Open NFC - NFC HAL Protocol Specification

Document Type:	Software Interface Specification
Reference:	SIS_NFC_0806-058 Version 3.9 (14684)
Release Date:	Aug. 31, 2012
File Name:	SIS_NFC_0806-058 Open NFC - NFC HAL Protocol Specification.pdf
Security Level:	General Business Use

Disclaimer

History

Version	Date	Comments
0.1	June 16, 2008	First Draft
3.0.1	June 30, 2008	Review 1
3.0.2	July 15, 2008	Review 2
3.0.3	Sept. 12, 2008	Second Draft
3.0.4	Oct. 30, 2008	Review 3
3.0.5	Feb. 6, 2009	Release for Open NFC 3.0
3.0.6	April 4, 2009	Release for Open NFC 3.1
3.0.7	July 3, 2009	Release for Open NFC 3.2
3.0.8	Oct. 4, 2009	Release for Open NFC 3.3
3.0.9	Dec. 15, 2009	Update for Open NFC 3.4 - Renaming the document, updating the SE management
3.1	July 9, 2010	Technology Preview
3.2	Oct. 15, 2010	Adding support for the Secure Element. Delivery for Open NFC 4.1
3.3	Dec. 24, 2010	Adding the support for the raw mode. Delivery for Open NFC 4.2
3.4	May 16, 2011	Delivery for Open NFC 4.3 Beta Adding explanation of the battery off configuration Adding the support of multi-card detection (FeliCa) Updating the reader B Prime specification. Renaming the document from FSP_NFC_0806-058 into SIS_NFC_0806-058.
3.5	June 1, 2011	Delivery for Open NFC 4.3 Added the reader Kovio specification. Modified the reader 14443-3A specification for support of NFC Forum Type 2 non-standard ACK and NACK frames. Updating the specification of the P2P protocol to add the connection speed in the discovery events and to add the flags F-only and Passive-only.
3.6	Aug. 16, 2011	Delivery for Open NFC 4.3.1 Updating the interface with the Secure Elements and the UICC.
3.7	Dec. 14, 2011	Added new 14443-3A bit oriented data exchange capability Added Mifare Classic commands description in 14443-3A reader description
3.8	June 1, 2012	Delivery for Open NFC 4.4.3 Added routing table specification
3.9	Aug. 31, 2012	Delivery for Open NFC 4.5.0, Clarified Kovio tags support, Updated event NAL_EVT_SE_CARD_EOT.

Summary of Contents

Reference Documents

[14443-3] ISO/IEC 14443-3:2001(E) - Identification cards - Contactless integrated circuit(s) cards - Proximity cards - Part 3: Initialization and anticollision

[14443-4] ISO/IEC 14443-4:2001(E) - Identification cards – Contactless integrated circuit(s) cards - Proximity cards - Part 4: Transmission protocol

[15693-2] ISO/IEC 15693-2:2001(E) - Identification cards - Contactless integrated circuit(s) cards - Vicinity cards - Part 2: Air interface and initialization

[15693-3] ISO/IEC 15693-3:2001(E) - Identification cards - Contactless integrated circuit(s) cards - Vicinity cards - Part 3: Anticollision and transmission protocol

[ISO 18092] ISO/IEC 18092:2004(E) - Information technology - Telecommunications and information exchange between systems - Near Field Communication -Interface and Protocol (NFCIP-1)

[ISO 7816-4] ISO/IEC 7816-4:1995(E) – Information Technology – Identification Cards – Integrated Circuit(s) Cards with contacts – Part 4: Inter-industry commands for interchange

[PICOPASS] picopass chip specification are available from Inside Contactless (http://www.insidecontactless.com).

[TOPAZ] Topaz and Jewel cards specifications are available from Innovision Research & Technology plc (http://www.innovision-group.com).

[SWP] ETSI TS 102 613 V7.3.0 (2008-08) Smart Cards; UICC-CLF interface; Physical and data link layer characteristics (Release 7)

[NDEF] “NFC Data Exchange Format (NDEF) Specification”, NFC Forum http://www.nfc-forum.org), 2006.

[RTD] “NFC Record Type Definition (RTD) Specification”, NFC Forum, 2006.

[NFC TS 1] “NFC Type 1 Tag Operation Specification”, v.1.1, NFC Forum

[NFC TS 2] “NFC Type 2 Tag Operation Specification”, v.1.1, NFC Forum

[NFC TS 3] “NFC Type 3 Tag Operation Specification”, v.1.0, NFC Forum

[NFC TS 4] “NFC Type 4 Tag Operation Specification”, v.2.0, NFC Forum

[NFC TS 5] SPE_NFC_0707_001 NFC Type 5 Tag Operation Specification for Picopass tags, Inside Contactless

[NFC TS 6] SPE_NFC_0707_002 NFC Type 6 Tag Operation Specification for ISO 15693-3 tags, Inside Contactless

[LLCP] Logical Link Control Protocol, Technical Specification v1.1, NFC Forum

[HCI] ETSI TS 102 622 V7.2.0 (2008-08) Smart Cards; UICC-CLF interface;

Host Controller Interface (Release 7)

1 Introduction

This document is the specification of the interface between the Open NFC software stack and a NFC hardware component called the NFC Controller.

Audience

This specification document should be read by the following audience:

- Technologists wanting to understand the functionalities shared between a NFC software stack and a NFC Controller.
- Architects and developers of NFC software stack.
- Architects and developers of firmware for the NFC Controller.

Definitions

The following definitions are used in this document:

- The "*NFC Controller"* is the hardware chip providing the NFC functionalities.
- The *"NFC Controller Firmware"* is the software executed by the NFC Controller.
- The "*NFC Device"* is the device containing the NFC Controller and processor executing the NFC Stack.
- The "*NFC Stack"* is the software stack executed by the NFC Device to control the NFC Controller.
- The "*NFC Controller Hardware Abstraction Layer"* (NFC HAL) is the protocol used between the NFC Stack and the NFC Controller.
- The "*UICC*" is a separate hardware component including a processor and connected to the NFC Controller through the SWP physical link.
- The "*SE*" (Secure Element) is a hardware component including a processor and connected to the NFC Controller. The SE may be stacked within the NFC Controller package. Several secure elements may be connected to the NFC Controller. Some of them may be removable.

The next figure illustrates the physical location of the different elements described in this specification.

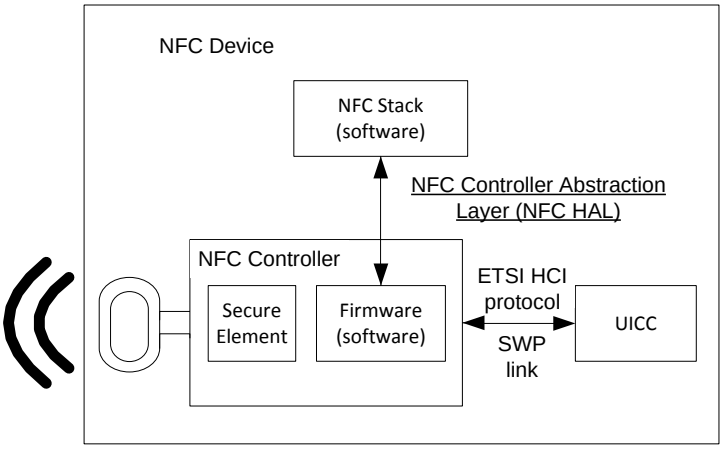

Figure 1 – Physical Overview

Version

The version of the interface is identified by a number. This specification describes the version NAL_VERSION of the interface (see constant values in Appendix A).

Chapters

The following topics are addressed in the next chapters:

- Chapter 2: describes the principles of design used for the NFC Controller HAL.
- Chapter 3: describes the main configuration parameters.
- Chapter 4: describes the reader protocols for Type 1, ISO 14443-3 A, FeliCa, ISO 14443-4 A and B, ISO 14443-3 B and ISO 15693.
- Chapter 5: describes the card protocols ISO 14443-4 A and B.
- Chapter 6: describes the Peer 2 Peer protocols.
- Chapter 7: describes the UICC management including the access policy and the communication.
- Chapter 8: describes the Secure Element management and communication
- Chapter 9: describes the power management of the NFC Controller, the boot procedure, the standby mode and the resume procedure.
- Chapter 10: describes the commands related to the detection process for the card protocols, the reader protocols and the Peer 2 Peer protocols.
- Chapter 11: describes the test functions.
- Appendix A: describes the constants values

2 Overview

This chapter describes the architecture implementing the NFC Controller HAL, the principles of the protocol, the generic syntax of the messages and the notations used in this document.

2.1 Architecture

The NFC Controller HAL (NFC HAL) defines the high level functions required by a NFC Device to communicate with a NFC Controller. The NFC HAL does not define the low level binding to a specific physical link with a NFC Controller. Such physical binding should be implemented for each specific physical link and NFC Controller.

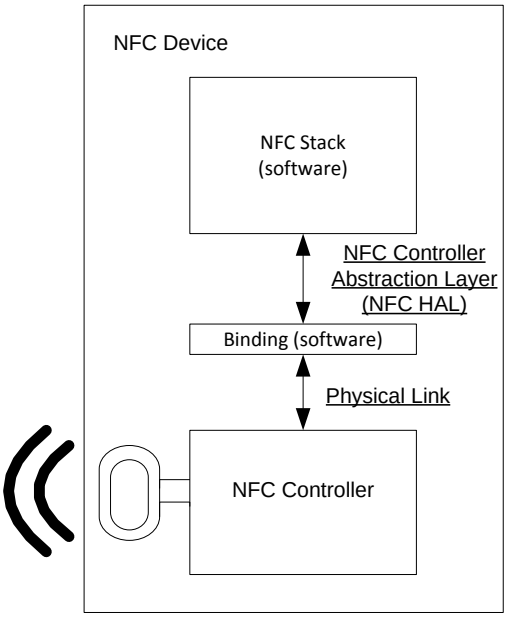

Figure 2 – Example of Physical Binding

The following features shall be provided by the physical binding:

- connection and disconnection detection,
- message fragmentation and reassembly,
- message integrity checking with message repetition in case of error,
- integrated handling of the length of the messages,
- syntax adaptation for specific NFC Controller.

2.2 Notation and Terminology

This specification uses the following terminology:

- A bit value is *set* when its value is one. A bit value is *not set* or *reset* when its value is zero.
- *RFU* means “Reserved for Future Use”.
- *SBZ* means “Should Be Zero”.
- SBLU means “Should Be Left Unchanged”

2.2.1 Variables and Constants

Variable values are used in the explanations and in the figures of this document. Their name is prefixed with “VAR_”. Constant values are also used in the document. Their name is prefixed with “CST_”.

The variables and the constants are used to explain the interface and the behavior of the NFC Controller. The variables and constants are not necessarily used in the implementation of the NFC Controller firmware. The only requirement for the NFC Controller and the NFC HAL Binding is to follow exactly the NFC HAL and the behavior described in this document.

2.2.2 Protocol Bit Field

In this specification, the card and the reader protocols are identified in a 16-bit field. The tables below describe the bit field mapping:

Part “R”: The reader protocols

Bit Position	15-13	12	11	10	9	8	7	6	5	4	3	2	1	0
Meaning	RFU-SBZ	Mifare Plus	Kovio RF barcode ROM ID Tag	B’	Mifare Classic	NFC Type 1	P2P Ini-tiator	Felica protocol	ISO 15693 level 2	ISO 15693 level 3	ISO 14443 B level 3	ISO 14443 A level 3	ISO 14443 B level 4	ISO 14443 A level 4

Part “C”: The card protocols

Bit Position	15-13	12	11	10	9	8	7	6	5	4	3	2	1	0
Meaning	RFU-SBZ	Mifare Plus	Kovio FR barcode ROM ID Tag	B’	Mifare Classic	NFC Type 1	P2P Target	FeliCa & Type F CLT	ISO 15693 level 2	ISO 15693 level 3	ISO 14443 B level 3	ISO 14443 A level 3	ISO 14443 B level 4	ISO 14443 A level 4 & Type A CLT

The meaning of a 1 or a 0 for a given bit is explained for each variable.

In this document, if a variable “VAR” includes a protocol bit field:

- the reader part is represented with the notation “VAR-R”.
- the card part is represented with the notation “VAR-C”.

2.3 Protocol Principles

The NFC Controller HAL is based on the following principles:

2.3.1 Persistence

Unless explicitly stated in this document, none of the NFC Controller parameter is persistent. The default value for each volatile parameter is given in this specification.

2.3.2 Undefined Messages

All the messages, events and parameters used by the NFC HAL are defined in this document.

A device shall never send a message not defined in this specification. Sending such an unknown message may cause unspecified behavior from the NFC Controller including a modification of the persistent parameters. In no case may this corrupt the NFC Controller firmware. After a reboot of the NFC Controller, the behavior of the NFC controller shall return to a normal state described in this specification.

2.4 Protocol Syntax

This section describes the generic syntax of the NFC HAL messages. Every exchange of information between the NFC Device and the NFC Controller is based on messages.

2.4.1 Messages

There is no generic limit to the length of a message. However the length of each NFC HAL message is implicitly limited by the nature of the data included in the message.

The generic syntax of a message is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	The service code
1	1	The message type and the command code or the result code
2	N	The message data. N may be zero.

There are 3 types of messages: the command messages, the answer messages and the event messages. The two most significant bits of the second byte of the message define the type of the message. The encoding is the following:

bit7	bit 6	Message Type	bit 5 to bit 0
0	0	Command	The command code
0	1	Answer	The result code
1	0	Event	The event code
1	1	RFU	-

2.4.2 Services

Every NFC HAL message is related to a given service. The service is identified by a code in the first byte of the message.

The following table gives the list of the services:

Name	Description
NAL_SERVICE_ADMIN	The administration service is used for most of the generic functions not related to any particular protocol (card, reader or peer to peer)
Reader Services	
NAL_SERVICE_READER_14_A_4	The reader service for ISO 14443-A level 4
NAL_SERVICE_READER_14_B_4	The reader service for ISO 14443-B level 4
NAL_SERVICE_READER_14_A_3	The reader service for ISO 14443-A level 3
NAL_SERVICE_READER_14_B_3	The reader service for ISO 14443-B level 3
NAL_SERVICE_READER_TYPE_1	The reader service for Type 1 tags
NAL_SERVICE_READER_FELICA	The reader service for FeliCa cards
NAL_SERVICE_READER_15_3	The reader service for ISO 15693 level 3
NAL_SERVICE_READER_15_2	The reader service for ISO 15693 level 2
NAL_SERVICE_READER_B_PRIME	The reader service for B Prime cards
NAL_SERVICE_READER_KOVIO	The reader service for Kovio RF barcode ROM ID cards
Card Service	
NAL_SERVICE_CARD_14_A_4	The card service for ISO 14443-A level 4
NAL_SERVICE_CARD_14_B_4	The card service for ISO 14443-B level 4
Peer 2 Peer	
NAL_SERVICE_P2P_INITIATOR	The P2P service for the initiator side
NAL_SERVICE_P2P_TARGET	The P2P service for the target side
UICC	
NAL_SERVICE_UICC	The UICC service
Secure Element	
NAL_SERVICE_SECURE_ELEMENT	The Secure Element service

Note
The NFC HAL does not include the notion of “pipes” defined in **[HCI]**. The notion of pipe creation and the open/close commands are not required.

2.4.3 Commands

A command is identified by a command code. A command may optionally include some data. The receiver of a command always returns an answer message. The answer message includes a success code or an error code and optionally some data.

Depending on their meaning, commands can be issued in both directions (NFC Device to NFC Controller or NFC Controller to NFC Device).

For a given service and a given direction, the commands cannot be interleaved. No new command message should be sent between the sending of a command and the reception of the command answer.

The encoding of a command is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	The service code
1	1	The command code.
2	N	The command data. N may be zero.

The encoding of the answer is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	The service code
1	1	The result code (success or an error code). The NFC HAL command codes are defined in this document.
2	N	The answer data in case of success (N may be zero). In case of error, N is zero.

2.4.4 Command NAL_CMD_SET_PARAMETER

The generic command NAL_CMD_SET_PARAMETER is used to set a parameter.

The encoding of the command is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	The service code
1	1	NAL_CMD_SET_PARAMETER
2	1	The parameter code.
3	N	The new parameter data.

The encoding of the answer is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	The service code
1	1	The result code (success or an error code).

The command returns the following result codes:

Result Code	Meaning
NAL_RES_OK	Success, the parameter is set
NAL_RES_UNKNOWN_PARAM	Error : unknown parameter
NAL_RES_BAD_LENGTH	Error : wrong data length for this parameter (out of the range specified in this specification)
NAL_RES_BAD_DATA	Error : wrong data value for this parameter (out of the value specified in this specification).
NAL_RES_FEATURE_NOT_SUPPORTED	Operational error: the data value and length is valid for this specification but the NFC Controller does not support it. This error is

	explicitly documented for each parameter.

2.4.5 Command NAL_CMD_GET_PARAMETER

The generic command NAL_CMD_GET_PARAMETER is used to get a parameter.

