Lock
	03_H	00_H	00_H	00_H	00_H	OTP
	04_H	00_H	00_H	00_H	00_H	Data
	...	...				Data
	$0F_H$	...				Data
	10_H	...				Data
	...	...				Data
	23_H	00_H	00_H	00_H	00_H	Data
	24_H	LOCK2 = 00_H	LOCK3 = 00_H	LOCK4 = 00_H	LOCK5 = 00_H	Lock
	25_H	Manufacturing Data				Reserved

In order to initialize my-d™ move variants the Capability Container bytes (Block 03_H) and an empty NDEF message (Block 04_H) need to programmed.

1) 00_H indicates Block Read / Write access.

Table 17 defines settings of the Capability Container bytes to address the NFC memory.

Table 17 Capability Container settings for my-d™ move lean and my-d™ move (NFC)

Chip Type	CC0[1)]	CC1[2)]	CC2[3)]	CC3[4)]
SLE 66R01L	$E1_H$	10_H or 11_H	06_H	00_H
SLE 66R01P	$E1_H$	10_H or 11_H	10_H	00_H
SLE 66R01PN	$E1_H$	10_H or 11_H	10_H	00_H

1) Magic Number defines that NFC Forum™ defined data is stored in the data area
2) Version number of the supported specification
3) CC2 indicates the memory size of the data area of the Type 2 Tag; the given values represent the maximum values for the chips
4) Indicates read and write access capabilities of the data area

Table 18 defines the empty NDEF Message TLV (identified with the Tag field value of 03_H). The Length field value is set to 00_H; due to that the Value field is not present.

The Terminator TLV is the last TLV block in the data area.

Table 18 Empty NDEF message

NDEF Message TLV			Terminator TLV		
Tag field	**Length field**	**Value field**	**Tag Field**	**Length field**	**Value field**
03_H	00_H	-	FE_H	-	-

The programming of this information (CCx and empty NDEF) has to be done with the NFC Forum™ Type 2 Tag Write command. After that the my-d™ move is in INITIALIZED state.

Table 19 shows a my-d™ move device in INITIALIZED state (correlates with the SLE 66R01PN memory content at delivery).

Table 19 my-d™ move NFC - SLE 66R01PN in INITIALIZED state

	NFC Block address	Byte Number within a block				Description
		0	1	2	3	
Type 2 Tag address range	00_H	uid0	uid1	uid2	BCC0	Serial Number / BCC
	01_H	uid3	uid4	uid5	uid6	Serial Number
	02_H	BCC1	INT2 = 00_H	LOCK0 = 00_H	LOCK1 = 00_H	BCC1 / Internal / Lock
	03_H	$E1_H$	10_H	10_H	00_H	Capability Container
	04_H	03_H	00_H	FE_H	...	Empty NDEF
	05_H	...				Data
	...	...				Data
	23_H	...				Data
	24_H	LOCK2 = 00_H	LOCK3 = 00_H	LOCK4 = 00_H	LOCK5 = 00_H	Lock
	25_H	Manufacturing Data				Reserved

Note: Please note that the configuration of the different my-d™ move devices to INITIALIZED state is non-reversible; the Capability Containers (located in the OTP memory block 03_H) cannot be overwritten to use the my-d™ move as plain, standard device anymore.

2.3 my-d™ NFC

The my-d™ NFC chip in UNINITIALIZED state comes with the whole NFC memory set to 00_H; they need to be initialized before being used in an NFC Forum™ Type 2 Tag infrastructure.

Table 20 my-d™ NFC - SLE 66R16P in UNINITIALIZED state

	NFC Block address	Byte Number within a block				Description
		0	1	2	3	
Type 2 Tag address range	00_H	00	00	00	00	Data
	01_H	00	00	00	00	Data
	02_H	00	00	00	00	Data
	03_H	00	00	00	00	Data
	...	...				Data
	FE_H	00	00	00	00	Data
	FF_H	00	00	00	00	Data

In order to enable the INITIALIZED state at least the Capability Containers and the Empty NDEF have to be programmed to the memory.

If the application requires some internal information Block 00_H, Block 01_H and the first two bytes of Block 02_H may be used (internal0 - internal9).

- Block 02_H, byte 2 and 3: Static Lock bytes
- Block 03_H: Capability Containers
- Block 04_H: start of data area

Table 21 defines valid settings of the Capability Container bytes.

Table 21 Capability Container settings for my-d™ NFC

Chip Type	CC0	CC1	CC2[1)]	CC3
SLE 66R16P	$E1_H$	10_H or 11_H	$7E_H$	00_H
SLE 66R32P	$E1_H$	10_H or 11_H	FE_H	00_H

1) CC2 indicates the memory size of the data area of the Type 2 Tag; the given values represent the maximum values for the chips

Table 22 my-d™ NFC - SLE 66R16P in INITIALIZED state

	NFC Block address	Byte Number within a block				Description
		0	1	2	3	
Type 2 Tag address range	00_H	internal0	internal1	internal2	internal3	Internal
	01_H	internal4	internal5	internal6	internal7	Internal
	02_H	internal8	internal9	LOCK0 = 00_H	LOCK1 = 00_H	Internal / Lock
	03_H	$E1_H$	10_H	$7E_H$	00_H	Capability Container
	04_H	03_H	00_H	FE_H	00_H	Empty NDEF
	05_H	...				Data
	...	...				Data
	FE_H	...				Data
	FF_H	00_H	00_H	00_H	00_H	Data

3 NFC Forum™ Type 2 Tag Examples

Following examples show the Type 2 Tag memory on a my-d™ chip in two different memory structures (static and dynamic) as defined in the NFC Forum™ Type 2 Tag specification.

3.1 Static Memory Structure

The Static Memory Structure may be applied to Type 2 Tag platforms with a physical memory size equal to 64 bytes or if the application requires small amounts of user memory (up to 48 bytes).

3.1.1 Empty NDEF message in Static Memory Structure: State INITIALIZED

The example shows a my-d™ move chip in INITIALIZED state with a static memory structure, an empty NDEF message TLV and the Terminator TLV. The 48 bytes of NFC Forum™ Type 2 Tag data memory are located in the address range from 04_H to $0F_H$.