The encoding of the command is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	The service code
1	1	NAL_CMD_GET_PARAMETER
2	1	The parameter code.

The encoding of the answer is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	The service code
1	1	The result code (success or an error code).
2	N	The value of the parameter in case of success. N is the size of the parameter. In case of error, N is zero.

The command returns the following result codes:

Result Code	Meaning
NAL_RES_OK	Success, the parameter is read
NAL_RES_UNKNOWN_PARAM	Error : unknown parameter

2.4.6 Events

An event is identified by an event code. An event may optionally include some data. The receiver of an event does not return an answer message.

Depending on their meaning, events can be issued in both directions (NFC Device to NFC Controller or NFC Controller to NFC Device).

The encoding of an event is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	The service code
1	1	The event code. The encoding follows the HCI specification **[HCI]**. The NFC HAL event codes are defined in this document.
2	N	The event data. N may be zero.

2.4.7 Endianess

In the protocol messages, the 2-bytes long integer values and the 4-bytes long integer values are encoded with the big endian convention.

2.4.8 Protocol Errors

When a message is received by the NFC Device or the NFC Controller, the following rules apply by priority order:

- If the service code is unknown, the message is ignored.
- If the message type is different from command, answer or event, the message is ignored.
- If the message is an event:
  a) If the event code is unknown or is not defined for the service, the message is ignored.
  b) If the length of the payload is invalid, the message is ignored.
  c) If the content of the payload is invalid, the behavior is unspecified.
- If the message is a command:
  a) If the command code is unknown or is not defined for the service, the payload of the command is ignored and an answer message is returned with the result code NAL_RES_UNKNOWN_COMMAND.
  b) If the length of the payload is invalid, the message is ignored.
  c) If the content of the payload is invalid, the behavior is unspecified.
- If the message is an answer:
  a) If no command is pending for the service, the message is ignored.
  b) If the result code is unknown or is not defined for the pending command, the message is ignored.
  c) If the length of the payload is invalid, the message is ignored.
  d) If the content of the payload is invalid, the behavior is unspecified.

2.4.9 Unsupported Features

If a value, a range, a size or a feature is specified in this specification but is not supported by a specific NFC controller, the commands returns the error code NAL_RES_FEATURE_NOT_SUPPORTED. This specification describes explicitly this error when it is returned by a parameter setting or by a command execution.

3 Configuration

The NFC Controller configuration is implemented by the service NAL_SERVICE_ADMIN.

3.1 Variables and Constants

This section describes the variables and constant values used in the document.

3.1.1 Constant CST_CAPABILITIES (-R or -C)

This constant value represents the protocols implemented by the NFC Controller. The value is a protocol bit-field with the following meaning:

- If a bit is set, the corresponding protocol is implemented by the NFC Controller firmware.
- If a bit is not set, the corresponding protocol is not implemented by the NFC Controller firmware.

3.1.2 Variable VAR_DEVICE_DETECTION (-R or -C)

This volatile variable is a protocol bit field representing the reader protocols requested by the NFC Device for the detection sequence.

- If a bit is set, the corresponding protocol is requested for the detection sequence.
- If a bit is not set, the corresponding protocol is not requested for the detection sequence.

3.1.3 Variable VAR_ACTIVE_PROTOCOL (-R or -C)

This volatile variable represents the current active protocol:

- A reader protocol is active when card (or the Secure Element) is detected for this protocol and the NFC controller is emitting a RF field to communicate with this card.
- A card protocol is active when this protocol is used for the detection sequence, an external reader has selected the emulated card and the external reader is emitting a RF field to communicate with the card.

VAR_ACTIVE_PROTOCOL is a protocol bit field with zero or one bit set. If the protocol N is active, the bit VAR_ACTIVE_PROTOCOL(N) is set.

3.2 Configuration Parameters

3.2.1 Parameter NAL_PAR_PERSISTENT_POLICY

This parameter stores the persistent access policy of the NFC Controller:

- The loader/firmware reads this parameter at boot time in battery on mode, in battery low mode or in battery off mode. The parameter is not modified by the loader/firmware.
- The modification of this parameter by the NFC stack is stored in persistent memory but has no immediate effect.
- If an invalid configuration is written in this parameter, the behavior of the NFC Controller is unspecified.

Service	NAL_SERVICE_ADMIN
Code	NAL_PAR_PERSISTENT_POLICY
Access	Read & Write
Storage	Persistent
Size	0x08 bytes
Default	0x0000000300000000

The mapping of the parameter is the following:

Offset	Size	Field	Description
0	2	UICC Card Policy	The structure of this bit field follows the mapping of the card protocols. • If every bit of this field is cleared, the loader/firmware does not power up the UICC in battery off mode. The card emulation protocols are not allowed for the UICC in battery on mode. • If at least one of the bits is set, the loader/firmware does power up the UICC in battery off mode. The corresponding card emulation protocols are allowed for the UICC in battery on mode and in battery off mode. After the download of new NFC Controller Firmware, the default value of the parameter is 0x0000 i.e.: Card protocols are not allowed for the UICC.
2	2	UICC Reader Policy	The structure of this bit field follows the mapping of the reader protocols. After the download of new NFC Controller Firmware, the default value of the parameter is 0x0000 i.e.: Reader protocols are not allowed for the UICC
4	2	Policy Flags	The policy flags are described below
6	2	SE Filter	The structure of this bit field follows the mapping of the card protocols.

The mapping of the policy flags is the following:

Position (bits)	15-8	7-6	5-4	3-2	1	0
Field	RFU-SBZ	BATT-OFF	SE-ID	SE-SW	CL	RL

		SW				

After the download of new NFC Controller Firmware, the default value of the flags is 0x0000000300000000 i.e.:

- The SE(s), if any, are not connected.
- The reader protocols and the card protocols are not locked.
- The UICC has no access to reader and card protocols.

Bit 0 (RL)
If the RL bit is not set, the reader protocols are locked at the RF level for the NFC Device and the UICC. If the RL bit is set, the reader protocols are not locked.

Bit 1 (CL)
If the CL bit is not set, the card protocols are locked at the RF level for the NFC Device, the UICC and the SE(s). If the CL bit is set, the card protocols are not locked.

Bits 2-3, 4-5, (SE-SW and SE-ID)
The position of each SE switch is defined by the value of the SE fields. The values of the SE switch are defined in the section “Position of the SE Switch”.

Bits 7-6 (BATT OFF SW)

These two bits are write-only, meaning they are always set to zero when retrieved from the NFCC device using NAL_CMD_GET_PARAMETER.

They are used to define which secure elements are to be powered in BATT-OFF mode.

When set, Bit-6 indicates the UICC should be powered in BATT-OFF mode.
When set, Bit-7 indicates the SE should be powered in BATT-OFF mode.
When both bit are reset, there’s no change done in the BATT-OFF configuration.

3.2.2 Parameter NAL_PAR_POLICY

This parameter stores the dynamic policy information:

- At boot time in battery on mode, this parameter is initialized with the value stored in NAL_PAR_PERSISTENT_POLICY. After the boot procedure, the parameter is not modified by the loader/firmware.
- The firmware interprets the modifications of value of this parameter as an instruction to undertake an immediate change in the configuration.

Service	NAL_SERVICE_ADMIN
Code	NAL_PAR_POLICY
Access	Write only
Storage	Volatile
Size	0x08 bytes
Default	Copied from NAL_PAR_PERSISTENT_POLICY

This volatile parameter follows the mapping of the parameter NAL_PAR_PERSISTENT_POLICY. See this parameter for the meaning of the fields and the valid configurations. The SE bits can be set with the “Host Interface” value (2). If an invalid configuration is written in this parameter, the behavior of the NFC Controller is unspecified.

The BATT-OFF SW bits are not relevant in NAL_PAR_POLICY.

3.3 NFC Controller Version and Capabilities

The NFC Controller Version and Capabilities are the first parameters requested during the boot sequence. The NFC stack uses these parameters to identify the version of the NFC HAL and to get the main capabilities of the NFC Controller.

3.3.1 Parameter NAL_PAR_HARDWARE_INFO

Describes the hardware and loader information including the following elements:

- NFC Protocol version
- NFC Controller Hardware type and version
- NFC Controller Hardware serial number
- Loader type and version
- Presence of the firmware
- SE Number and capabilities

The NFC Device reads this parameter only once during the boot sequence. Since the content of this parameter is constant, it is not read again afterward.

Service	NAL_SERVICE_ADMIN
Code	NAL_PAR_HARDWARE_INFO
Access	Read only
Storage	Persistent
Size	0xFB bytes

The mapping of the parameter is the following:

Offset	Size	Field	Description
0	1	NFC HAL Protocol Version	The version of the protocol defined in this specification. This value should be NAL_VERSION if the protocol follows this specification.
1	32	Hardware Type and Version	The hardware type and version. The format is a zero ended ASCII string. The value is defined by the NFC Controller Manufacturer.
33	32	Hardware Serial Number	The serial number of the NFC controller chip. The format is a zero ended ASCII string. The value is defined by the NFC Controller Manufacturer.
65	32	Loader Description	The loader type and version. The format is a zero ended ASCII string. The value is defined by the NFC Controller Manufacturer.
97	1	Firmware status	0 if no firmware is present, 1 if a valid firmware is present.
98	1	SE Slot Number	Between 0 and 4 slots are supported
99	32	SE #1 Description	Description of the SE #1. The format is a zero ended ASCII string. The value is defined by the NFC Controller Manufacturer. Empty if there is no SE.
131	2	SE #1 Capabilities	See section “8.1 Secure Element ”. 0 if no there is no Secure Element slot.
133	2	SE #1 Host Interface	See section “8.1 Secure Element ”. 0 if no there is no Secure Element slot.
135	2	SE #1 RF Interface	See section “8.1 Secure Element ”. 0 if no there is no Secure Element slot.
137	32	SE #2 Description	Description of the SE #2. The format is a zero ended ASCII string. The value is defined by the NFC Controller Manufacturer. Empty if there is no SE.
169	2	SE #2 Capabilities	See section “8.1 Secure Element ”. 0 if no there is no Secure Element slot.
171	2	SE #2 Host Interface	See section “8.1 Secure Element ”. 0 if no there is no Secure Element slot.
173	2	SE #2 RF Interface	See section “8.1 Secure Element ”. 0 if no there is no Secure Element slot.
175	32	SE #3 Description	Description of the SE #3. The format is a zero ended ASCII string. The value is defined by the NFC Controller Manufacturer. Empty if there is no SE.
207	2	SE #3 Capabilities	See section “8.1 Secure Element ”. 0 if no there is no Secure Element slot.
209	2	SE #3 Host Interface	See section “8.1 Secure Element ”. 0 if no there is no Secure Element slot.
211	2	SE #3 RF Interface	See section “8.1 Secure Element ”. 0 if no there is no Secure Element slot.
213	32	SE #4 Description	Description of the SE #4. The format is a zero ended ASCII string. The value is defined by the NFC Controller Manufacturer. Empty if there is no SE.
245	2	SE #4 Capabilities	See section “8.1 Secure Element ”. 0 if no there is no Secure Element slot.
247	2	SE #4 Host Interface	See section “8.1 Secure Element ”. 0 if no there is no Secure Element slot.

Offset	Size	Field	Description
249	2	SE #4 RF Interface	See section “8.1 Secure Element ”. 0 if no there is no Secure Element slot.

3.3.2 Parameter NAL_PAR_FIRMWARE_INFO

Describes the firmware information including the following elements:

- NFC Protocol version
- Firmware version and build number
- Reader and Card emulation capabilities
- Other Firmware Capabilities

The NFC Device reads this parameter only once during the boot sequence. Since the content of this parameter is constant, it is not read again afterward.

Service	NAL_SERVICE_ADMIN
Code	NAL_PAR_FIRMWARE_INFO
Access	Read only
Storage	Persistent
Size	0x2F bytes
Default	Set by the firmware

The mapping of the parameter is the following:

Offset (in bytes)	Size (in bytes)	Field	Description
0	1	NFC HAL Protocol Version	The version of the protocol defined in this specification. This value should be NAL_VERSION if the protocol follows this specification.
1	32	Firmware Description	The firmware type, version, build number, etc... The format is a zero ended ASCII string. The value is defined by the NFC Controller Manufacturer.
33	2	Card Protocol Capabilities	The value of the constant CST_CAPABILITIES-C
35	2	Reader Protocol Capabilities	The value of the constant CST_CAPABILITIES-R
37	2	Firmware Capabilities	See below.
39	1	Reader ISO A data rate max	The maximum data rate supported by the NFC Controller for the ISO 14443 A reader protocol. The encoding follows the data rate max parameter of the ISO 14443 A reader service.
40	1	Reader ISO A input buffer size	The maximum size for input data received in a frame by the ISO A level 4 reader. This value is sent by the NFC Controller in the RATS frame. The value is in the range [0, 8] following the encoding of the FSDI in **[ISO 14443-4]**.
41	1	Reader ISO B data rate max	The maximum data rate supported by the NFC Controller for the ISO 14443 B reader protocol. The encoding follows the data rate max parameter of the ISO 14443 B reader service.
42	1	Reader ISO B input buffer size	The maximum size for input data received in a frame by the ISO B level 3 reader. This value is sent by the NFC Controller in the ATTRIB frame. The value is in the range [0, 8] following the encoding of the ATTRIB in **[ISO 14443-3]**.
43	1	Card ISO A data rate max	The maximum data rate supported by the NFC Controller for the ISO 14443 A card protocol. The encoding follows the data rate max parameter of the ISO 14443 A card service.
44	1	Card ISO B data rate max	The maximum data rate supported by the NFC Controller for the ISO 14443 B card protocol. The encoding follows the data rate max parameter of the ISO 14443 B card service.
45	2	NFC controller auto standby timeout	The duration of the timeout in milliseconds before NFC controller enters spontaneously in standby mode when no reader mode is activated or when the RF reader is locked.

Firmware Capabilities

The firmware capabilities are defined as a bit field of two bytes with the following structure:

Bit Position	Meaning
0	Battery off supported
1	Battery low supported
2	Standby supported
3	CID supported for the card emulation ISO 14443 A
4	NAD supported for the card emulation ISO 14443 A
5	CID supported for the card emulation ISO 14443 B
6	NAD supported for the card emulation ISO 14443 B
7	CID supported for the reader ISO 14443 A
8	NAD supported for the reader ISO 14443 A
9	CID supported for the reader ISO 14443 B
10	NAD supported for the reader ISO 14443 B
11	Support the Picopass UID detection in the reader ISO 14443 B
12	Bit oriented exchanges supported for reader ISO 14443-3A
13	Routing table supported by the firmware.
14-15	RFU-SBLU

When the bit is set, the feature is implemented by the firmware. When the bit is not set, the feature is not available. If the protocol is not supported, a feature related to this protocol is not available.

If the protocol detection is not implemented, the reader protocols are activated one by one during the detection sequence until a card is detected. In this mode, the parameter NAL_PAR_DETECT_PULSE has no effect. If the protocol detection is implemented, a detection pulse is emitted with an interval defined by NAL_PAR_DETECT_PULSE. When a card is detected, the detection protocol automatically selects the right protocol.

3.3.3 Parameter NAL_PAR_PERSISTENT_MEMORY

The NFC Device persistent memory “parameter” is used as a persistent storage area for the NFC stack. The NFC Device uses this parameter to store its persistent data and settings. The value stored in this area has no effect on the behavior of the NFC Controller.

Service	NAL_SERVICE_ADMIN
Code	NAL_PAR_PERSISTENT_MEMORY
Access	Read & Write
Storage	Persistent
Size	0x08 bytes
Default	When the firmware is downloaded, the initial content of this parameter is 0 for all the bytes.

4 Reader Protocols

This chapter describes the general principles of the reader protocols and then describes the specificities of each protocol.

4.1 Common Principles

Single Registry
The access to the reader protocols is controlled by the NFC Device. The “reader pipes” of the UICC are partially redirected through the NFC Device.

The reader registry contains the detection sequence configuration of each protocol (for example ISO 15693 requires only one byte: the AFI byte) and the list of the card detected for reader protocols that support multi-card detection (see below).

Detection sequence Configuration
The reader protocol configuration used during card detection depends on each protocol. The protocol configuration is volatile and its default value is set during the boot of the NFC Controller.

Detection Sequence Command
The single command NAL_CMD_DETECTION is used to start and stop the reader protocols and the card protocols. This command is global to all the protocols.

Card Detection Events
The card detection events are specific to each protocol. They carry all the information gathered during the card discovery. There is no need to keep this information in RAM parameter.

Detected Card List
The description of the cards found for reader protocols that support multi-card detection. The format is specific for each protocol. It typically contains all the information gathered during the card discovery. They are only used when several cards are detected, e.g. when a collision event is generated.

Exchange Command
The exchange command is used to exchange data with a card. This command is identical to the HCI command HCI_WR_WCHG_DATA.

4.1.1 Parameter NAL_PAR_READER_CONFIG

The NFC Device sets this parameter to configure the reader detection sequence. The structure of the parameter depends on the reader protocol.

Service	One of the reader service
Code	NAL_PAR_READER_CONFIG
Access	Write only
Storage	Volatile
Size	Depends on the protocol
Default	Depends on the protocol

4.1.2 Parameter NAL_PAR_READER_LIST_CARD

After collision detection and following the type of cards detected, you may collect the cards information by this parameter.

Service	One of the reader service
Code	NAL_PAR_READER_LIST_CARD
Access	Read only
Storage	Volatile
Size	Depends on the protocol
Default	Depends on the protocol

4.1.3 Event NAL_EVT_READER_TARGET_DISCOVERED

The NFC Controller sends this event when a card is detected. The encoding of the event is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	Code of the reader service
1	1	NAL_EVT_READER_TARGET_DISCOVERED
2	N	Description of the card discovered

The nature of the card information depends on the type of card. The details of the event structure are defined in the paragraphs describing each types of card.

4.1.4 Event NAL_EVT_READER_TARGET_COLLISION

The NFC Controller sends this event when a target collision is detected. The encoding of the event is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	Code of the service where the collision is detected.
1	1	NAL_EVT_READER_TARGET_COLLISION
2	2*N	N reader protocol bit fields representing each target detected. N is at least 2.

4.1.5 Command NAL_CMD_READER_XCHG_DATA

The NFC Device sends this command to a reader gate of the NFC Controller to exchange data with an external card. This command is also used for the routing of the UICC command HCI_WR_XCHG_DATA toward the NFC Device.

The encoding of the command is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	The reader service code
1	1	NAL_CMD_READER_XCHG_DATA

2	1	The protocol control byte (see below)
3	N	The protocol parameters and the protocol data (see details in each reader protocol).

The first byte of the command parameters is the Protocol Control Byte (see details in each reader protocol). The encoding of the protocol control byte is the following:

Bit position	**Description**
b7-b5	Depends on the protocol. Set to zero if not specified.
b4	Timeout activation: 1: Application level timeout is given into b3 to b0 0: application level timeout is deactivated
b0-b3	Timeout value. If no answer is received after the timeout expiration, the error NAL_RES_TIMEOUT is returned. The timeout in seconds is calculated by the ISO 14443 formula: 256 × 16 / 13.56MHz) × 2^{value}: **00** 302µs **04** 4.83ms **08** 77.3ms **12** 1.237s **01** 604µs **05** 9.67ms **09** 154ms **13** 2.475s **02** 1.21ms **06** 19.3ms **10** 309ms **14** 4.949s **03** 2.42ms **07** 38.7ms **11** 618ms This value is ignored if b4 is set to zero.

The encoding of the answer is the following:

Offset (in bytes)	**Length (in bytes)**	**Description**
0	1	The service code
1	1	The result code : • NAL_RES_OK for a success • NAL_RES_TIMEOUT if there is no answer after the specified timeout • NAL_RES_PROTOCOL_ERROR A RF protocol error occurred while trying to communicate with RF card
2	N	The payload data of the answer. This data is only present when the result code is a success.

4.2 Reader ISO 14443 A Part 3

4.2.1 Parameter NAL_PAR_READER_CONFIG

This parameter does not exist for the reader ISO 14443 A 3.

4.2.2 Parameter NAL_PAR_READER_LIST_CARD

This parameter does not exist for the reader ISO 14443 A 3.

4.2.3 Event NAL_EVT_READER_TARGET_DISCOVERED

The NFC Controller sends this event when a card is detected. The encoding of the event is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_READER_14_A_3
1	1	NAL_EVT_READER_TARGET_DISCOVERED
2	2	The ATQA parameter defined in **[HCI]**.
4	1	The SAK parameter defined in **[HCI].**
5	4,7 or 10	The UID parameter defined in **[HCI]**.

4.2.4 Command NAL_CMD_READER_XCHG_DATA

The encoding of the command is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_READER_14_A_3
1	1	NAL_CMD_READER_XCHG_DATA
2	1	The protocol control byte
3	N	The raw frame data.

The bits 5, 6 and 7 of the protocol control byte have the following meaning:

bit 7	bit 6	bit 5	Effect

<table>
<tr><th>bit 7</th><th>bit 6</th><th>bit 5</th><th>Effect</th></tr>
<tr><td>0</td><td>0</td><td>1</td><td>

The frame sent on the RF includes a fixed number of bytes with a terminating CRC added by the NFC Controller.
The frame received on the RF includes a fixed number of bytes with a terminating CRC checked by the NFC Controller.

The format of the payload sent and received is the following:
<table><tr><td>The raw frame data (N bytes)</td></tr></table>
The format of the RF frames sent and received is the following:
<table><tr><td>The raw frame data (N bytes)</td><td>2-bytes CRC</td></tr></table>
</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>

The frame sent on the RF includes a fixed number of bytes with a terminating CRC added by the NFC Controller.
The frame received on the RF includes a fixed number of bytes but does not include a terminating CRC.

The format of the payload sent is the following:
<table><tr><td>The raw frame data (N bytes)</td></tr></table>
The format of the RF frames sent is the following:
<table><tr><td>The raw frame data (N bytes)</td><td>2-bytes CRC</td></tr></table>
The format of the RF frames received is the following:
<table><tr><td>The raw frame data (N bytes)</td></tr></table>
The format of the payload received is the following:
<table><tr><td>The raw frame data (N bytes)</td></tr></table>
</td></tr>
</table>

<table>
<tr><th>bit 7</th><th>bit 6</th><th>bit 5</th><th>Effect</th></tr>
<tr><td>1</td><td>0</td><td>0</td><td>

The frame sent on the RF includes a fixed number of bytes with a terminating CRC added by the NFC Controller
The frame received on the RF is either a 4 bits ACK or a NACK, as defined in the [NFC TS 2] (no CRC is sent by the card). If the NFC controller receives an ACK, it sends a response NAL_RES_OK, otherwise it sends a NAL_RES_PROTOCOL_ERROR.

The format of the payload sent is the following:
<table><tr><td>The raw frame data (N bytes)</td></tr></table>
The format of the RF frames sent is the following:
<table><tr><td>The raw frame data (N bytes)</td><td>2-bytes CRC</td></tr></table>
The format of the RF frames received is the following
<table><tr><td>ACK or NACK (4 bits)</td></tr></table>
The received frame is empty.
</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>

The frame sent on the RF includes a fixed number of bits. The last two bytes of the payload contains the number of significant bits in the last bytes. The NFC Controller does not add a CRC to the frame.
The frame received on the RF includes a fixed number of bits. The last two bytes of the payload contains the number of significant bits in the last bytes. The frame does not include a terminating CRC.

This configuration is only supported if the firmware capability bit “Bit oriented exchanges supported for reader ISO 14443-3A” is set.
The format of the payload sent and received is the following:
<table><tr><td>The raw frame data (N bytes)</td><td>V (1 byte)</td><td>C (1 byte)</td></tr></table>
Where C is the number of bits used in the last byte V of the frame. C is in the range [1 – 8]. The first bit to use is the bit 0 of V (LSb).
The format of the RF frames sent and received is the following:
<table><tr><td>The raw frame data (N*8 + C bits)</td></tr></table>
</td></tr>
</table>

bit 7	bit 6	bit 5	Effect
1	1	1	The content of the packet payload is a MIFARE Classic command. This configuration is only supported if the reader capability bit “Mifare Classic” is set. Depending on the command to be sent, the processing of the command requires one or several RF packets exchanges, encrypted using crypto1 algorithm. The following commands are supported: *Authentication using key A:* 0x60 <block number> <key0> …. <key 6> *Authentication using key B:* 0x61 <block number> <key0> …. <key 6> *Read block:* 0x30 <block number> *Write block:* 0xA0 <block number> <data0> … <data15> *Increment:* 0xC1 <block number> <data0> <data1> <data2> <data3> where <data0> <data1> <data2> <data3> represents a 32 bits signed value *Decrement:* 0xC0 <block number> <data0> <data1> <data2> <data3> where <data0> <data1> <data2> <data3> represents a 32 bits signed value *Restore:* 0xC2 <block number> *Transfer:* 0xB0 <block number> The response of the MIFARE command is returned in the NAL_CMD_READER_XCHG_DATA answer.

4.3 Reader ISO 14443 A Part 4

4.3.1 Parameter NAL_PAR_READER_CONFIG

The NFC Device sets this parameter to configure the reader detection sequence. The structure of the parameter is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	The maximum data rate usable for the transaction (DATARATE_MAX)**.** To set the data rate, the NFC controller computes the minimum of the following parameters: - The maximum data rate of the NFC controller. This value is given in the parameter NAL_PAR_FIRMWARE_INFO. - The maximum data rate given by the card - The value of this field (DATARATE_MAX) The default value for this parameter is the maximum value returned in the parameter PAR_FIRMWARE_VERSION. The value is expressed as the maximum divisor supported in **[ISO 14443-4]**. Only the same divisor for both directions (reader to card and card to reader) is supported: - '00' = the maximum divisor supported is 1 - '01' = the maximum divisor supported is 2 - '02' = the maximum divisor supported is 4 - '03' = the maximum divisor supported is 8
1	1	The flag for the CID usage see **[ISO 14443-4]**. 0 is the CID is not used, 1 if the CID is used. The default value is zero.
2	1	The CID value defined in **[ISO 14443-4]**. This value is ignored if the flag for the CID usage is not set. The default value for this parameter is zero.
3	1	The FSD parameter defined in **[ISO 14443-4]**. The FSD code is the size in bytes of the Reader input buffer. To set the Input Buffer Size, the NFC controller computes the minimum of the following parameters: - The maximum Input Buffer size of the NFC controller. This value is given in the parameter NAL_PAR_FIRMWARE_INFO. - The value of this field (FSD parameter). The default value for this parameter is the maximum value returned in the parameter PAR_FIRMWARE_VERSION.

4.3.2 Parameter NAL_PAR_READER_LIST_CARD

This parameter does not exist for the reader ISO 14443 A 4.

4.3.3 Event NAL_EVT_READER_TARGET_DISCOVERED

The NFC Controller sends this event when a card is detected. The encoding of the event is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_READER_14_A_4
1	1	NAL_EVT_READER_TARGET_DISCOVERED
2	2	The ATQA parameter defined in **[HCI]**.
4	1	The SAK parameter defined in **[HCI].**
5	Given by the first byte TL in [1, 253]	The ATS frame specified in **[ISO 14443-4].** Includes TL, T0, TA(1), TB(1), TC(1), and the historical bytes.
6-25	4,7 or 10	The UID parameter defined in **[HCI]**.

4.3.4 Command NAL_CMD_READER_XCHG_DATA

The encoding of the command is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_READER_14_A_4
1	1	NAL_CMD_READER_XCHG_DATA
2	1	Protocol Control Byte
3	1	Optional NAD byte
3 or 4	N	The payload data.

The timeout of the application control byte is an applicative timeout. It is not used a timeout in the ISO 14443 4 protocol but the NFC Controller should return an error NAL_RES_TIMEOUT if this timeout expires before the exchange operation is completed.

The bit 5 of the protocol control byte has the following meaning:

- 1: The NAD byte is present and should be send on the RF.
- 0: The NAD is not present and it should not be sent on the RF.

The format of the frames sent on the RF is the following (see **[ISO 14443 4]** for further details):

Length (in bytes)	Description
1	PCB: The frame type is added by the NFC Controller
1	CID: The Card IDentifier is optional, it may be added by the NFC Controller
1	NAD: The Node ADdress is optional; it may be added by the NFC

	Controller if the bit b5 of the protocol control byte is set.
N	The payload data or part of it
2	EDC: The checksum computed by the NFC Controller

Several frames may be sent on the RF for a single exchange command. The NFC Controller is in charge of the segmentation-reassembly as defined in **[ISO 14443 4]**.

The format of the frames received on the RF is the following:

Length (in bytes)	**Description**
1	PCB: The frame type
1	CID: The optional Card IDentifier
1	NAD: The optional Node ADdress
P	The payload data or part of it
2	EDC: The checksum

Several frames may be received on the RF for a single answer. The NFC Controller is in charge of the reassembly as defined in **[ISO 14443 4]**. The answer message for the exchange command contains the concatenation of the payload data.

4.4 Reader ISO 14443 B Part 3

4.4.1 Parameter NAL_PAR_READER_CONFIG

The NFC Device sets this parameter to configure the reader detection sequence. The structure of the parameter is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	The maximum data rate requested for the communication. To set the data rate, the NFC controller computes the minimum of the following parameters: - The maximum data rate of the NFC controller. This value is given in the parameter NAL_PAR_FIRMWARE_INFO. - The maximum data rate given by the card - The value of this field (DATA_RATE_MAX). The default value for this parameter is the maximum value returned in the parameter PAR_FIRMWARE_VERSION. The value is expressed as the maximum divisor supported in **[ISO 14443-4]**. Only the same divisor for both directions (reader to card and card to reader) is supported: - '00' = the maximum divisor supported is 1 - '01' = the maximum divisor supported is 2 - '02' = the maximum divisor supported is 4 - '03' = the maximum divisor supported is 8
1	1	The Application Family Identifier (AFI) as defined in **[ISO 14443-3]**. The default value for this parameter is 0.
2	1	The flag for the CID usage see **[ISO 14443-3]**. 0 is the CID is not used, 1 if the CID is used. The default value is zero.
3	1	The CID value defined in **[ISO 14443-3]** and **[ISO 14443-4]**. This value is ignored if the flag for the CID usage is not set. The default value for this parameter is zero.
4	1	The FSD parameter defined in **[ISO 14443-3]**. The FSD code the size in bytes of The Reader input buffer. To set the Input Buffer Size, the NFC controller computes the minimum of the following parameters: - The maximum Input Buffer size of the NFC controller. This value is given in the parameter NAL_PAR_FIRMWARE_INFO. - The value of this field (FSD parameter). The default value for this parameter is the maximum value returned in the parameter PAR_FIRMWARE_VERSION.
5	0-245	The HIGHER_LAYER_DATA defined in **[ISO 14443-3]**. The default value for this field is “no data”, i.e. a length of 0.

If the NFC Controller does not support high layer data or supports only a data size inferior to the length provided in parameter, setting the parameter returns the error NAL_RES_FEATURE_NOT_SUPPORTED.

4.4.2 Parameter NAL_PAR_READER_LIST_CARD

This parameter does not exist for the reader ISO 14443 B 3.

4.4.3 Event NAL_EVT_READER_TARGET_DISCOVERED

The NFC Controller sends this event when a card is detected. The encoding of the event is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_READER_14_B_3
1	1	NAL_EVT_READER_TARGET_DISCOVERED
2	11	The ATQB frame from byte 2 to byte 12, as defined in **[ISO 14443-3]**.
13	1-254	The answer to ATTRIB frame, excluding the CRC, as defined in **[ISO 14443-3]** Chapter 7.11: MBLI+CID bytes followed by the optional high layer response data.

4.4.4 Command NAL_CMD_READER_XCHG_DATA

The encoding of the command is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_READER_14_B_3
1	1	NAL_CMD_READER_XCHG_DATA
2	1	The protocol control byte
3	N	The raw frame data.

The format of the frame sent on the RF is the following:

The raw frame data (N bytes)	2-bytes CRC added by the NFC Controller

The format of the frame received on the RF is the following:

The raw frame data (P bytes)	2-bytes CRC checked by the NFC Controller

The P bytes of the raw frame data are returned in the payload of the command answer.

4.5 Reader ISO 14443 B Part 4

4.5.1 Parameter NAL_PAR_READER_CONFIG

This parameter does not exist for the reader ISO 14443 B 4. The Part 4 implementation of the NFC Controller uses the reader configuration of the ISO 14443 B Part 3.

4.5.2 Parameter NAL_PAR_READER_LIST_CARD

This parameter does not exist for the reader ISO 14443 B 4.

4.5.3 Event NAL_EVT_READER_TARGET_DISCOVERED

The NFC Controller sends this event when a card is detected. The encoding of the event is identical to the target discovered event in ISO 14443 B Part 3:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_READER_14_B_4
1	1	NAL_EVT_READER_TARGET_DISCOVERED
…	…	Same as NAL_EVT_READER_TARGET_DISCOVERED for ISO 14443 B Part 3

4.5.4 Command NAL_CMD_READER_XCHG_DATA

The encoding of the command is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_READER_14_B_4
1	1	NAL_CMD_READER_XCHG_DATA
2	1	Protocol Control Byte
3	1	Optional NAD byte
3 or 4	N	The payload data.

The timeout of the application control byte is an applicative timeout. It is not used a timeout in the ISO 14443 4 protocol but the NFC Controller should return an error NAL_RES_TIMEOUT if this timeout expires before the exchange operation is completed.

The bit 5 of the protocol control byte has the following meaning:

- 1: The NAD byte is present and should be send on the RF.
- 0: The NAD is not present and it should not be sent on the RF.