Table 23 Static Memory Structure - INITIALIZED State (based on my-d™ move)

NFC Block address (hex)	Byte Number within a block			
	0	**1**	**2**	**3**
00_H	uid0	uid1	uid2	INT0
01_H	uid3	uid4	uid5	uid6
02_H	INT1	INT2	LOCK0 = 00_H	LOCK1 = 00_H
03_H	CC0 = $E1_H$ [1]	CC1 = 10_H	CC2 = 06_H [2]	CC3 = 00_H
04_H	03_H [3]	00_H	FE_H [4]	...
05_H	...	...	...	...
...	...	...	...	...
...	...	...	...	...
$0F_H$	D44	D45	D46	D47
10_H	...	...	...	...
...	...	...	...	...
23_H	...	...	....	...
24_H	LOCK2 = 00_H	LOCK3 = 00_H	LOCK4 = 00_H	LOCK5 = 00_H
25_H	Manufacturing Data			

1) $E1_H$ = 'Magic Number'
2) NFC Data memory size is 48 bytes (highest block address $0F_H$). The whole memory may be used for an NFC application with the CC2 set to 10_H (highest block address 23_H).
3) NDEF message TLV (here: empty NDEF)
4) Terminator TLV

The Static Lock bytes LOCK0 and LOCK1 are set to 00_H indicating that the memory blocks are not locked.

The Capability Containers indicate:

- CC0 = $E1_H$: present NDEF data inside the tag
- CC1 = 10_H: support of Version 1.0 of the NFC Forum™ Type 2 Tag specification
- CC2 = 06_H: 48 bytes of memory for data
- CC3 = 00_H: read and write access is granted without any security

The data area starts at Block 04_H and holds two TLV blocks:

- NDEF message TLV
  - T: 03_H indicates the NDEF message
  - L: 00_H defines the length of the stored NDEF message (here: empty NDEF)
  - V: not present
- Terminator TLV

3.1.2 NDEF message in Static Memory Structure: State READ / WRITE, CC3 = 00_H

Based on the INITIALIZED NFC Forum™ Type 2 Tag configuration given in **Chapter 3.1.1** following example shows a NFC Forum™ Type 2 Tag in READ / WRITE state in a static memory structure, a NDEF message TLV and the Terminator TLV.

Table 24 Static Memory Structure - READ / WRITE State (based on my-d™ move)

NFC Block address (hex)	Byte Number within a block			
	0	**1**	**2**	**3**
00_H	uid0	uid1	uid2	INT0
01_H	uid3	uid4	uid5	uid6
02_H	INT1	INT2	LOCK0 = 00_H	LOCK1 = 00_H
03_H	CC0 = $E1_H$	CC1 = 10_H	CC2 = 06_H	CC3 = 00_H
04_H	03_H [1)]	13_H [2)]	$D1_H$	01_H
05_H	$0F_H$	54_H	02_H	65_H
06_H	$6E_H$	48_H	65_H	$6C_H$
07_H	$6C_H$	$6F_H$	20_H	77_H
08_H	$6F_H$	72_H	$6C_H$	64_H
09_H	21_H	FE_H	...	...
...	...	...	...	...
$0F_H$	D44	D45	D46	D47
10_H	...	...	...	...
...	...	...	...	...
23_H	...	...	...	...
24_H	LOCK2	LOCK3	LOCK4	LOCK5
25_H	Manufacturing Data			

1) NDEF Message TLV
2) Length of value field (NDEF Message TLV)

Settings for the Static Lock bytes LOCK0 and LOCK1 and Capability Containers remain unchanged.

The data area starts at Block 04_H and holds two TLV blocks:

- NDEF message TLV
  - T: 03_H indicates the NDEF message
  - L: 13_H defines the length of the stored NDEF message
  - V: NDEF record holding the “Hello world!” payload according to the NFC Forum™ Text RTD.

The data area is programmed with the NFC Forum™ Type 2 Tag Write command.

The Static Lock bytes LOCK0 and LOCK1 may be used to lock blocks within the data area.

3.1.3 NDEF message in Static Memory Structure: State READ-ONLY, CC3 = $0F_H$

Based on the example of the NFC Forum™ Type 2 Tag in READ / WRITE state (see **Chapter 3.1.2**) the NFC Forum™ Type 2 Tag platform may be set to READ-ONLY state.

The data area of the NFC Forum™ Type 2 Tag platform already contains an NDEF Message TLV and the length field of the NDEF Message TLV is different to 00_H. This data remains unchanged.

In order to set the NFC Forum™ Type 2 Tag platform to READ-ONLY state the Capability Container byte CC3 needs to be modified.

Table 25 Static Memory Structure - READ-ONLY State (based on my-d™ move)

NFC Block address (hex)	Byte Number within a block			
	0	**1**	**2**	**3**
00_H	uid0	uid1	uid2	INT0
01_H	uid3	uid4	uid5	uid6
02_H	INT1	INT2	LOCK0 = 00_H	LOCK1 = 00_H
03_H	CC0 = $E1_H$	CC1 = 10_H	CC2 = 06_H	CC3 = $0F_H$ [1)]
04_H	03_H	13_H	$D1_H$	01_H
05_H	$0F_H$	54_H	02_H	65_H
06_H	$6E_H$	48_H	65_H	$6C_H$
07_H	$6C_H$	$6F_H$	20_H	77_H
08_H	$6F_H$	72_H	$6C_H$	64_H
09_H	21_H	FE_H	...	...
...	...	...	...	...
$0F_H$	D44	D45	D46	D47
10_H	...	...	...	...
...	...	...	...	...
23_H	...	...	...	...
24_H	LOCK2	LOCK3	LOCK4	LOCK5
25_H	Manufacturing Data			

1) CC3 indicates the access capabilities of the data area

In this example the settings for the Static Lock bytes LOCK0 and LOCK1 remain unchanged.

An NFC Forum™ device shall identify the READ-ONLY state of the NFC Forum™ Type 2 Tag platform.