The format of the frames sent on the RF is the following (see **[ISO 14443 4]** for further details):

Length (in bytes)	Description
1	PCB: The frame type is added by the NFC Controller
1	CID: The Card IDentifier is optional, it may be added by the NFC Controller
1	NAD: The Node ADdress is optional; it may be added by the NFC Controller if the bit b5 of the protocol control byte is set.

N	The payload data or part of it
2	EDC: The checksum computed by the NFC Controller

Several frames may be sent on the RF for a single exchange command. The NFC Controller is in charge of the segmentation-reassembly as defined in **[ISO 14443 4]**.

The format of the frames received on the RF is the following:

Length (in bytes)	**Description**
1	PCB: The frame type
1	CID: The optional Card IDentifier
1	NAD: The optional Node ADdress
P	The payload data or part of it
2	EDC: The checksum

Several frames may be received on the RF for a single answer. The NFC Controller is in charge of the reassembly as defined in **[ISO 14443 4]**. The answer message for the exchange command contains the concatenation of the payload data.

4.6 Reader ISO 15693 Part 3

4.6.1 Parameter NAL_PAR_READER_CONFIG

The NFC Device sets this parameter to configure the reader detection sequence. The structure of the parameter is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	The AFI to use for the inventory request as defined in **[ISO 15693-3]**. The default value for this parameter is zero.

4.6.2 Parameter NAL_PAR_READER_LIST_CARD

This parameter does not exist for the reader ISO 15693-3.

4.6.3 Event NAL_EVT_READER_TARGET_DISCOVERED

The NFC Controller sends this event when a card is detected. The encoding of the event is the following:

Offset (in bytes)	Length (in bytes)	Description	
0	1	NAL_SERVICE_READER_15_3	
1	1	NAL_EVT_READER_TARGET_DISCOVERED	
2	1	**bits**	The flags
		b7-b4	RFU = 0
		b3	Extension flag = 0 : no protocol format information = 1: RFU (Protocol format is extended)
		b2-b1	RFU = 0
		b0	Error flag = 0 : no error = 1: Error detected. Error code is in the “Error” field which will be present only if an error occurred. It will provide information about this error. Error code field meaning is detailed in ISO 15693-3 table 7.
3	1	The DSFID is the Data storage format identifier indicating how the data is stored in the card memory. This value is 0x00 if the programming of the DSFID is not supported by the card. See **[ISO 15693-3]**.	
4	8	**Bits**	The UID
		b63 (MSB)-b56	= 0xE0
		b55-b48	IC manufacturer code according to ISO 7816-6:1996/Amd1
		b47-b0 (LSB)	Serial number

4.6.4 Command NAL_CMD_READER_XCHG_DATA

The encoding of the command is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_READER_15_3
1	1	NAL_CMD_READER_XCHG_DATA
2	1	The protocol control byte
3	N	The raw frame data.

The bit 5 of the protocol control byte has the following meaning:

- 1: Send EOF only
- 0: Send nominal data

The format of the frame sent on the RF is the following:

The raw frame data (N bytes)	2-bytes CRC added by the NFC Controller

The format of the frame received on the RF is the following:

The raw frame data (P bytes)	2-bytes CRC checked by the NFC Controller

The P bytes of the raw frame data are returned in the payload of the command answer.

4.7 Reader Type 1

4.7.1 Parameter NAL_PAR_READER_CONFIG

This parameter does not exist for the Type 1 reader.

4.7.2 Parameter NAL_PAR_READER_LIST_CARD

This parameter does not exist for the reader Type 1.

4.7.3 Event NAL_EVT_READER_TARGET_DISCOVERED

The NFC Controller sends this event when a card is detected. The encoding of the event is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_READER_TYPE1
1	1	NAL_EVT_READER_TARGET_DISCOVERED
2	4	The UID of the selected card
6	2	The HR bytes of the selected card
8	2	The ATQA of the selected card.

4.7.4 Command NAL_CMD_READER_XCHG_DATA

The encoding of the command is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_READER_TYPE1
1	1	NAL_CMD_READER_XCHG_DATA
2	1	The protocol control byte
3	N	The raw frame data.

The format of the frame sent on the RF is the following:

The raw frame data (N bytes)

The format of the frame received on the RF is the following:

The raw frame data (P bytes)

The P bytes of the raw frame data are returned in the payload of the command answer.

4.8 Reader FeliCa

4.8.1 Parameter NAL_PAR_READER_CONFIG

The NFC Device sets this parameter to configure the reader detection sequence. The structure of the parameter is the following:

Offset (in bytes)	Length (in bytes)	Description
0	2	The system code bytes to use for the polling command. The default value for this field is 0xFFFF meaning “All system codes”.

The reserved field of the polling command is set to 0x01 to get the system code in the answer.

4.8.2 Parameter NAL_PAR_READER_LIST_CARD

When several FeliCa cards have been detected, this parameter contains the description of the different cards found. The encoding of the parameter is the following:

Offset (in bytes)	Length (in bytes)	Description
0	18 * N	For each detected card - The manufacture ID (ID) : 8 bytes - The manufacture Parameter (PM) : 8 bytes - The System Code (SC) : 2 bytes N is the number of card detected.

4.8.3 Event NAL_EVT_READER_TARGET_DISCOVERED

The NFC Controller sends this event when a card is detected. The encoding of the event is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_READER_FELICA
1	1	NAL_EVT_READER_TARGET_DISCOVERED
2	16	The ID and PM included in the polling response of the selected card.
18	2	The system code returned by the card.

4.8.4 Command NAL_CMD_READER_XCHG_DATA

The encoding of the command is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_READER_FELICA
1	1	NAL_CMD_READER_XCHG_DATA

2	1	The protocol control byte
3	N	The raw frame data.

The format of the frame sent on the RF is the following:

The raw frame data (N bytes)	2-bytes CRC added by the NFC Controller

The format of the frame received on the RF is the following:

The raw frame data (P bytes)	2-bytes CRC checked by the NFC Controller

The P bytes of the raw frame data are returned in the payload of the command answer.

4.9 Reader B Prime

4.9.1 Parameter NAL_PAR_READER_CONFIG

The NFC Device sets this parameter to configure the reader detection sequence. The structure of the parameter is the following:

Offset (in bytes)	Length (in bytes)	Description
0	3 or 4	The APGEN value. The default value for this field is 0x000B7F.

4.9.2 Parameter NAL_PAR_READER_LIST_CARD

This parameter does not exist for the reader B Prime.

4.9.3 Event NAL_EVT_READER_TARGET_DISCOVERED

The NFC Controller sends this event when a card is detected. The encoding of the event is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_READER_B_PRIME
1	1	NAL_EVT_READER_TARGET_DISCOVERED
2	N	The REPGEN value.

4.9.4 Command NAL_CMD_READER_XCHG_DATA

The encoding of the command is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_READER_B_PRIME
1	1	NAL_CMD_READER_XCHG_DATA
2	1	The protocol control byte
3	N	The raw frame data.

The format of the frame sent on the RF is the following:

The raw frame data (N bytes)	2-bytes CRC added by the NFC Controller

The format of the frame received on the RF is the following:

The raw frame data (P bytes)	2-bytes CRC checked by the NFC Controller

The P bytes of the raw frame data are returned in the payload of the command answer.

4.10 Reader Kovio RF barcode ROM ID

4.10.1 Parameter NAL_PAR_READER_CONFIG

This parameter does not exist for the reader Kovio RF barcode ROM ID.

4.10.2 Parameter NAL_PAR_READER_LIST_CARD

This parameter does not exist for the reader Kovio RF barcode ROM ID.

4.10.3 Event NAL_EVT_READER_TARGET_DISCOVERED

The NFC Controller sends this event when a card is detected. The encoding of the event is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_READER_KOVIO
1	1	NAL_EVT_READER_TARGET_DISCOVERED
2	N	Kovio RF barcode ROM ID data.

4.10.4 Command NAL_CMD_READER_XCHG_DATA

This command does not exist for the reader Kovio RF barcode ROM ID.

5 Card Protocols

The NFC Device uses the card protocols to emulate a card. An external reader may detect and select the emulated card. The NFC Device may emulate cards of type ISO 14443 Level 4 for the types A or B.

5.1.1 Common Principles

The NFC Controller is in charge of the anti-collision procedure. The card parameters for the anti-collision are set with a parameter NAL_PAR_CARD_CONFIG. The content of the parameter depends on the card type.

When an external reader selects the emulated card after the anti-collision procedure, the event NAL_EVT_CARD_SELECTED is sent to the NFC Device. The content of the event data payload depends on the card type.

When the external reader sends a command to the emulated card, the NFC Device receives an event NAL_EVT_CARD_SEND_DATA. The command is stored in the data payload of the event.

The NFC Device sends the answer to the command with an event NAL_EVT_CARD_SEND_DATA. The answer data is stored in the data payload of the event.

If the reader deselects the card or if the field is shutdown for any reason, the NFC Device receives an event NAL_EVT_CARD_END_OF_TRANSACTION.

5.1.2 Parameter NAL_PAR_CARD_CONFIG

The parameter NAL_PAR_CARD_CONFIG contains the card configuration used for the anti-collision. The structure of the parameter depends on the card protocol. See each card protocol for the description of the content.

Service	One of the card service
Code	NAL_PAR_CARD_CONFIG
Access	Write only
Storage	Volatile
Size	Depends on the protocol
Default	Depends on the protocol

5.1.3 Event NAL_EVT_CARD_SELECTED

The NFC Controller sends this event when an external reader selects an emulated card. See each card type for the data payload of the event.

5.1.4 Event NAL_EVT_CARD_SEND_DATA

This event is sent by the NFC Controller or by the NFC Device to exchange data between the external reader and the card emulation. The data payload contains the data exchanged with the external reader following the card protocol. The format of this data depends on the card type.

The syntax of this event is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	The card emulation service.
1	1	NAL_EVT_CARD_SEND_DATA
2	N	The data to send. N may be zero.

5.1.5 Event NAL_EVT_CARD_END_OF_TRANSACTION

The NFC Controller sends this event when a card emulation is selected and one of the following events occurs:

- The external reader deselects the card with a de-selection command.
- The RF field is shutdown by the external reader.
- The RF field is lost for any reason.

After this event, the card emulation is no longer selected. The NFC Device shall not send the event again until a new event NAL_EVT_CARD_SELECTED is sent. For example, if the event NAL_EVT_CARD_END_OF_TRANSACTION is sent because the card is de-selected by the reader, the NFC Device will not send the event again when the RF field is shutdown.

The syntax of this event is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	The card emulation service.
1	1	NAL_EVT_CARD_END_OF_TRANSACTION
2	1	0 if the reader sent a de-selection command, 1 for any other reason.

5.1.6 Event NAL_EVT_RF_FIELD

This event is sent when the NFC Controller detects the presence of an external RF field or when an external RF field is switched off.

The syntax of this event is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_ADMIN
1	1	NAL_EVT_RF_FIELD
2	1	1 if the external RF field is ON, 0 if the RF field is OFF

5.2 Card Emulation ISO 14443 A Part 4

The card emulation ISO 14443 A Part 4 is implemented by the service NAL_SERVICE_CARD_14_A_4.

The data exchanged with the event NAL_EVT_CARD_SEND_DATA is transferred with the T=CL protocol (See **[ISO 14443-4]**) The NFC Controller is in charge of the T=CL framing, of fragmentation-reassembly, of the repetition upon error and of the lower layer protocol including the CRC.

5.2.1 Parameter NAL_PAR_CARD_CONFIG

The structure of this parameter is the following:

Offset (in bytes)	Length (in bytes)	Name	Description
0	10	UID	See **[ISO 14443-3]** Specification
10	1	UID_LEN	The UID length in bytes. This value may be 0, 4, 7 or 10. If set to zero, the NFC Controller generates a random UID of 4 bytes, the first byte is set to 0x08.
11	1	ATQA MSB	The MSB of the ATQA as defined in **[ISO 14443-3].**
12	1	T0	See **[ISO 14443-4]** Specification
13	1	TA	“ (optional)
-	1	TB	“ (optional)
-	1	TC	“ (optional)
-	0-252	APPLICATION_DATA	“

The SAK bits are set to 1 to signal the compliance with the part 4 of the protocol. The NFC Controller may modify part of the values in the bytes T0, TA, TB and TC depending of its capabilities.

When the NFC Controller is booted or after a reset, the default values are the following:

- The value of UID_LEN is set to zero (random UID).
- The ATQA MSB is set to 0.
- The value of T0 is 0x72 because TA(1), TB(1) and TC(1) are present, also the FSCI code is set to its default value (10b) equivalent to 32 bytes. See **[ISO 14443-4]** Specification. The FSCI codes the maximum frame size.
- The value of TA is 0x00 because the bit rate in both directions is set to default value and the divisor may be different in both direction. See **[ISO 14443-4]** Specification.
- The value of TB is set to 0x40 because the FWI (0100b) and SFGT (0000b) are set to default value regarding **[ISO 14443-4]** Specification.
- The value of TC is 0x00 because NAD and CID are not supported by default.
- The APPLICATION_DATA is optional and is not carried by default.

If the NFC Controller does not support application data or supports only a data size inferior to the length provided in parameter, setting the parameter returns the error NAL_RES_FEATURE_NOT_SUPPORTED.

5.2.2 Event NAL_EVT_CARD_SELECTED

The syntax of this event is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_CARD_14_A_4
1	1	NAL_EVT_CARD_SELECTED
2	1	The data rate used for the communication with the remote reader
3	1	The CID Parameter defined in [**ISO 14443-4].**
4	10	The UID value. See **[ISO 14443-3]** Specification.
14	1	The UID length in bytes. This value may be 4, 7 or 10.

5.3 Card Emulation ISO 14443 B Part 4

The card emulation ISO 14443 B Part 4 is implemented by the service NAL_SERVICE_CARD_14_B_4.

The data exchanged with the event NAL_EVT_CARD_SEND_DATA is transferred with the T=CL protocol (See **[ISO 14443-4]**). The NFC Controller is in charge of the T=CL framing, of fragmentation-reassembly, of the repetition upon error and of the lower layer protocol including the CRC.

5.3.1 Parameter NAL_PAR_CARD_CONFIG

The structure of this parameter is the following:

Offset (in bytes)	Length (in bytes)	Description
0	11	The ATQB defined in **[ISO 14443-4]** Specification The NFC Controller may change the following fields of the ATQB: - If the value in the PUPI is null, the NFC controller shall generate a random PUPI - The protocol bytes may be updated as following: ✓ The bit rate capability field may be updated if the value in the field is not supported by the NFC controller ✓ The maximum frame size field may be updated if the value in the field is not supported by the NFC controller ✓ The FWI and FO bits may be changed
11	0-253	The higher layer response data defined in **[ISO 14443-4]** Specification (Chapter 7.11 "Response to ATTRIB Command").

If the NFC Controller does not support high layer response data or supports only a data size inferior to the length provided in parameter, setting the parameter returns the error NAL_RES_FEATURE_NOT_SUPPORTED.

When the NFC Controller is booted or after a reset, the default values are the following:

- The value of ATQB (11bytes) excluding the header byte and the two CRC bytes is : 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x21, 0x40, because:
  - The PUPI is set to zero (random PUPI)
  - The application data (AFI, CRC_B (AID) and number of application) are set to zero
  - The bit rate is set to default (106kbts/s)
  - The maximum frame size is 32 bytes
  - The FWI is set to ~ 4,8 ms
  - The NAD and CID are not supported
- There is no higher layer response data.

5.3.2 Event NAL_EVT_CARD_SELECTED

The syntax of this event is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_CARD_14_B_4
1	1	NAL_EVT_CARD_SELECTED
2	1	The Application Family Identifier (AFI) as defined in **[ISO 14443-3]**.
3	1	The data rate used for the communication
4	1	The CID Parameter defined in [**ISO 14443-3].**
5	1	The Frame Size parameter defined in **[ISO 14443-3].**
6	4	The PUPI sent in the ATQB
10	0-245	The HIGHER_LAYER_DATA defined in **[ISO 14443-3]**.

6 Peer 2 Peer Protocol

The Peer 2 Peer protocol (P2P) creates a symmetric communication channel between two NFC Devices. The symmetry is created by the upper layer of the protocol. The lower layer is asymmetric and is based on two modes: the Initiator mode emitting the RF field and the Target mode receiving the RF field. Both devices may include the Target mode and the initiator mode in their detection sequence. Due to the time slot allocated to each mode, the Initiator mode of one of the device should discover the target mode of the other.