For CC3 the higher nibble (0_H) indicates that read access is granted without any security; the lower nibble (F_H) controls the write access condition. Here: no write access is granted. The READ-ONLY state indicates a soft-lock only; physical locking of appropriate memory is done by setting dedicated bits within the LOCK bytes.

3.2 Dynamic Memory Structure

The Dynamic Memory Structure is defined for tags with memory sizes bigger than 64 bytes.

3.2.1 Empty NDEF message in Dynamic Memory Structure: State INITIALIZED

The example shows an NFC Forum™ Type 2 Tag in INITIALIZED state with a dynamic memory structure with 32 data area block. It holds one Lock Control TLV, one Memory Control TLV, an empty NDEF Message TLV and the Terminator TLV.

Table 26 Dynamic Memory Structure - INITIALIZED State (based on SLE 66R16P)

NFC Block address (hex)	Byte Number within a block			
	0	**1**	**2**	**3**
00_H	internal0	internal1	internal2	internal3
01_H	internal4	internal5	internal6	internal7
02_H	internal8	internal9	LOCK0 = 00_H	LOCK1 = 00_H
03_H	CC0 = $E1_H$	CC1 = 10_H	CC2 = $7E_H$	CC3 = 00_H
04_H	01_H [1)]	03_H	$D0_H$	20_H
05_H	24_H	02_H [2)]	03_H	$E0_H$
06_H	$2C_H$	04_H	03_H [3)]	00_H
07_H	FE_H [4)]	...	...	...
08_H	...			
...	...			
34_H	DynLock2 = 00_H	DynLock3 = 00_H	DynLock4 = 00_H	DynLock5 = 00_H
35_H	Reserved	Reserved	Reserved	Reserved
...	Reserved	Reserved	Reserved	Reserved
$3F_H$	Reserved	Reserved	Reserved	Reserved
40_H	...			
...	...			
FF_H	...			

1) Start of Lock Control TLV
2) Start of Memory Control TLV
3) NDEF message TLV (here: Empty NDEF)
4) Terminator TLV

Internal bytes (internal0 - internal9) are reserved to hold application specific internal data like a serial number or manufacturing data. The NFC Forum™ device shall not use it to store information data.

The Static Lock bytes LOCK0 and LOCK1 are set to 00_H indicating that the memory blocks are not locked.

Dynamic Lock bytes (DynLock2 - DynLock5) are located in the data area.

The Capability Containers indicate:

- CC0 = $E1_H$: present NDEF data inside the tag
- CC1 = 10_H: support of Version 1.0 of the NFC Forum™ Type 2 Tag specification
- CC2 = $7E_H$: 1008 bytes of memory for data (here: SLE 66R16P, Sector 0)
- CC3 = 00_H: read and write access is granted without any security

The data area starts at Block 04_H and holds following four TLV blocks.

3.2.1.1 Lock Control TLV

A Lock Control TLV provides details and control information about the lock area(s) like the location of the dynamic lock bytes, the number of dynamic lock bits.

- **T: 01_H** indicates the Lock Control TLV
- **L: 03_H** defines the length of the value field
- **V:** indicates the position and size of the lock area
  - **$D0_H$:** defines the position of the dynamic lock bytes inside the tag

    The higher nibble codes the 'PageAddr' by the number of pages (D_H).
    The lower nibble codes the 'ByteOffset' within the page (0_H).

  - **20_H:** coded as the number of bits. It indicates the size of the dynamic lock area in bits (here: 20_H to lock 32 data area blocks)

  - **24_H:** page control information defines the size of a page and the number of bytes each dynamic lock bit is able to lock.

    The higher nibble codes the number of locked bytes: $2^{BytesLockedPerLockBit} = 2^2 = 4$
    The lower nibble codes the size of a page: $2^{BytesPerPage} = 2^4 = 16$

Using the settings within the Lock Control TLV an NFC Forum™ device may calculate the byte address of the dynamic lock area:

(1)

$$ByteAddr = PageAddr \times 2^{BytesPerPage} + ByteOffset$$

$$ByteAddr = D_H \times 2^4 + 0$$

$$ByteAddr = 13 \times 16 = 208 (BlockAddr = 34_H)$$

So the dynamic lock area starts at byte 00_H of Block 34_H and contains 32 dynamic lock bits, where every lock bit is able to lock 4 bytes within the data area.

3.2.1.2 Memory Control TLV

The Memory Control TLV provides control information about the reserved area(s); this data covers information about the location and the size of the reserved bytes.

- **T: 02_H** indicates the Memory Control TLV which identifies reserved memory areas
- **L: 03_H** defines the length of the value field
- **V:** indicates the position of the reserved area and the number of reserved bytes
  - **$E0_H$:** defines the position of the reserved bytes inside the tag
    The higher nibble codes the 'PageAddr' by the number of pages, the lower nibble codes the 'ByteOffset' within the page.
  - **$2C_H$:** coded as the number of bytes it indicates the size of the reserved area in bytes (here: $2C_H$ to reserve 44 bytes of memory)
  - **04_H:** partial page control information indicates the size of a page ('BytesPerPage'); this information is coded to the lower nibble. The higher nibble is RFU.

Using these settings within the Memory Control TLV an NFC Forum™ device may calculate the byte address of the reserved area:

(2)

$$\mathrm{ByteAddr} = \mathrm{PageAddr} \times 2^{\mathrm{BytesPerPage}} + \mathrm{ByteOffset}$$

$$\mathrm{ByteAddr} = \mathrm{E_H} \times 2^4 + 0$$

$$\mathrm{ByteAddr} = 14 \times 16 = 212 (\mathrm{BlockAddr} = 35_H)$$

3.2.1.3 NDEF Message TLV

The NDEF Message TLV stores NDEF coded messages inside the Value field.

- **T: 03_H** indicates the NDEF message
- **L: 00_H** defines the length of the stored NDEF message (here: empty NDEF)
- **V:** not present

3.2.1.4 Terminator TLV

The Terminator TLV is the last TLV block in the data area of a NFC Forum™ Type 2 Tag.

- **T: FE_H** indicates the Terminator TLV
- **L:** not present
- **V:** not present

3.2.2 NDEF message in Dynamic Memory Structure: State READ / WRITE, CC3 = 00_H

Based on the INITIALIZED NFC Forum™ Type 2 Tag configuration given in **Chapter 3.2.1** following example shows a NFC Forum™ Type 2 Tag in READ/WRITE state with a dynamic memory structure, a NDEF message TLV and the Terminator TLV.

Table 27 Dynamic Memory Structure - READ/WRITE State (based on SLE 66R16P)

NFC Block address (hex)	Byte Number within a block			
	0	**1**	**2**	**3**
00_H	internal0	internal1	internal2	internal3
01_H	internal4	internal5	internal6	internal7
02_H	internal8	internal9	LOCK0 = 00_H	LOCK1 = 00_H
03_H	CC0 = $E1_H$	CC1 = 10_H	CC2 = $7E_H$	CC3 = 00_H
04_H	01_H	03_H	$D0_H$	20_H
05_H	24_H	02_H	03_H	$E0_H$
06_H	$2C_H$	04_H	03_H [1)]	13_H
07_H	$D1_H$	01_H	$0F_H$	54_H
08_H	02_H	64_H	65_H	48_H
09_H	65_H	$6C_H$	$6C_H$	$6F_H$
$0A_H$	20_H	77_H	$6F_H$	72_H
$0B_H$	$6C_H$	64_H	21_H	FE_H [2)]
$0C_H$	...			
...	...			
34_H	DynLock2 = 00_H	DynLock3 = 00_H	DynLock4 = 00_H	DynLock5 = 00_H
35_H	Reserved	Reserved	Reserved	Reserved
...	Reserved	Reserved	Reserved	Reserved
$3F_H$	Reserved	Reserved	Reserved	Reserved
40_H	...			
...	...			
FE_H	...			
FF_H	...			

1) Start of changed NDEF message record
2) Terminator TLV

Settings for the Static Lock bytes LOCK0 and LOCK1 and Capability Containers remain unchanged.

The data area starts at block 04_H and holds the unchanged values for Lock Control TLV and Memory Control TLV.

The NDEF message record is updated.

- NDEF message TLV
  - T: 03_H indicates the NDEF message
  - L: 13_H defines the length of the stored NDEF message
  - V: NDEF record holding the “Hello world!” payload according to the NFC Forum™ Text Record Type Definition

Note: The Read-only state can be configured by setting the CC3 byte to $0F_H$ (“soft-lock”).

4 Details about my-d™ NFC

The following section focusses on my-d™ NFC and the available derivatives.

4.1 my-d™ NFC Derivatives

The derivatives of the my-d™ NFC will be delivered with different settings to identify the chip. This information is programmed into the Service Area at manufacturing and cannot be modified. **Table 28** gives details on the chip variants.

Access to the memory can be executed with both the NFC Forum™ Type 2 Tag and the my-d™ command set.

Table 28 Chip variants of my-d™ NFC (memory sizes in bytes)

Chip type	Total memory size	Service Area	User Area (my-d™)	NFC memory size
SLE 66R16P	2560	24	2024	1024
SLE 66R32P	5120	24	4072	2048

The my-d™ NFC devices are available in UNINITIALIZED or in INITIALIZED configuration:

- in UNINITIALIZED state the whole memory in the my-d™ Data Area is set to 00_H and the memory needs to be initialized prior to be used in a NFC Forum™ Type 2 Tag infrastructure. The NFC sector can be set up using NFC Forum™ commands to enter the INITIALIZED and/or READ / WRITE state. Please refer to **Chapter 2.3**.
- in INITIALIZED state the my-d™ NFC devices come with the Capability Container bytes programmed and an empty NDEF message in the first NFC Forum™ Type 2 Tag sector. **Table 44** shows the memory content of a SLE 66R16P chip in INITIALIZED state. As long as this information is not locked (see **Chapter 4.4**) the data may be overwritten to use the my-d™ NFC as plain, standard device again (= UNINITIALIZED state).

4.2 my-d™ NFC Memory Configuration

Table 29 and **Table 30** show the default memory configuration of the my-d™ NFC devices. The whole memory is configured to be plain memory and the chip is in User mode.

Table 29 my-d™ NFC SLE 66R16P memory configuration

	my-d™ Page address	Byte Number within a page							
		0	1	2	3	4	5	6	7
my-d™ Service Area	0000_H	my-d™ Service Area							
	0001_H								
	0002_H								
my-d™ Data Area	0003_H	my-d™ User Data							
	0004_H								
	...								
	$007F_H$								
	0080_H	Dedicated 1 kByte NFC Forum™ Type 2 Tag memory Sector 0							
	...								
	...								
	...								
	$00FF_H$								

Table 30 my-d™ NFC SLE 66R32P memory configuration

	my-d™ Page address	Byte Number within a page							
		0	1	2	3	4	5	6	7
my-d™ Service Area	0000_H	my-d™ Service Area							
	0001_H								
	0002_H								
my-d™ Data Area	0003_H	my-d™ User Data							
	0004_H								
	...								
	...								
	$00FF_H$								
	0100_H	Dedicated 1 kByte NFC Forum™ Type 2 Tag memory Sector 0							
	...								
	...								
	$017F_H$								
	0180_H	Dedicated 1 kByte NFC Forum™ Type 2 Tag memory Sector 1[1)]							
	...								
	...								
	$01FF_H$								

1) Sector 1 has to be selected with the ‘Sector Select’ command

The general memory organization consists of pages with 8 bytes each. In the dedicated Type 2 Tag memory area two 4-byte NFC memory blocks are mapped into one my-d™ page. Additionally the my-d™ NFC provides an associated byte for each page to control read/write access (Access Condition byte, write-only).

The whole memory area (Service Area and User Area) is accessible with the my-d™ Command set. The given page address range is valid for the my-d™ Command set.

The NFC Forum™ Type 2 Tag memory is mapped into the my-d™ Data Area; the start address of the dedicated NFC Forum™ Type 2 Tag memory depends on the chip variant (see **Table 34**). This portion of the memory is accessible with the NFC Forum™ Type 2 Tag command set.

my-d™ NFC provides for each page an associated byte to control read/write access.