The following figure is an overview of the P2P communication between 2 NFC HAL compliant devices:

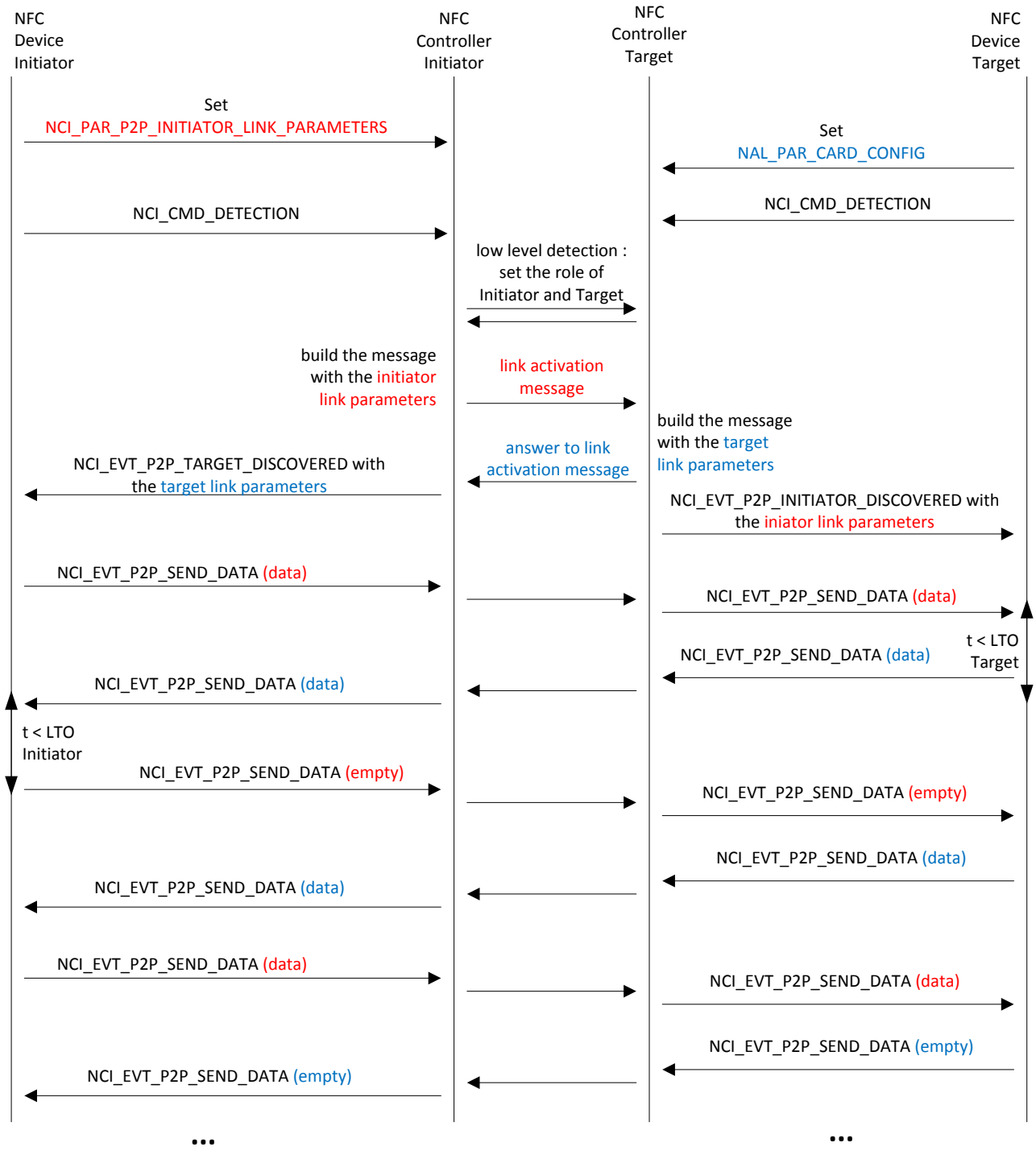

Figure 3 – Overview of the P2P Communication

6.1 Initiator Service

The behavior of the Peer 2 Peer Initiator service is similar to the behavior of the reader protocols.

When the Peer 2 Peer Initiator mode is included in the detection sequence, the NFC Controller should try to detect the Peer 2 Peer targets. When a P2P target is detected at the low level (See **[LLCP]**), the Initiator sends a link activation message. The structure of this message is described in **[LLCP]**. The NFC controller copies in the link activation message

the link activation parameters included in the parameter NAL_PAR_P2P_INITIATOR_LINK_PARAMETERS.

Upon reception of this message the Target sends back an answer message including the target link parameters. The NFC Controller checks the consistency of this message as specified in **[LLCP]**. The NFC Controller does not check the target link parameters. If the answer message is not correct, the NFC Controller resumes the detection sequence. If the answer message is correct, the NFC Controller sends an event NAL_EVT_P2P_TARGET_DISCOVERED to the NFC Device. This event contains the target link parameters. It is up to the NFC Device to decide if the target link parameters are valid or not. If the NFC Device decides that the target link parameters are not valid, it may just resume the detection sequence.

The following exchanges of messages are performed with the event NAL_EVT_P2P_SEND_DATA. The NFC Device is responsible of the content of the messages and of the respect of the P2P timeouts.

The NFC Device shuts the P2P link down by resuming the detection sequence.

6.1.1 Parameter NAL_PAR_P2P_INITIATOR_LINK_PARAMETERS

The content of this parameter is included at the end of the link. This parameter includes the P2P magic word and the configuration transferred to the target. See **[LLCP]** for the details on the content of the parameter.

Service	NAL_SERVICE_P2P_INITIATOR
Code	NAL_PAR_P2P_INITIATOR_LINK_PARAMETERS
Access	Write only
Storage	Volatile
Size	0 to N
Default	Empty parameter, i.e. size is set to zero

6.1.2 Event NAL_EVT_P2P_TARGET_DISCOVERED

The NFC Controller sends this event when a target is detected. The encoding of the event is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_P2P_INITIATOR
1	1	NAL_EVT_P2P_TARGET_DISCOVERED
2	4	The link baud rate
6	N	The link configuration parameters sent by the target in the answer to the link activation message. This value may be empty, i.e. N=0.

The NFC Controller does not check the consistency of the link configuration parameters returned by the Target. See **[LLCP]** for the details on the content of the link configuration parameters.

6.1.3 Event NAL_EVT_P2P_SEND_DATA

The event send data is used to exchange data between the NFC Device and the remote Target. This event is similar to the event NAL_EVT_CARD_SEND_DATA of the card protocols.

The data payload of the event to send may be empty if the NFC Device has no data to send. The data payload of the event received may be empty if no data is received by the NFC Controller.

To avoid timeout problems, the NFC controller should answer to the event NAL_EVT_CARD_SEND_DATA as soon as it is ready to receive another event NAL_EVT_CARD_SEND_DATA, regardless of the availability of data from the Target.

6.2 Target Service

The behavior of the Peer 2 Peer Target service is similar to the behavior of the card protocols.

When the P2P Target service is included in the detection sequence, the NFC Controller may be detected by an external P2P initiator. After the low level detection, see **[LLCP]**, the Initiator should send a link activation message. The NFC Controller should check the consistency of this activation message. The NFC Controller does not check the Initiator link configuration parameters. The structure of this message and the checking procedure is described in **[LLCP]**. If the message is not valid, the NFC Controller resumes the detection sequence.

If the message is valid, the NFC Controller responds to the Initiator with a message containing the Target link configuration parameters included in the parameter NAL_PAR_CARD_CONFIG.

When the answer message is sent, the NFC Controller sends to the NFC Device an event NAL_EVT_P2P_INITIATOR_DISCOVERED including the link parameters sent by the Initiator.

After this phase of link connection, the communication with the Initiator is based on the data events like the other card protocols. The content of the data event received from the Initiator is checked by the NFC Device. The NFC Device builds the messages sent to the Initiator and checks the P2P timeouts.

6.2.1 Parameter NAL_PAR_CARD_CONFIG

The content of this parameter is included in the answer to the link activation message. This parameter includes the P2P magic word and the link configuration transferred to the initiator. The content of this message is described in **[LLCP]**.

Service	NAL_SERVICE_P2P_TARGET
Code	NAL_PAR_CARD_CONFIG
Access	Write only
Storage	Volatile
Size	6 to 6+N bytes
Default	Empty parameter, i.e. size is set to 6

The structure of the parameter is the following:

Offset (in bytes)	Length (in bytes)	Description
0	4	The repetition timeout of the target in ms.
4	1	The Type A flag. If the value is set to zero and the P2P connection is on the target side, the Type F protocol is used exclusively. Otherwise, the P2P connection may use Type A or Type F protocol. The default value is 1.
5	1	The active mode flag. The active mode is used if the following conditions are all true: • the value of the flag is not zero, • the NFC Controller supports the active mode, and • the remote device is using the active mode. Otherwise the passive mode is used. The default value is 1.
6	N	Payload of the link activation message. The default value is empty (i.e. N=0).

6.2.2 Event NAL_EVT_P2P_INITIATOR_DISCOVERED

The NFC Controller sends this event when an Initiator is detected. The encoding of the event is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_P2P_TARGET
1	1	NAL_EVT_P2P_INITIATOR_DISCOVERED
2	4	The link baud rate
6	N	The link configuration parameters sent by the Initiator in the link activation message. This value may be empty, i.e. N=0.

The NFC Controller does not check the consistency of the link configuration parameters provided by the Initiator. See **[LLCP]** for the details on the content of the link configuration parameters.

6.2.3 Event NAL_EVT_P2P_SEND_DATA

The event send data is used to exchange data between the NFC Device and the remote Initiator. This event is similar to the event NAL_EVT_CARD_SEND_DATA of the card protocols.

7 SWP Management

This chapter describes the management of the UICC by the NFC Device. The UICC is a removable element; it may or may not be present. Even if a UICC is present in the NFC Device, it may not implement the SWP interface.

The next figure illustrates the location and the link with the UICC.

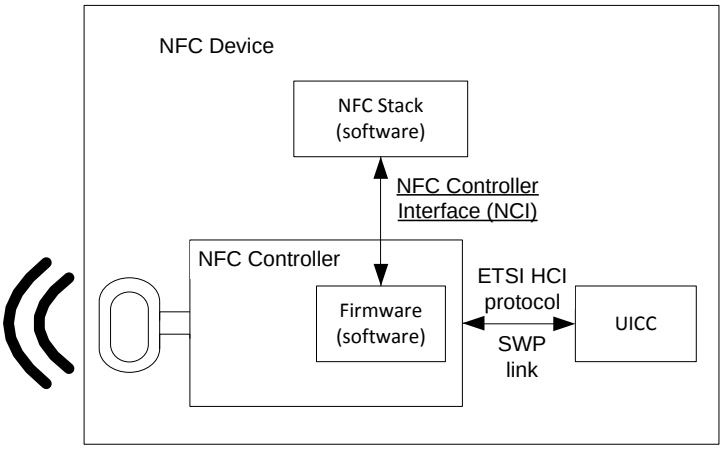

Figure 4 – UICC Connection

7.1 Variables and Constants

This section describes the variables and constant values used with the UICC.

7.1.1 Variable VAR_UICC_DETECTION-C

This persistent variable is a protocol bit field representing the card protocols requested by the UICC for the detection sequence.

- If a bit is set, the corresponding protocol is requested for the detection sequence.
- If a bit is not set, the corresponding protocol is not requested for the detection sequence.

7.2 UICC Boot and Status

NFC devices are required to dynamically detect the presence of a UICC with a status of the SWP connection.

7.2.1 Command NAL_CMD_UICC_START_SWP

This command requests the NFC Controller to start the SWP communication with the UICC. The NFC Controller may decide to shutdown the SWP line to reduce the power consumption of the system. This decision is taken directly by the NFC Controller after a period of inactivity on the SWP line. See the HCI specification “ETSI TS 102 622 V7.5.0 (2009-06)” for further details on the SWP deactivation. When the SWP becomes inactive, the UICC is not powered down. The UICC may decide to communicate again with the NFC Controller. To do so, a new command “ACTIVATE” is added to the STK. See section 6.6.40 of the specification ETSI TS 102 223 V8.0.0 (2008-07) “Card Application Toolkit (CAT) (Release 8)” to get further information on the ACTIVATE command. When the NFC Device receives the ACTIVATE command, the command NAL_CMD_UICC_START_SWP is used to force the NFC Controller to re-activate the SWP line.

If the NFC Controller does not support SWP, if there is no UICC or if the SWP communication is already started and active, the command NAL_CMD_UICC_START_SWP has no effect.

The encoding of the command is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_UICC
1	1	NAL_CMD_UICC_START_SWP

There is no error code for this command. The encoding of the answer is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_UICC
1	1	NAL_RES_OK

7.2.2 Parameter NAL_PAR_UICC_SWP

The value of this parameter is dynamically updated with the current status of the SWP connection.

Service	NAL_SERVICE_UICC
Code	NAL_PAR_UICC_SWP
Access	Read only
Storage	Volatile
Size	0x01 byte

The possible values are the following:

Constant	Value	Description
NAL_UICC_SWP_NO_SE	0x00	No SE is detected on SWP
NAL_UICC_SWP_BOOTING	0x01	The boot process of the SWP protocol is pending
NAL_UICC_SWP_ERROR	0x02	An error is detected on the SWP line
NAL_UICC_SWP_ACTIVE	0x03	The SE is active
NAL_UICC_SWP_DOWN	0x04	The SWP line is down

7.3 UICC Access Policy

The NFC Device defines the access policy of the UICC for the reader protocols and the card protocols.

Card Access Policy
The access policy for the card protocols is defined by the parameters NAL_PAR_POLICY and NAL_PAR_PERSISTENT_POLICY.

The UICC must have been configured to be powered in BATT-OFF mode to be able to perform card emulation when in BATT-OFF mode. In this situation, the card protocols granted to the UICC are those specified in the NAL_PAR_PERSISTENT_POLICY parameter.

When in BATT-ON mode, the card protocols granted to the UICC are those specified in the NAL_PAR_POLICY parameter.

Reader Access Policy

The access policy for the reader protocols is implicitly defined by the behavior of the NFC Device. The NFC Device intercepts every card detection for the UICC and may allow or hide the card to the UICC (See the section “Reader Protocols” bellow).

The reader modes are not available for the UICC in BATT-OFF mode.

7.4 Reader Protocols

The usage of the reader protocols from the UICC is supervised by the NFC Device. The UICC uses the HCI commands to access the reader protocols (See ETSI HCI Specification **[HCI]**).

Power-On Mode
When the NFC Device is active, i.e. in Power-On Mode, almost every HCI messages exchanged between the UICC and the NFC Controller on the Reader pipes are interpreted by the NFC Controller. The HCI command HCI_WR_XCHG_DATA may be interpreted by the NFC Controller or routed to the NFC Device. The routing of HCI_WR_XCHG_DATA depends on the routing value set by the command NAL_CMD_COM_TRANSFER described below.

Battery-Off Mode and Battery-Low
In Battery-Off mode (also called “power by the field” mode) and in Battery-Low mode, the reader functions are not accessible by the UICC.

Example of sequence
In this scenario, the UICC and the NFC Device application share the reader B. 2 reader clients are registered in the NFC Device. The client A has a higher priority than the UICC; the client B has a lower priority.

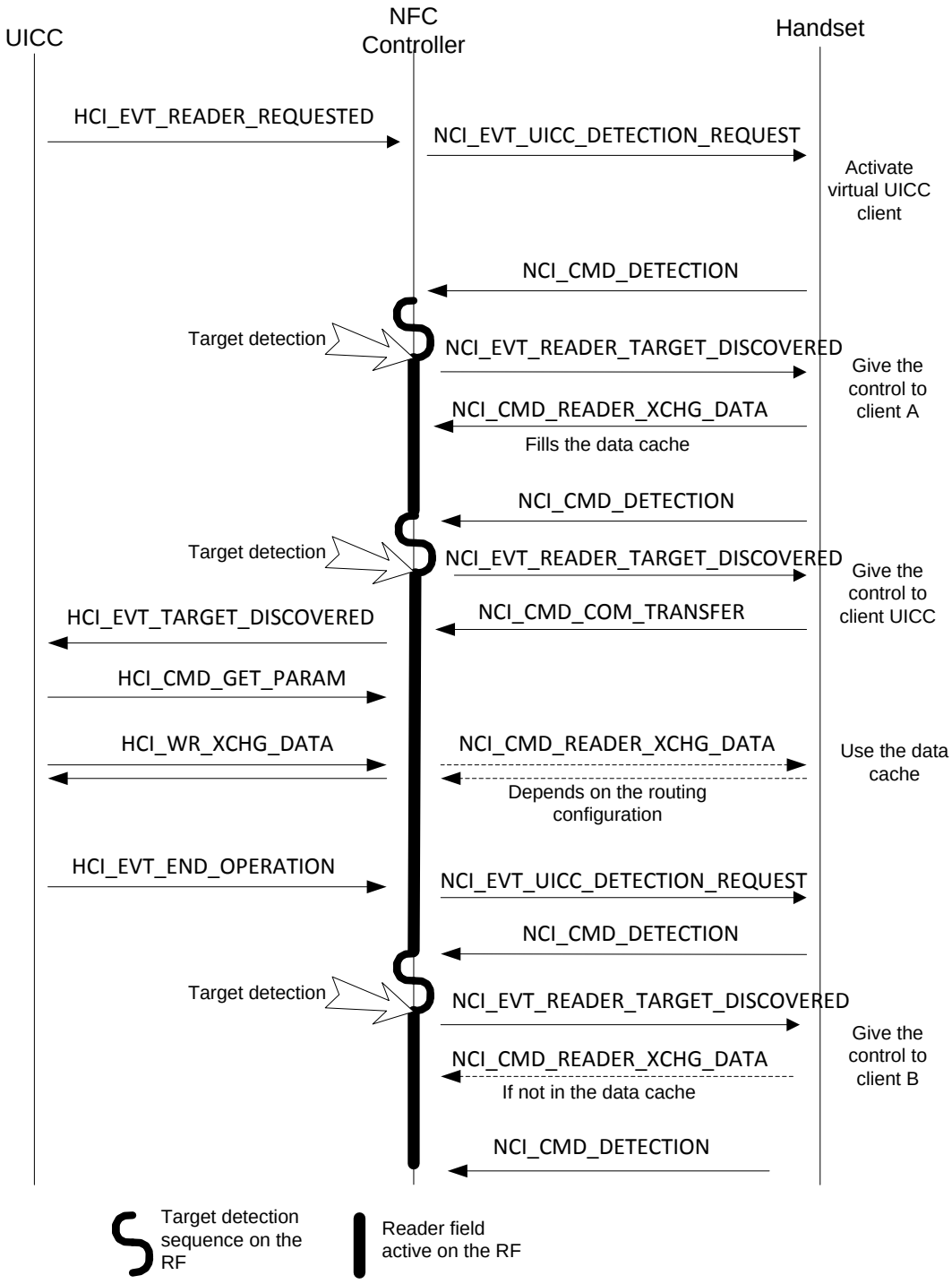

Figure 5 – Example of reader protocol sharing with the UICC

Boot Sequence

In power-on mode, the NFC Controller executes the UICC reader protocol messages. If a reader detection is requested by the UICC, the detection sequence is not executed by the NFC Controller until the command Get Parameter NAL_PAR_UICC_READER_PROTOCOLS is called for the first time.