Table 31 NFC Forum™ Type 2 Tag memory mapping (based on SLE 66R32P)

	NFC Block address	Byte Number within a block				my-d™ page address		
		0	1	2	3			
	n/a[1]	my-d™ Service Area and my-d™ User Data				0000_H		my-d™ address range
						0001_H		
						...		
						$00FF_H$		
Type 2 Tag address range Sector 0	00_H	Dedicated 1 kByte NFC Forum™ Type 2 Tag memory Sector 0				0100_H	L[2]	
	01_H						H[3]	
	02_H					0101_H	L	
	03_H						H	
	...					...	...	
	...						...	
	FC_H					$017E_H$	L	
	FD_H						H	
	FE_H					$017F_H$	L	
	FF_H						H	
Type 2 Tag address range Sector 1	00_H	Dedicated 1 kByte NFC Forum™ Type 2 Tag memory Sector 1				0180_H	L	
	01_H						H	
	02_H					0181_H	L	
	03_H						H	
	...					...		
	...							
	FC_H					$01FE_H$	L	
	FD_H						H	
	FE_H					$01FF_H$	L	
	FF_H						H	

1) n/a ... not addressable with the NFC Forum™ Type 2 Tag command set

2) L ... even NFC block addresses references to byte 0 - 3 of a my-d™ page.
The lower nibble of the access byte controls the read/write access of the byte 0 - 3 of a my-d™ page

3) H ... odd NFC block addresses references to byte 4 - 7 of a my-d™ page.
The higher nibble of the access byte controls the read/write access of the byte 4 - 7 of a my-d™ page

4.2.1 my-d™ NFC Service Area

The Service Area is built-up with three my-d™ pages; they contain the unique chip ID (uid0 - uid6), the Issuer Tag (IT) and the Chip Information (CI). This information shall be used to determine

- chip type incl. memory size
- supported command set
- NFC memory configuration

Table 32 my-d™ NFC Service Area Content

	my-d™ Page address	Byte Number within a page							
		0	1	2	3	4	5	6	7
Service Area	0000_H	88_H [1]	uid0 [2]	uid1 [3]	uid2	uid3	uid4	uid5	uid6
	0001_H		IT [4]						
	0002_H	CI [5]							

1) Cascade Tag = 88_H according to ISO/IEC 14443-3 Type A
2) Manufacturer ID
3) Chip Family Identifier
4) Issuer Tag
5) Chip Information

The Service Area is outside the address range of the NFC Forum™ Type 2 Tag command set; in order to access the Service Area memory my-d™ commands have to be used. Writing to it is prohibited by the Read-only Access Condition which is stored in the internal chip administration area.

The Chip Family Identifier is available immediately after the anti-collision (page 0000_H); the chip manufacturer and the supported command set can be determined quickly.

Table 33 Chip Family Identification my-d™ NFC

UID field	Value	Description
uid0	05_H	Manufacturer ID according to ISO/IEC7816-6/AM1
uid1	20_H [1]	Identifies the support of the my-d™ and the NFC Forum™ Type 2 Tag command set Lower nibble: RFU and set to 0_H

1) Please note that the lower nibble shall not be checked for chip family identification!

In order to determine the exact chip variant it is necessary to select the chip (to read pages 0001_H and 0002_H). After selecting the my-d™ NFC chip memory access is possible for further operations.

The lower nibble of the Issuer Tag (IT) defines parts of the memory to be accessible with NFC Forum™ Type 2 Tag commands and the start address of the NFC sector.

Table 34 Issuer Tag (IT) of my-d™ NFC

Issuer Tag (lower nibble)	Type	NFC Sector Start address	NFC memory size [bytes]	Highest page address
0010_B	SLE 66R16P	0080_H	1024	$00FF_H$
0100_B	SLE 66R32P	0100_H	2048	$01FF_H$

Table 35 Chip Information (CI) of my-d™ NFC

Chip Information[1)]	Type
11xx $x100_B$	SLE 66R16P
11xx $x101_B$	SLE 66R32P

1) Please note that Bit[5:3] shall not be evaluated as this information may change for future revisions.

For more details please refer to the my-d™ NFC data book.

4.3 Command sets

Table 36 and **Table 37** list the commands which shall be used to set up the NFC sector in the my-d™ device.

my-d™ NFC devices can be operated using the NFC Forum™ Type 2 Tag command set only within the whole life cycle of the NFC platform.

Table 36 NFC Forum™ Type 2 Tag command set

Command[1)]	Op-Code	Comment	Address range
Sector Select[2)]	$C2_H$	Sector selection	-
Read	30_H	Reads 16 bytes of data	NFC sector, 00_H - FF_H
Write	$A2_H$	Writes 4 byte of data	NFC sector, 00_H - FF_H

1) Please refer to the NFC Forum™ Type 2 Tag specification for detailed command descriptions

2) For variants with more than 1 kByte of Type 2 Tag memory

It is recommended to use the my-d™ commands in order to clearly identify the chip type. This information is necessary to realize correct block locking within the my-d™ NFC devices. NFC memory block can only be physically locked with the my-d™ Write Byte command.

Table 37 my-d™ NFC command set

Command[1)]	Op-Code	Comment
my-d™ Read	10_H	Reads data from one page (8 bytes)
my-d™ Write	30_H	Writes data to one page (8 bytes)
my-d™ Write Byte	$E0_H$	Write data to the specified byte within a page

1) Please refer to the my-d™ NFC data book for detailed command descriptions

4.3.1 my-d™ Read

The my-d™ Read command reads 8 bytes of the memory (one page) if the Access Condition permits a read access.

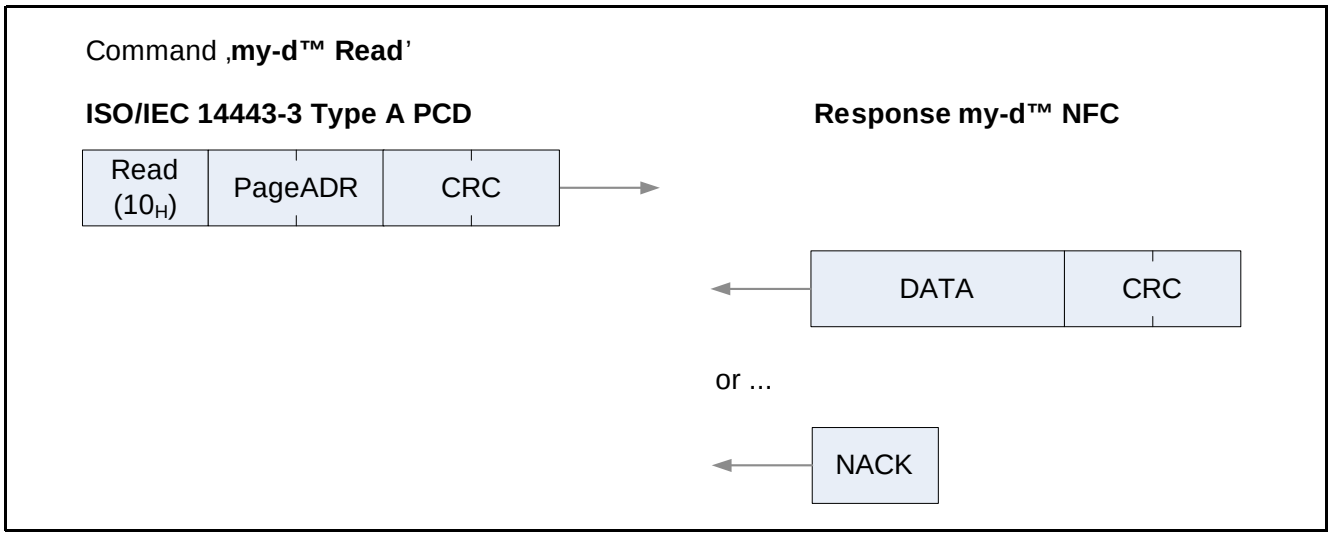

Figure 6 my-d™ Read Command

Table 38 'my-d™ Read' command parameters

Name	Length[1)]	Remark
Read	1	command opcode: 10_H
PageADR[2)]	2	Page address to be read The valid address range depends on product variant SLE 66R16P: 0000_H - $00FF_H$ SLE 66R32P: 0000_H - $01FF_H$
CRC	2	CRC

1) length of parameter field in [byte(s)]
2) lower byte sent first

Table 39 'my-d™ Read' command response parameters

Name	Length	Remark
DATA	8	Page content
CRC	2	CRC
or		
NACK	1	Error on Read command; replied without CRC

4.3.2 my-d™ Write

The my-d™ Write command writes data to the specified page (8 bytes).

In case of successful programming the my-d™ chip sends back an acknowledge. In case of an error or if the Access Conditions do not allow to write a “not acknowledge” will be returned to the PCD.

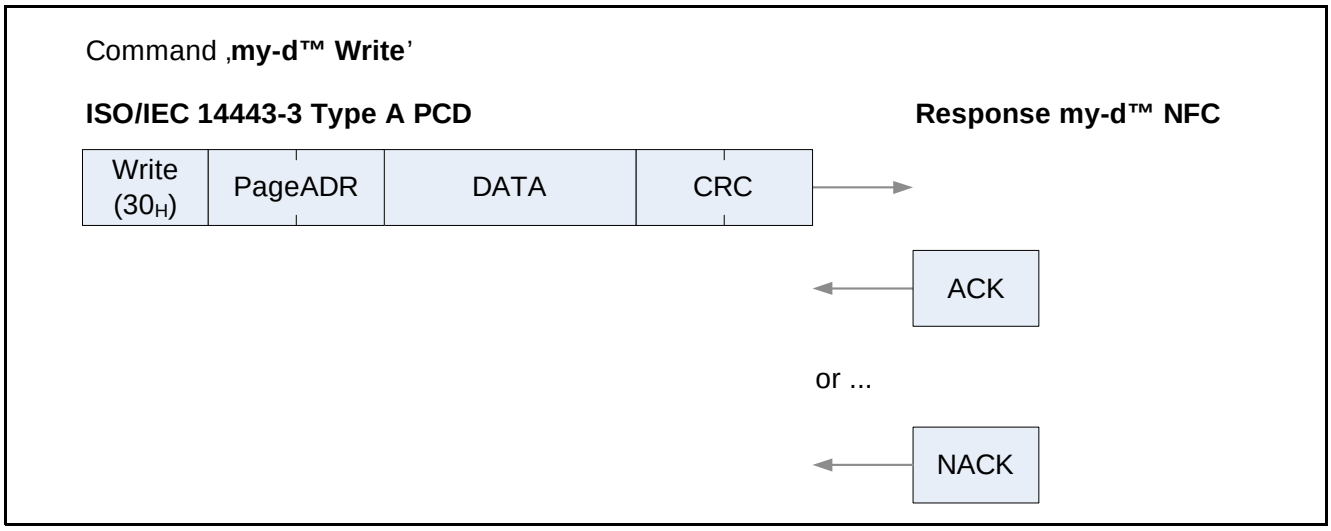

Figure 7 my-d™ Write Command

Table 40 ‘my-d™ Write’ command parameters

Name	Length[1)]	Remark
Write	1	command opcode: 30_H
PageADR[2)]	2	Page address to be written The valid address range depends on product variant SLE 66R16P: 0000_H - $00FF_H$ SLE 66R32P: 0000_H - $01FF_H$
DATA	8	Data to be written
CRC	2	CRC

1) length of parameter field in [byte(s)]

2) lower byte sent first

Table 41 ‘my-d™ Write’ command response parameters

Name	Length	Remark
ACK	1	Write successful; replied without CRC
NACK	1	Error on Write command; replied without CRC

4.3.3 my-d™ Write Byte

The my-d™ Write Byte command writes the data to the specified byte in the specified page. This command shall be used to modify the Access Condition of a page.

In case of successful programming the chip sends back an acknowledge frame.

In case of an error or if the Access Condition does not allow the access a “not acknowledge” frame will be send back to the PCD. Please note that the only possible Access Condition for the execution of this command is READ / WRITE (A_H).