After the first occurrence of the command Get Parameter NAL_PAR_UICC_READER_PROTOCOLS, the UICC events

HCI_EVT_READER_REQUESTED and HCI_EVT_END_OPERATION are transferred to the NFC Device with the event NAL_EVT_UICC_DETECTION_REQUEST described below.

Forcing the UICC to stop a reader protocol
When the reader communication is transferred to the UICC with the command NAL_CMD_COM_TRANSFER, the NFC Device may stop the reader communication at any time by calling the command NAL_CMD_DETECTION.

7.4.1 Event NAL_EVT_UICC_DETECTION_REQUEST

The NFC controller sends this event to the NFC Device whenever an event HCI_EVT_READER_REQUESTED is sent by the UICC. The NFC Controller does not trigger any detection sequence when receiving the event HCI_EVT_READER_REQUESTED from the UICC.

The NFC controller sends this event to the NFC Device whenever an event HCI_EVT_END_OPERATION is sent by the UICC. The NFC Controller does not stop the detection sequence when receiving the event HCI_EVT_END_OPERATION from the UICC. The event is also sent if the UICC abruptly stops the reader communication with a command HCI_ANY_CLOSE.

The encoding of the event is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_UICC
1	1	NAL_EVT_UICC_DETECTION_REQUEST
2	N	Identical to the value of the parameter NAL_PAR_UICC_READER_PROTOCOLS
0	1	The UICC request detection flag: • 0 if the reader detection is not requested by the UICC • 1 if the reader detection is requested by the UICC
1	2	A protocol bit field representing the reader protocols opened by the UICC: • If a bit is set, the corresponding protocol is opened by the UICC. • If a bit is not set, the corresponding protocol is not opened by the UICC.
Reader A-4		
3	1	The value of the DATARATE_MAX parameter in the HCI registry
Reader B-4		
4	1	The value of the AFI parameter in HCI registry
5	0-20	The value of the HIGHER_LAYER_DATA parameter in the HCI registry. The size of the parameter is limited to 20 bytes

7.4.2 Command NAL_CMD_COM_TRANSFER

The NFC Device sends this command to the NFC controller to transfer the communication of an active reader protocol to the UICC. If the corresponding reader protocol is not opened by the UICC or if this reader protocol is not active, the NFC controller returns an error. This control is mandatory to avoid error due to concurrent closing from the UICC.

When this command is received, the NFC Controller sends the event HCI_EVT_TARGET_DISCOVERED to the UICC. Then the commands HCI_WR_XCHG_DATA are accepted from the UICC for the reader protocol. The HCI_WR_XCHG_DATA commands are directly executed by the NFC controller or they are routed to the NFC Device with the command NAL_WR_XCHG_DATA. A flag of the command NAL_CMD_COM_TRANSFER activates the routing.

Before receiving the command NAL_CMD_COM_TRANSFER, the event HCI_EVT_TARGET_DISCOVERED is not sent to the UICC and the UICC command HCI_WR_XCHG_DATA returns an error indicating the absence of card.

The encoding of the command is the following:

Offset (in bytes)	**Length (in bytes)**	**Description**
0	1	Code of the reader service
1	1	NAL_CMD_COM_TRANSFER
2	1	The routing flag for the HCI_WR_XCHGDATA commands is: • 0 if the commands HCI_WR_XCHGDATA should be executed by the NFC Controller; • 1 if the commands HCI_WR_XCHGDATA should be routed to the NFC Device.

7.4.3 Parameter NAL_PAR_UICC_READER_PROTOCOLS

This parameter describes the reader protocols open by the UICC with the HCI command HCI_ANY_OPEN_PIPE.

The firmware initializes this value during the boot procedure and updates the value each time the UICC opens a service.

Service	NAL_SERVICE_UICC
Code	NAL_PAR_UICC_READER_PROTOCOLS
Access	Read only
Storage	Volatile
Size	0x05 to 0x19 bytes
Default	When the firmware is downloaded, every reader protocol is closed.

The mapping of the parameter is the following:

Offset (bytes)	Length (bytes)	Name	Value
0	1	DETECTION	The UICC request detection flag: • 0 if the reader detection is not requested by the UICC • 1 if the reader detection is requested by the UICC
1	2	OPEN_PROTOCOLS	A protocol bit field representing the reader protocols opened by the UICC: • If a bit is set, the corresponding protocol is opened by the UICC. • If a bit is not set, the corresponding protocol is not opened by the UICC.
Reader A-4			
3	1	DATARATE_MAX	The value of the DATARATE_MAX parameter in the HCI registry
Reader B-4			
4	1	AFI	The value of the AFI parameter in HCI registry
5	0-20	HIGHER_LAYER_ DATA	The value of the HIGHER_LAYER_DATA parameter in the HCI registry. The size of the parameter is limited to 20 bytes

7.5 Card Emulation

This section defines the behavior of the NFC Controller when the UICC uses the card emulation protocol. Creating a HCI pipe for card emulation has no effect. Destroying a HCI pipe of card emulation closes the service if it is open.

The creation of a card emulation pipe and the opening of the service are limited to the implemented protocols defined by CST_CAPABILITIES-C.

7.5.1 Parameter NAL_PAR_UICC_CARD_PROTOCOLS

This parameter describes the card protocols open by the UICC with the HCI command HCI_ANY_OPEN_PIPE.

The firmware initializes this value during the boot procedure and updates the value each time a service is opened by the UICC.

Service	NAL_SERVICE_UICC
Code	NAL_PAR_UICC_CARD_PROTOCOLS
Access	Read only
Storage	Persistent
Size	0x46 bytes
Default	When the firmware is downloaded, every card protocol is closed.

The mapping of the parameter is the following:

Offset (bytes)	**Length (bytes)**	**Name**	**Value**
0	2	OPEN_PROTOCOLS	A protocol bit field representing the protocols opened by the UICC: • If a bit is set, the corresponding protocol is opened by the UICC. • If a bit is not set, the corresponding protocol is not opened by the UICC.
Card A-4			
2	1	MODE	See **[HCI]** Specification
3	10	UID	“
13	1	UID_LEN	“
14	1	SAK	“
15	2	ATQA	“
17	20	APPLICATION_DATA	“
37	1	FWI_SFGI	“
38	1	CID_SUPPORT	“
39	1	DATA_RATE_MAX	“
Card B-4			
40	1	MODE	See **[HCI]** Specification
41	4	PUPI	“
45	1	AFI	“
46	4	ATQB	“
50	20	HIGHER_LAYER_RESPONSE	“

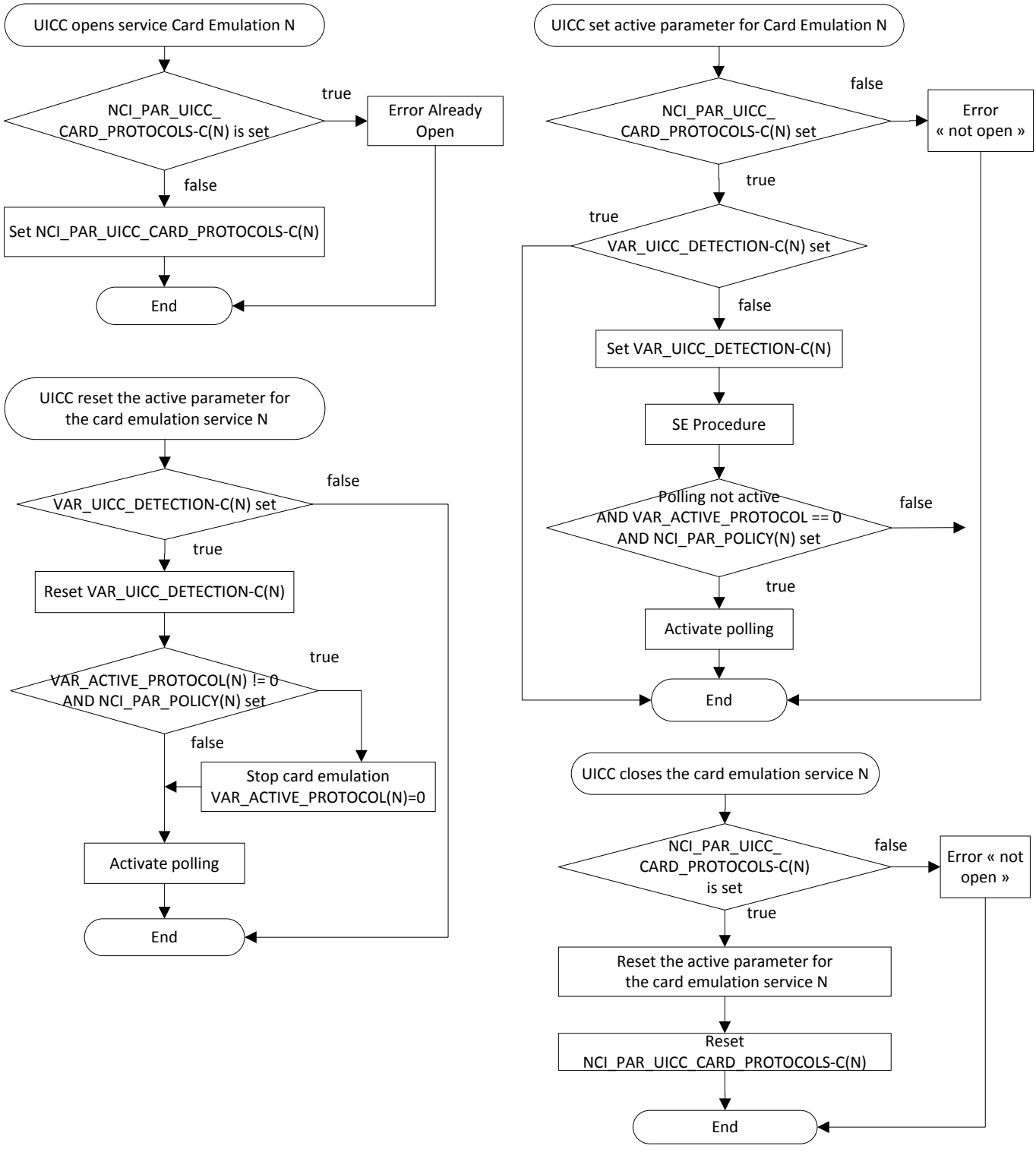

Figure 6 –Card Emulation for the UICC

7.6 Connectivity Messages

The UICC may send connectivity messages to the NFC Device. The connectivity messages are sent through the SWP link to the NFC controller. They are defined as HCI events sent on a specific connectivity gate of the NFC Device virtual host. The syntax and the meaning of the connectivity messages are defined in the **[HCI]** specification.

7.6.1 Event NAL_EVT_UICC_CONNECTIVITY

The syntax of the connectivity events is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_UICC
1	1	NAL_EVT_UICC_CONNECTIVITY
2	1	The message code defined in **[HCI]**.
3	N	The payload of the connectivity message defined in **[HCI]**. This value may be empty, i.e. N=0.

8 Secure Element

The Secure Element (SE) is an optional IC connected to the NFC Controller. The NFC Controller may be connected to several SE. A SE may be included in the NFC Controller package.

8.1 Secure Element Information

For each Secure Element, the parameter NAL_PAR_HARDWARE_INFO includes the identification of the Secure Element in a string, a field describing the capabilities of the Secure Element, the description of the Host Interface and the description of the RF Interface.

SE Capabilities

This value describes the capabilities of the SE. The value is a bit field of two bytes with the following mapping:

Bit position	Description
b0	If set, the SE may send end-of-transaction notifications.
b1-b15	RFU - SBZ

If there is no SE, this value is set to zero.

Host Interface

This value identifies the reader protocol required to communicate with the SE when the switch is set in host-interface position. The mapping of this value is a reader protocol bit field with one bit set for the right reader protocol. If the NFC Device cannot communicate with the SE or if there is no SE, this value is set to zero.

RF Interface

This value identifies the reader protocol required to communicate with the SE when the switch is set in host-interface position. The mapping of this value is a reader protocol bit field with one bit set for the right reader protocol. If the NFC Device cannot communicate with the SE or if there is no SE, this value is set to zero.

8.2 Position of the SE Switch

The position of the SE switch is defined by the SE bits of the parameters NAL_PAR_POLICY and NAL_PAR_PERSISTENT_POLICY. If no SE is present, the SE bits shall be set to the OFF position.

The mapping of the parameter is the following:

Offset	Size	Field	Description
0	2	UICC Card Policy	The structure of this bit field follows the mapping of the card protocols defining the card protocols allowed for the UICC.
2	2	UICC Reader Policy	The structure of this bit field follows the mapping of the reader protocols defining the reader protocols allowed for the UICC.
4	2	Policy Flags	The policy flags are described below
6	2	SE Filter	The structure of this bit field follows the mapping of the card protocols.

If a SE filter value is not supported by the SE, setting the volatile policy or the persistent policy with this value will return the error NAL_RES_FEATURE_NOT_SUPPORTED.