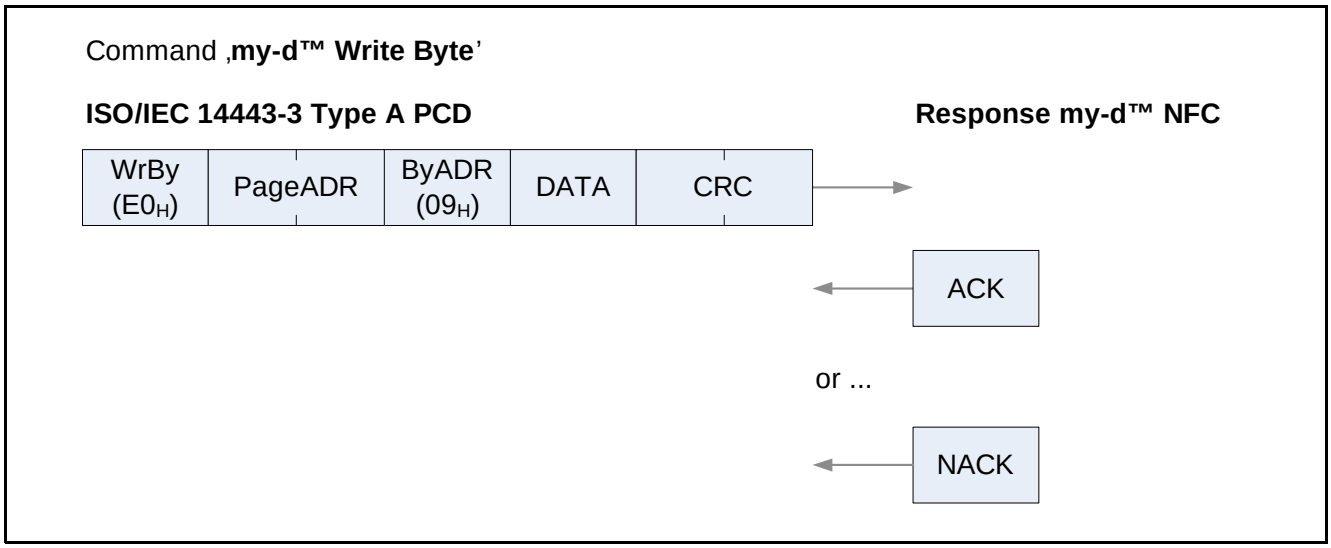

Figure 8 my-d™ Write Byte Command

Table 42 ‘my-d™ Write Byte’ command parameters

Name	Length[1)]	Remark
WrBy	1	command opcode: $E0_H$
PageADR[2)]	2	Page address to be written The valid address range depends on product variant SLE 66R16P: 0000_H - $00FF_H$ SLE 66R32P: 0000_H - $01FF_H$
ByADR	1	Byte address within a page; valid address range: 00_H - 09_H ByADR must be set to 09_H (address of Access Condition byte)
DATA	1	Data to be written
CRC	2	CRC

1) length of parameter field in [byte(s)]
2) lower byte sent first

Table 43 ‘my-d™ Write Byte’ command response parameters

Name	Length	Remark
ACK	1	Write successful; replied without CRC
NACK	1	Error on Write command; replied without CRC

4.4 Block locking with my-d™ NFC

my-d™ NFC devices may be in UNINITIALIZED (affects deliveries until 2011) or in INITIALIZED state (deliveries beginning in 2011 will be initialized already).

In UNINITIALIZED state the NFC sector is not prepared and the chip cannot be operated in an NFC Forum™ Type 2 Tag infrastructure. The NFC sector can be set up using NFC Forum™ commands to enter the INITIALIZED and/or READ / WRITE state. Please refer to **Chapter 2.3**.

In INITIALIZED state the my-d™ NFC chips come with the Capability Container bytes programmed and an empty NDEF message in the first NFC Forum™ Type 2 Tag sector.

Some applications may require to lock blocks in the NFC Forum™ Type 2 Tag user data area without changing the setting of the Capability Container to READ-ONLY (CC3 - read and write access capability).

my-d™ NFC offers irreversible block locking. This feature can be used by accessing the internal Adminstration Area memory of the chip. This portion of the memory can only be addressed with the my-d™ command set (listed in **Table 37**).

In order to correctly address the Adminstration area it is necessary to determine the exact chip type; for this reason the my-d™ Read command has to be used as the Service Area holding this information cannot be read with the NFC Forum™ Type 2 Tag command set to retrieve the information.

Figure 9 illustrates the sequence to determine the chip type.

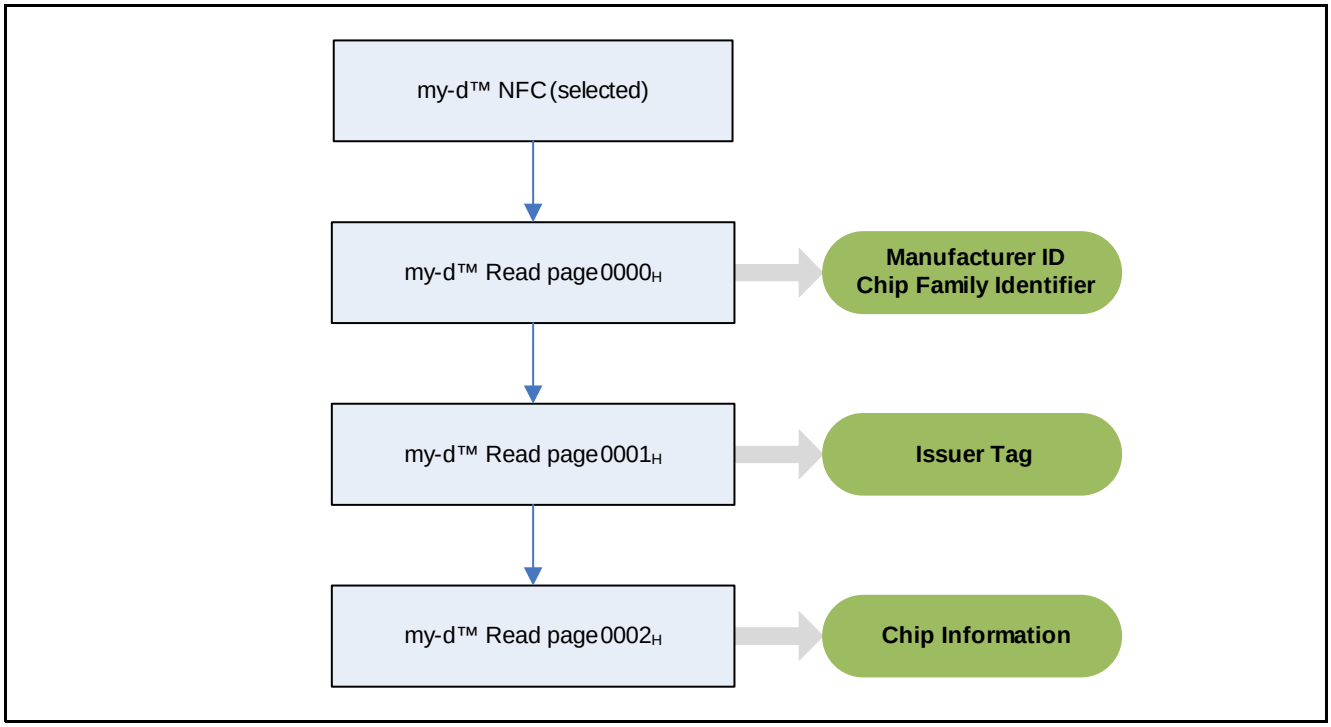

Figure 9 Chip type determination

After reading pages 0000_H to 0002_H the chip type, memory size and the start address of the NFC sector are identified.

To lock a block it is necessary to access the internal administration area to modify the Access Condition. For this reason the my-d™ Write Byte command has to be used. The default setting of the Access Condition allows READ / WRITE access.

As two NFC Forum™ Type 2 Tag blocks are mapped into one my-d™ page also the Access Condition is split into two parts; each nibble defines the access rights for a single Type 2 Tag block. The lower nibble of the Access Condition determines the access rights for the block with the even block address, whereas the block when the odd address is controlled by the higher nibble of the Access Condition of the corresponding my-d™ page.

Table 44 shows the mapping of the NFC Forum™ Type 2 Tag memory blocks into the my-d™ pages. It includes also the content of the my-d™ Service Area, but this memory portion is outside the Type 2 Tag address range. Furthermore the allocation of the Access Condition to my-d™ pages respectively NFC Forum™ Type 2 Tag memory blocks are listed.

Table 44 NFC Forum™ Type 2 Tag memory mapping (based on SLE 66R16P)

	NFC Block address	Byte Number within a block				my-d™ page address	Block reference	
		0	1	2	3			
	n/a	88_H	05_H [1]	20_H [2]	uid2	0000_H	L	my-d™ address range
		uid3	uid4	uid5	uid6		H	
		...	IT [3]	...	...	0001_H	L	
		...	...	...	...		H	
		CI [4]	...	...	...	0002_H	L	
		...	...	...	...		H	
		my-d™ User Data				0003_H	L	
							H	
						...	...	
						...	...	
						$007F_H$	L	
							H	
Type 2 Tag address range	00_H	Internal0	Internal1	Internal2	Internal3	0080_H	L	
	01_H	Internal4	Internal5	Internal6	Internal7		H	
	02_H	Internal8	Internal9	LOCK0=00_H	LOCK1=00_H	0081_H	L	
	03_H	CC0 = $E1_H$	CC1 = 10_H	CC2 = $7E_H$	CC3 = 00_H		H	
	04_H	03_H	00_H	FE_H		0082_H	L	
	05_H						H	
	...					...	...	
	...					...	...	
	FE_H					$00FF_H$	L	
	FF_H						H	

1) Manufacturer ID
2) Chip Family Identifier
3) Issuer Tag
4) Chip Information

The Access Condition for every my-d™ memory page is stored in the internal administration area of the my-d™ NFC chip. It is located at Byte 09_H on any my-d™ page. **Table 45** defines the available access rights. Please be aware that any value different to A_H or 6_H prevents any access to the corresponding block. **Chapter 4.5** summarizes the considerations for my-d™ NFC devices.

Table 45 Access Conditions and access rights

Access Condition[1)]	Access Right byte 0 - 3 (even block)	Access Right byte 4 - 7 (odd block)
AA_H	READ / WRITE	READ / WRITE
$A6_H$	READ / WRITE	READ-ONLY
$6A_H$	READ-ONLY	READ / WRITE
66_H	READ-ONLY	READ-ONLY
any other value	invalid	invalid

1) Access condition value $A6_H$ and 66_H lock the internal access condition.

4.5 Important Notes

In order to prevent problems using the Type 2 Tag memory within my-d™ NFC it is important to consider the following:

- Access Conditions can only be changed with my-d™ commands and not with Type 2 Tag commands
- to write-protect a certain Type 2 Tag block, the Access Condition in the lower nibble (for even pages) or the higher nibble (for odd pages) of the corresponding my-d™ page has to be set to 6_H (READ-ONLY)
- any Access Conditions other than A_H or 6_H prevent access to the Type 2 Tag memory
- the lower nibble of the Access Condition byte defines the access right for all my-d™ commands. If the lower nibble is set to 6_H, it is not possible to change the data content or the Access Condition anymore. Therefore, by programming the Access Condition byte to $A6_H$, page data bytes 0 to 3 are set to READ-ONLY whereas page data bytes 4 to 7 allow READ / WRITE access using the NFC command set.
- LOCK0 and LOCK1 values have to be programmed by the issuer/user and have no influence on the memory access rights of the Type 2 Tag memory. The memory can be altered with as long as the Access Condition allows the execution of the command.
- my-d™ Write Byte command shall only be used with the byte address parameter set to 09_H. Any other value may change unintentionally the Type 2 Tag memory content or the Access Condition. Access to certain blocks in the memory may be disrupted.

w w w . i n f i n e o n . c o m

Published by Infineon Technologies AG