The mapping of the policy flags is the following:

Position (bits)	15-8	7-6	5-4	3-2	1	0
Field	RFU-SBZ	BATT-OFF SW	SE-ID	SE-SW	CL	RL

The following figure describes the effects of the different configuration:

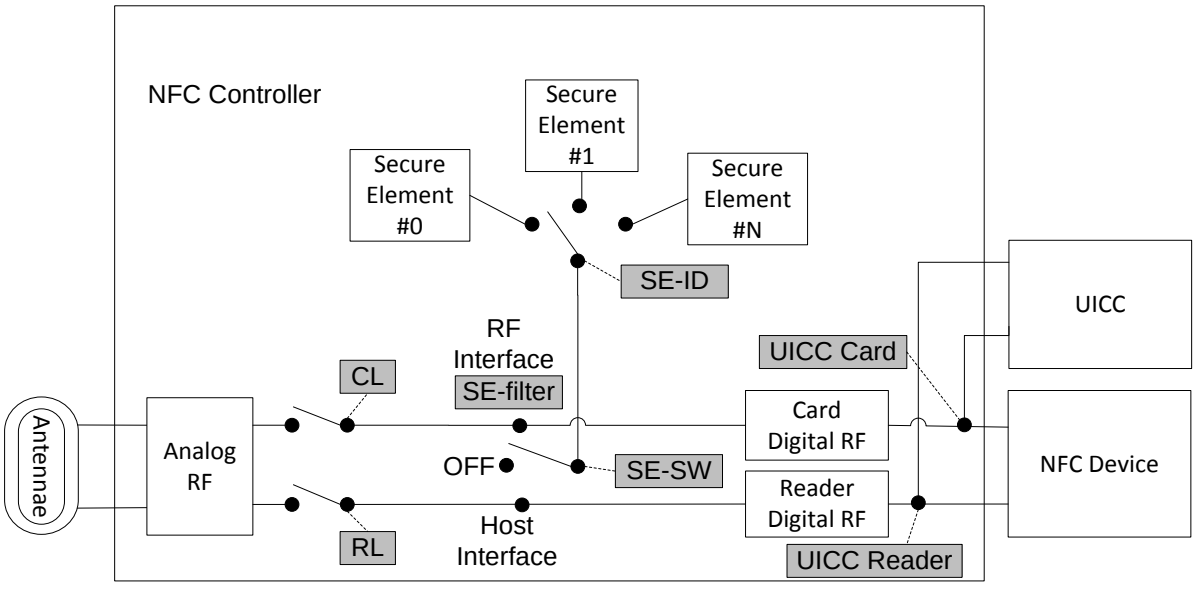

Figure 7 – Example of SE included in the NFC Controller

The effects of the SE bits are the following:

Position	SE -SW	SE-ID	SE filter	CL	RL	UICC Card	UICC Reader	Device Card	Device reader	Description
OFF	0	0	x	x	x	C 2	x	C 3	X	• All SE(s) are in OFF position • C2 and C3 shall not contain the same protocols. When in BATT-ON mode, or when the UICC has been configured to be powered in BATT-OFF mode: • The UICC may perform card emulation with the C2 protocols if CL is set. Exclusively in BATT-ON mode: • The NFC Device may perform card emulation with the C3 protocols if CL is set. • The UICC or the NFC Device may listen to cards if RL is set.
RF Interface	1	N	C 1	x	x	C 2	x	C 3	X	• The switch of the SE #N is on the RF Interface position. • The other SE(s) are OFF. • C1, C2 and C3 shall not contain the same protocols. When in BATT-ON mode, or when the SE has been configured to be powered in BATT-OFF mode: • The SE #N can perform a card emulation with any of the specified C1 protocols. Exclusively in BATT-ON mode: • The UICC or the NFC Device may listen to cards if RL is set (only in BATT-ON mode)

Position	SE -SW	SE-ID	SE filter	CL	RL	UICC Card	UICC Reader	Device Card	Device reader	Description
Host Interface	2 or 3	N	C 1	x	x	C 2	x	C 3	X	Exclusively in BATT-ON mode: • The switch of the SE #N is on the host Interface position. The NFC Device communicates with the SE #N. • The other SE(s) are OFF. • C1, C2 and C3 shall not contain the same protocols. • The value 2 forces the switch in Host Interface position even if a card emulation is being performed by the SE. • The value 3 waits for the end of a card emulation performed by the SE before switching to Host Interface.

The position “Host Interface” (2 or 3) is invalid for the persistent parameter NAL_PAR_PERSISTENT_POLICY. The “Host Interface” position (2 or 3) should only be used with the volatile parameter NAL_PAR_POLICY.

The values not defined in this array are invalid.

8.3 Detection of the End of Transaction Events

The NFC Device should be aware of the end of transaction events when the Secure Element is emulating a card. Even if the card transaction is not successful, the NFC Controller should send an event to the NFC Device at the end of the transaction.

8.3.1 Event NAL_EVT_SE_CARD_EOT

This event is sent to the NFC Device if all the following conditions are true:

- The Secure Element is in RF interface i.e. the SE bits of the parameter NAL_PAR_POLICY are set to “RF Interface”.
- The card emulation was selected by a remote reader.
- The card emulation is de-selected by the remote reader or the remote RF filed is shutdown for any reason.

Depending on the NFC Controller technology, the event NAL_EVT_SE_CARD_EOT may be sent even if there is no successful card transaction or if one of the above conditions is not true.

The format of payload is the protocol used for the transaction followed by the HCR error and flag then by the AID list. HCR error and flag are two bytes notified by the NFC controller to give specific information about this event. An AID is included in the list if the protocol is ISO 14443-4 A or B and a command ISO 7816-4 SELECT AID is sent by the remote reader. Each AID is stored with a length-value pair. The length is stored on one byte and is followed by the bytes of the AID. The length value is the number of bytes in the AID. The syntax of this event is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_SECURE_ELEMENT
1	1	NAL_EVT_SE_CARD_EOT
2	1	The SE identifier
3	2	A card protocol bit field with one and only one bit set for the card protocol used by the SE.
5	1	The HCR error
6	1	The HCR Flag (= SE_1)
7	1	The length N1 of the first AID in bytes
8	N1	The first AID
8+N1	1	The length N2 of the first AID in bytes
9+N1	N2	The second AID
		…

9 Routing table

The routing table is an optional component localized in the NFC Controller.

The routing table determines the routing target of commands coming from an external reader when the system is in card emulation mode and/or P2P mode.

The identification of the applications to be routed (routing objects) can be done on several ways as described below and depends on the specification of the object (e.g. proprietary technology).

- Technology-based Routing
- Protocol-based Routing, including proprietary technologies
- AID-based Routing

The routing table only implements the AID-based routing, technology based and protocol-based routing being performed using the NAL_PAR_POLICY and NAL_PAR_PERSISTENT_POLICY parameters.

9.1 Availability

The availability of the routing table feature is specified by the bit #13 of the field “Firmware capabilities” of the NAL_PAR_FIRMWARE_INFO parameter (see § 3.3.2).

Depending of the NFC Controller, the routing table feature may be only available when used in conjunction with a specific policy (for example, routing table may be only available when embedded SE is put in frontend mode). The specification of these constraints is out of scope of this specification.

9.2 Configuration of the routing table

The NFC device sets the parameter NAL_PAR_ROUTING_TABLE_CONFIG to specify the global configuration of the routing table. This parameter is persistent and remains unchanged across several reboots of the NFC Controller.

Service	NAL_SERVICE_ADMIN
Code	NAL_PAR_ROUTING_TABLE_CONFIG
Access	Read/Write
Storage	Persistent
Size	0x02 bytes

The bit #0 controls activation of the whole routing table functionality. When set to 0, the routing table is disabled, the routing being done statically according to current NAL_PAR_POLICY.

Other bits are RFU and should be set to zero.

A firmware update does not impact the content of the parameter.

9.3 Configuration of the routing table entries

The NFC Device sets the parameter NAL_PAR_ROUTING_TABLE_ENTRIES to specify the AID entries to be interpreted by the routing table component.

Service	NAL_SERVICE_ADMIN
Code	NAL_PAR_ROUTING_TABLE
Access	Read/Write
Storage	Persistent
Size	0x00 – 0xFF bytes

The parameter consists in a set of n routing entries, each entry corresponding to an AID. Each AID must be present only once.

Offset (in bytes)	Length (in bytes)	Description
0	N	Routing Item #1 (see below)
N	P	Routing Item #2
...	…	…

Each routing item has the following format:

Offset (in bytes)	Length (in bytes)	Description
0	1	Flags
1	1	AID Length: 1, or in the range [5-16].
5	1 or [5-16]	AID

Specific AIDs have been defined for specific routing purposes:

The flags item has the following format:

Bit Position	7-6	5	4	3	2-0
Meaning	RFU	Phone ON	Phone OFF	Battery OFF	Target

Target indicates the element to be selected when the AID matches. The following values are supported

- 0 Handset
- 1 UICC
- 2 SE #0
- 3 SE #1
- 4 SE #2
- 5 SE #3

When set, the bit Battery OFF indicates this rules is active in battery OFF mode.
When set, the bit Phone OFF indicates this rules is active in Phone OFF mode.
When set, the bit Phone ON indicates this rules is active in Phone ON mode.

The AID is either a valid AID (its size must be between 5 and 16 bytes) or a specific non-standard AID (1 byte long) used for specific routing purposes.

Specific AID value 0x00 is used to specify the target to be selected when a SELECT with an AID not present in the table or a SELECT without any AID (implicit selection) has been received.

Specific AID value 0x01 is used to specify the target to be selected when an APDU (not a SELECT) is received prior any reception of any SELECT APDU.

The routing table component intercepts the SELECT command to perform the routing, with the following limitations:

- The SELECT command is only authorized in SELECT BY NAME configuration.
- SELECT NEXT does not work with routing table.
- Use of default selected application (no AID provided) is not supported.
- Selection on different logical channels is not supported.
- SELECT with a partial AID will find the first AID that matches in the routing table.

The comparison is always:
- AID in SELECT – full match
- AID in RT – full or partial match

Example:

SELECT Command AID:	11223344556677
RT entry 1 AID:	112233445566 – partial match in SELECT (not considered)
RT entry 2 AID:	1122334455667788 – partial match in RT → OK
RT entry 3 AID:	1122334455667799 – partial match in RT (not considered)
RT entry 4 AID:	11223344556677 – full match in RT (not considered)

Result: RT entry 2 will match.

A firmware update does not impact the content of the parameter.

10 Power Management and Firmware Update

This chapter describes the power management of the NFC Controller, the power modes and the Firmware update.

10.1 Power Modes

The NFC Controller implements 5 power modes. The “Battery On” mode is the only mode where the NCF Device can communicate with the NFC Controller.

The precise power consumption of each mode is described in the data sheet of the NFC Controller.

Shutdown Mode
When the NFC Controller does not receive any electrical power and when no RF field is received from an external reader, the NFC Controller is in Shutdown Mode. In this mode, the NFC Controller cannot perform any operation and the UICC is not communicating through SWP.

Battery On Mode
The NFC Controller is powered by the NFC Device. The UICC is powered through SWP. The NFC Device is powered and communicates with the NFC Controller. The reader protocols, the card protocols and the P2P protocols can be used.

Battery Low Mode
The NFC Controller is powered by the NFC Device but the power is low and the NFC Device itself is not active. The UICC is powered through SWP. Only the card protocols can be used.

Battery Off Mode
The NFC Controller is not powered by the NFC Device. The power comes from a RF field received from an external NFC reader. The UICC is powered through SWP. Only the card protocols can be used. The NFC Controller can return back to the Shutdown mode at any time if the external RF field is cut.

Standby Mode
This mode is similar to the shutdown mode. The NFC Device decides to switch the NFC Controller from Battery On mode to Standby mode to reduce the power consumption of the NFC Device. The event NAL_EVT_STANDBY_MODE is used to switch to Standby mode. The RF communication, the NFC Device communication and the SWP communication are stopped but the UICC receives power from SWP. The NFC Controller exits from the Standby mode when the NFC Device sends a wakeup signal or when a RF field is detected. When the NFC Controller exits from the Standby mode, the volatile parameters and variables are reset to their default values.

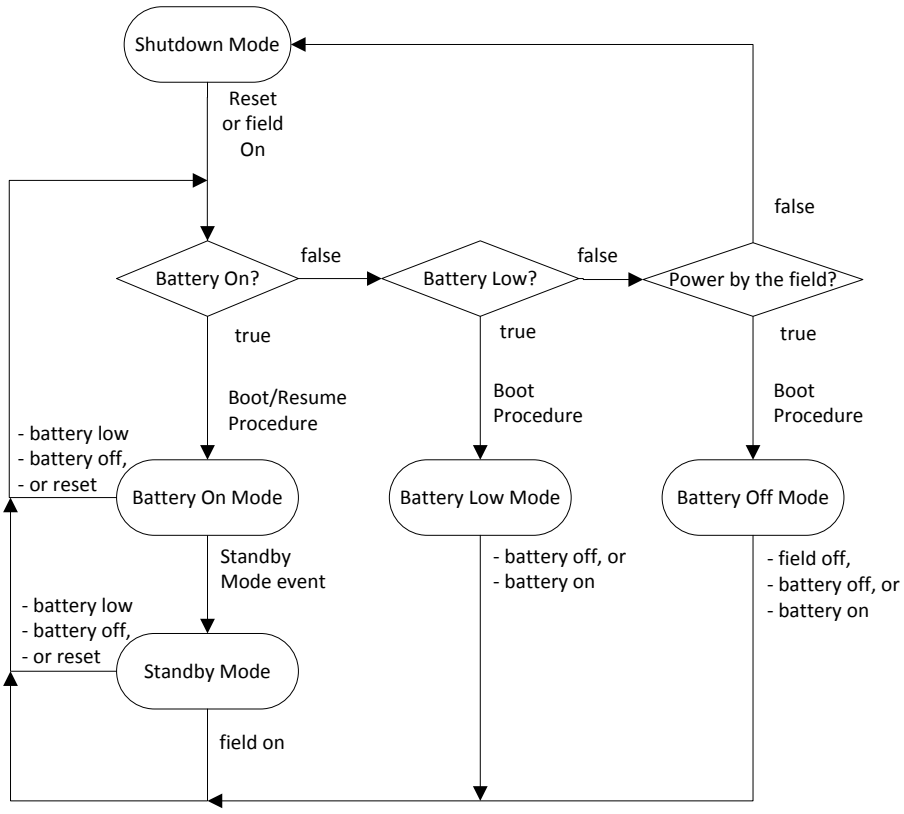

Figure 8 – Transitions between Power Modes

10.1.1 Event NAL_EVT_STANDBY_MODE

The NFC Device sends the event NAL_EVT_STANDBY_MODE to the NFC Controller to allow NFCC to enter in standby mode even if a reader mode is active.

The event NAL_EVT_STANDBY_MODE has no effect if the bit “Standby supported” is not set in the parameter NAL_PAR_FIRMWARE_INFO.

The NCF Controller will resume its activity for two reasons:

- The NFC Device decides to resume the NFC Controller by sending a wakeup signal.
- A RF field is detected by the NFC Controller and this event creates an interrupt causing the wake up of the NFC Device.

The encoding of the event is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_ADMIN
1	1	NAL_EVT_STANDBY_MODE.
2	1	Standby flag: • 0 if the NFC Controller should only enter standby mode if no reader mode is active. • 1 if the NFC Controller can enter standby mode even if the reader mode is active.

10.2 Boot and Resume Sequences

This chapter describes the boot sequence and the resume sequence of the system.

10.2.1 Boot/Resume Procedure for the NFC Controller

When the NFC Controller is booted or after a reset/resume, the SE switch is off and the UICC is not powered. The boot sequence of NFC Controller is the following:

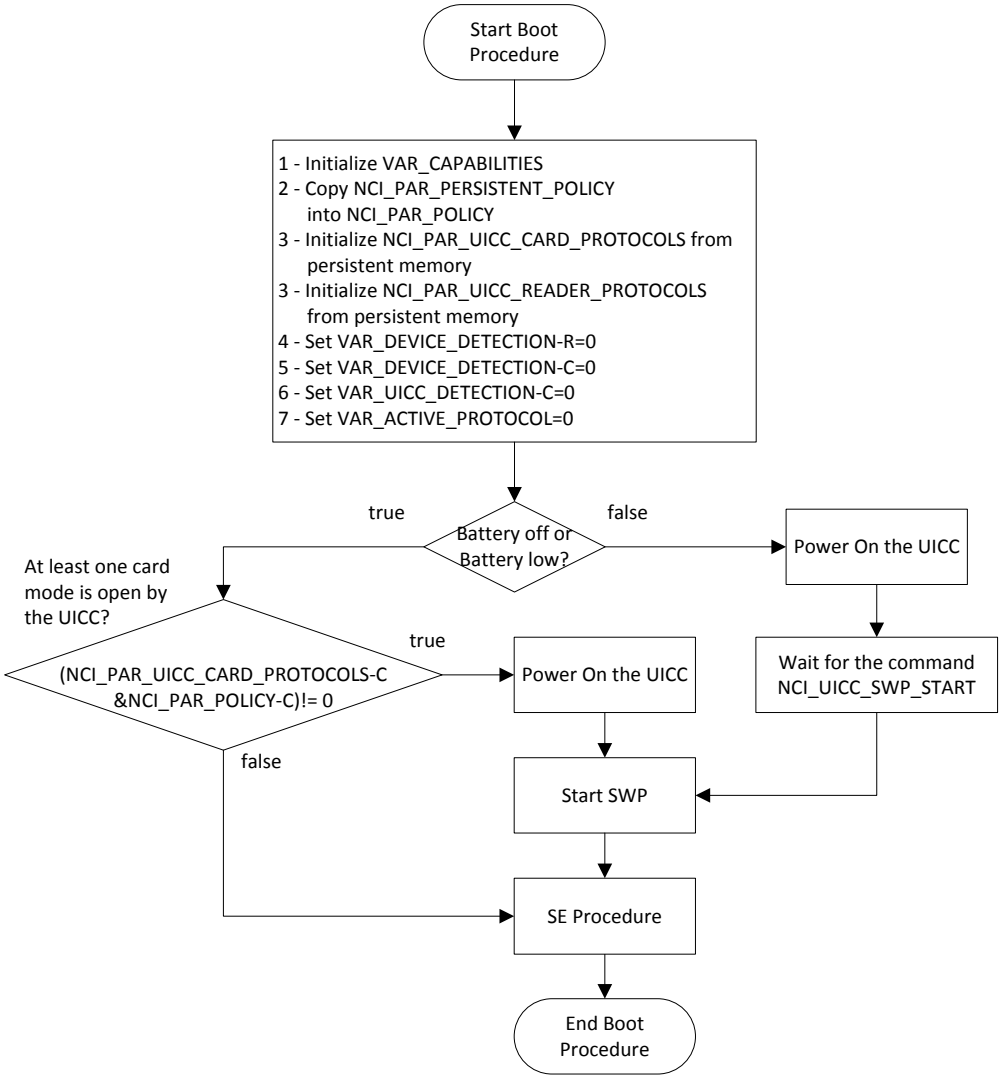

Figure 9 – Boot Procedure for the NFC Controller

10.2.2 Boot Sequence for the NFC Device

The boot sequence of NFC Device is the following:

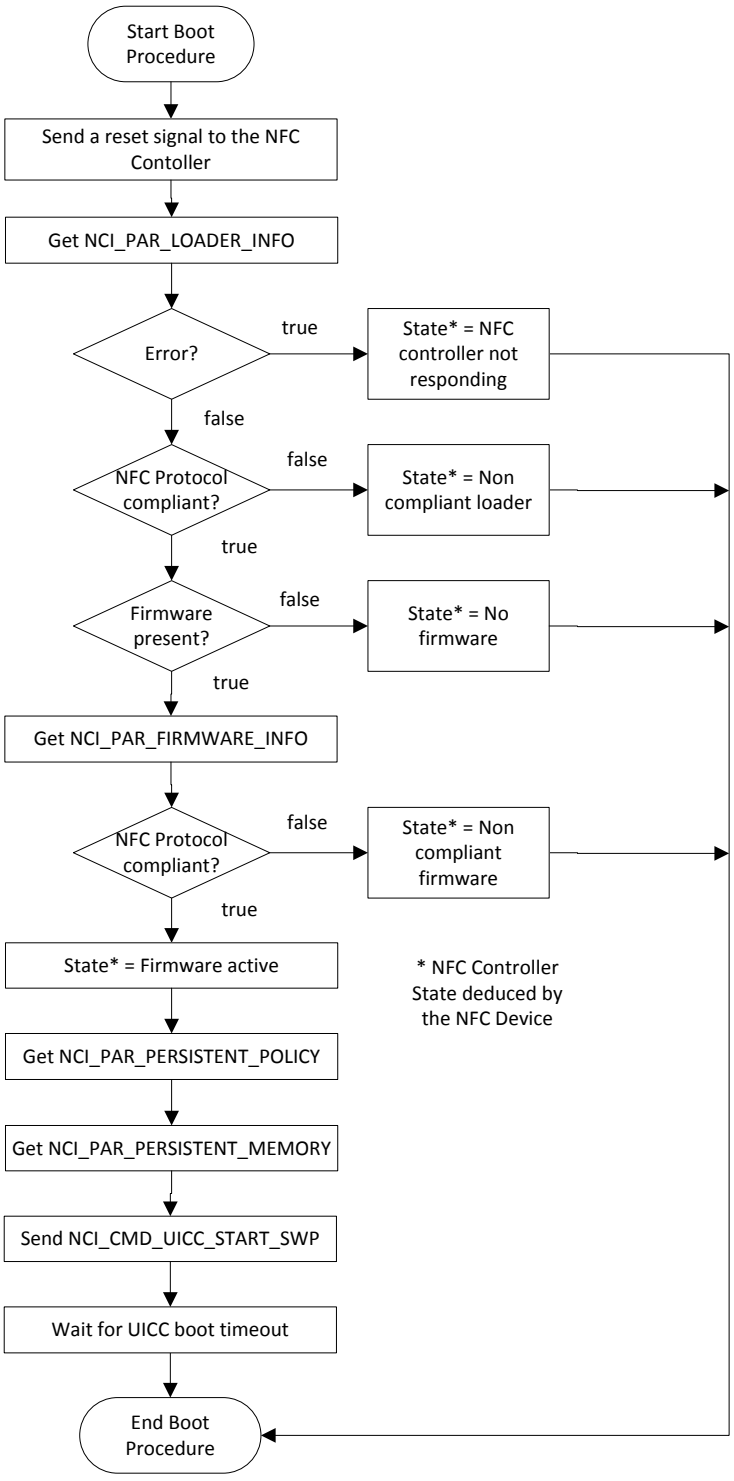

Figure 10 – Boot Sequence for the NFC Device

10.2.3 Resume Sequence for the NFC Device

After exiting from Standby mode, the resume sequence of NFC Device is the following:

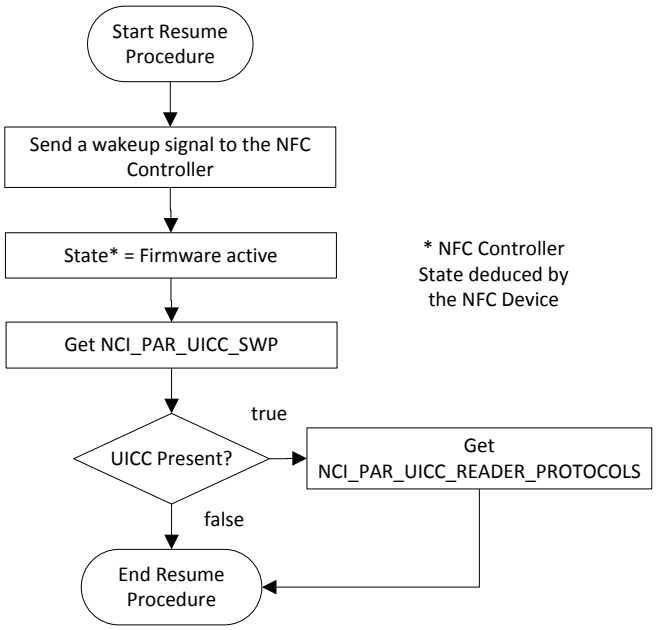

Figure 11 – Resume Sequence for the NFC Device

10.3 Error Detection

If an error is detected during the communication with the NFC Controller, the event NAL_EVT_NFCC_ERROR is returned.

10.3.1 Event NAL_EVT_NFCC_ERROR

The syntax of the event is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_ADMIN
1	1	NAL_EVT_NFCC_ERROR
2	4	The cause of the error, the value is depending on the NFC Controller.

10.4 Firmware Update

This section describes the command updating the firmware of the NFC Controller.

10.4.1 Command NAL_CMD_UPDATE_FIRMWARE

The command NAL_CMD_UPDATE_FIRMWARE allows updating the configuration of the NFC Controller (including operations such as firmware update, tuning of parameters, etc).

Depending on the NFC controller, the configuration is performed using one or several NAL_CMD_UPDATE_FIRMWARE commands.

A NFC Device shall not send to the NFC Controller any message other than NAL_CMD_UPDATE_FIRMWARE during the processing of a previously sent NAL_CMD_UPDATE_FIRMWARE command. Otherwise the behavior of the NFC Controller is unspecified.

An operation shall not be initiated when a detection sequence is ongoing or when a protocol is active. In these cases, the NFC Controller will reject the first command of the update operation with the result code NAL_RES_BAD_STATE.

If the format or the signature of data included in the command payload of the command is not valid, the NFC Controller will reject the corresponding command with the result code NAL_RES_BAD_DATA. The data integrity checking or the signature checking may only occur when the last update command is received.

If the version of the configuration data is not compliant with the NFC Controller, with the NFC Controller current configuration or with the NFC HAL Binding implementation, the command is rejected with the result code NAL_RES_BAD_VERSION.

The update operation is successful if and only if all the update commands returned the result code NAL_RES_OK.

The update operation is failed if any of the following conditions is true:

- Any of the update commands returns a result code other than NAL_RES_OK,
- A message other than NAL_CMD_UPDATE_FIRMWARE is sent to the NFC Controller between two commands of the update operation.
- The NFC Controller is shutdown or reset before the completion of the update operation

When the update operation is failed, the NFC Device should stop sending update operation for this update operation. Otherwise the NFC Controller will reject the commands with the code NAL_RES_BAD_STATE.

After a failed update operation, the former NFC configuration may be lost or corrupted. In this case, the only messages accepted by the NFC Controller are:

- NAL_CMD_GET_PARAMETER for the parameter NAL_PAR_HARDWARE_INFO
- NAL_CMD_UPDATE_FIRMWARE to reconfigure the NFC controller.

After a successful update operation, the NFC Device shall perform a boot sequence.

The encoding of the command is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_ADMIN
1	1	NAL_CMD_UPDATE_FIRMWARE
2	4	The start address of the buffer that contains the configuration information.
6	4	The length of the configuration buffer.

The encoding of the answer is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_ADMIN
1	1	• NAL_RES_OK Success • NAL_RES_BAD_DATA Error detected during the checking of the data integrity or during the checking of the data signature. • NAL_RES_BAD_STATE Command rejected

10.4.2 NFCC Configuration Format

The firmware format is the following:

Offset	Length	Field	Description
0x00	0x04	MAGIC_WORD	Constant value: 0x23D61F9B
0x04	0x01	VERSION	Constant value for version “1.0” : 0x10
0x05	0x04	FILE_SIZE	The total size of the file in bytes
0x09	0x10	LICENSE	The license identifier
0x19	0x01	RESERVED	Reserved
0x1A	0x01	TITLE_LENGTH	The length in bytes of the title. May be zero.
0x1B	TITLE_LENGTH	TITLE	The title of the file. ASCII string [0x20, 0x7E].
0x1B+ TITLE_LENGTH	n	DATA	The configuration data

The 4-bytes long integer values are encoded with the little endian convention.

The header of the NFCC configuration buffer may be used above the NFC HAL interface or in the NFC HAL Binding to check the compliance of the configuration buffer contents with the NFC Controller. This checking avoids starting the download of a configuration incompatible with the NFC Controller.

The whole configuration buffer (including the header) is provided to the NFC HAL binding. This header may or may not be downloaded in the NFC Controller. This is the choice of the NFC HAL Binding. In any case, the NFC Controller shall not rely solely on the header information to check the compliance of the configuration. It shall also use properly authenticated information included in the data section of the configuration buffer.

11 Protocol Detection Sequence

The NFC Controller follows a detection sequence to detect the different protocols. This sequence depends on the variables and the different commands received during the detection sequence.

11.1 Detection Sequence

The following figure illustrates the detection sequence.

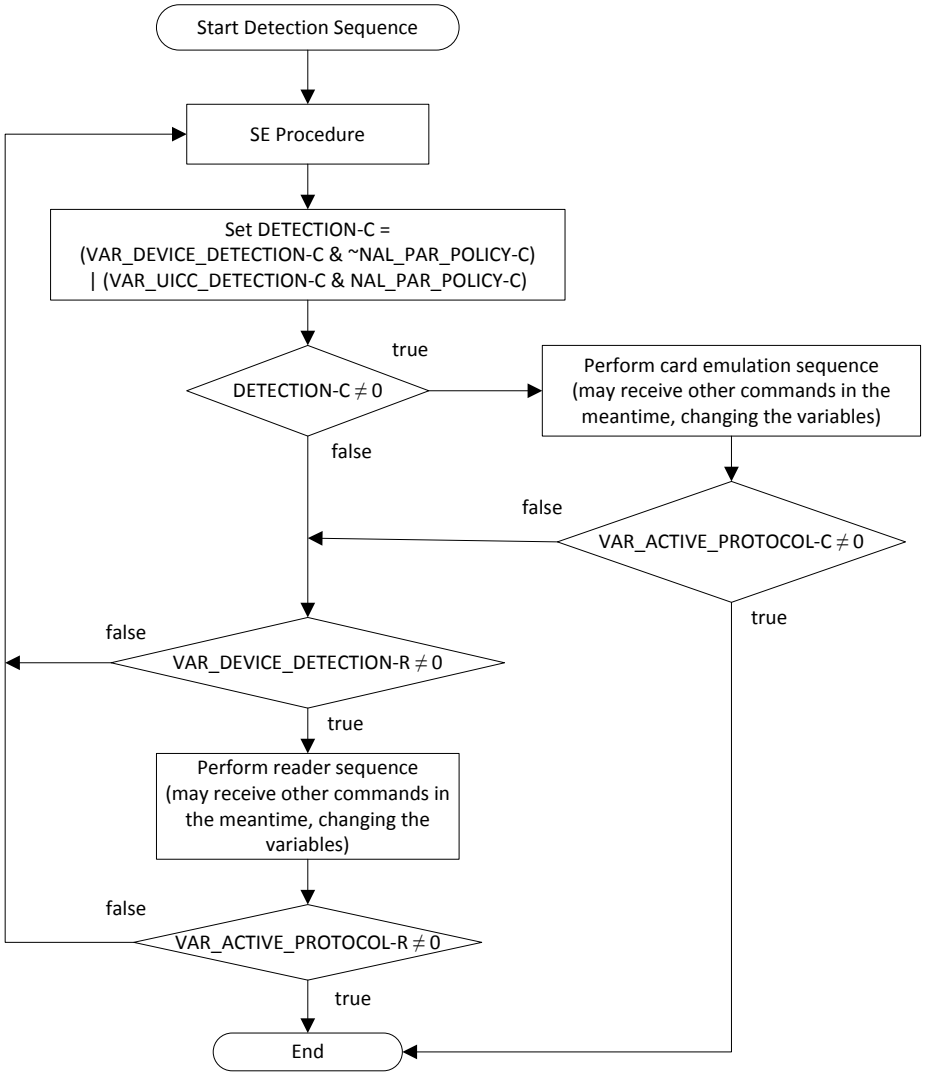

Figure 12 – The detection sequence

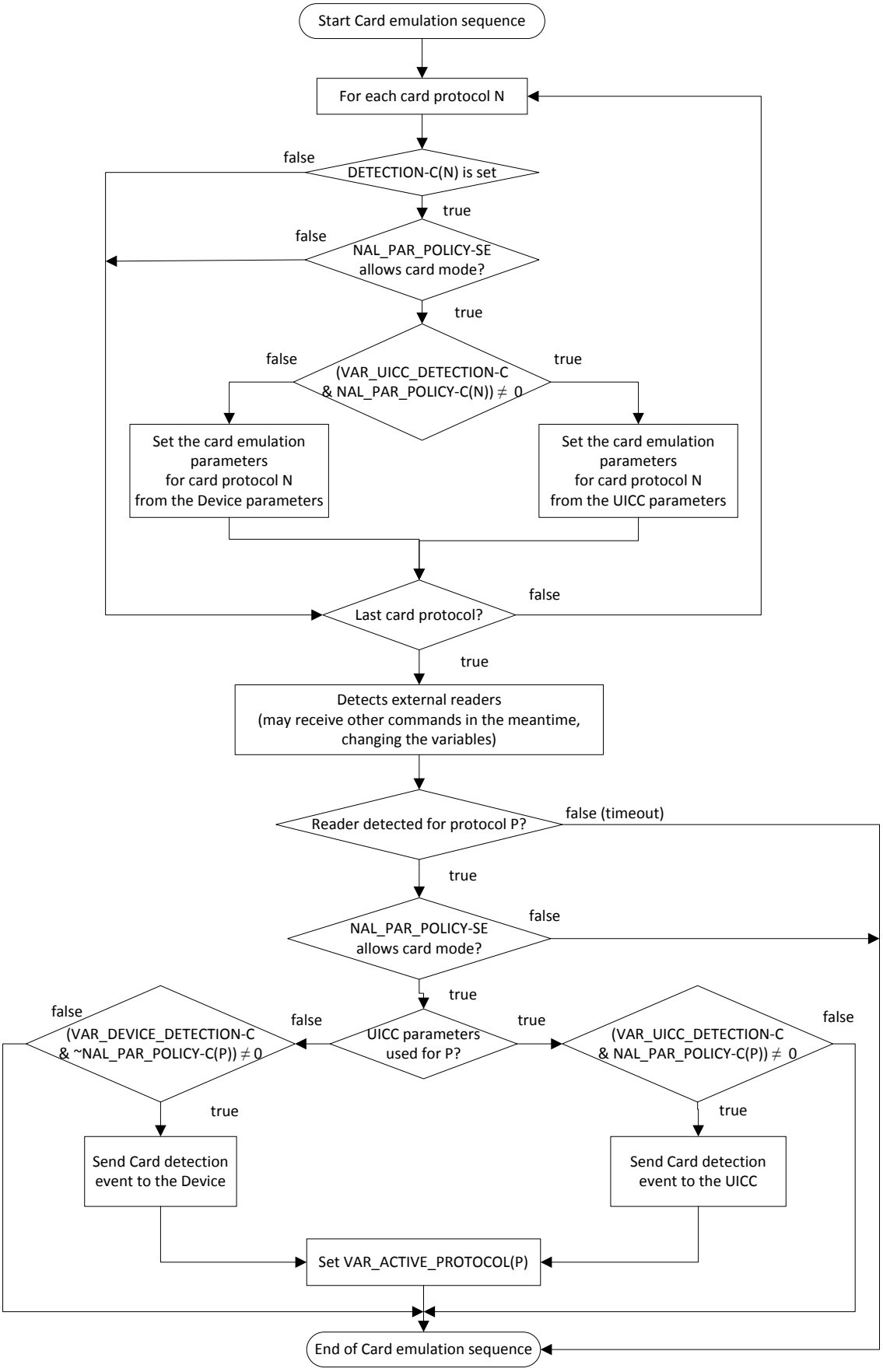

Figure 13 – The card emulation detection sequence

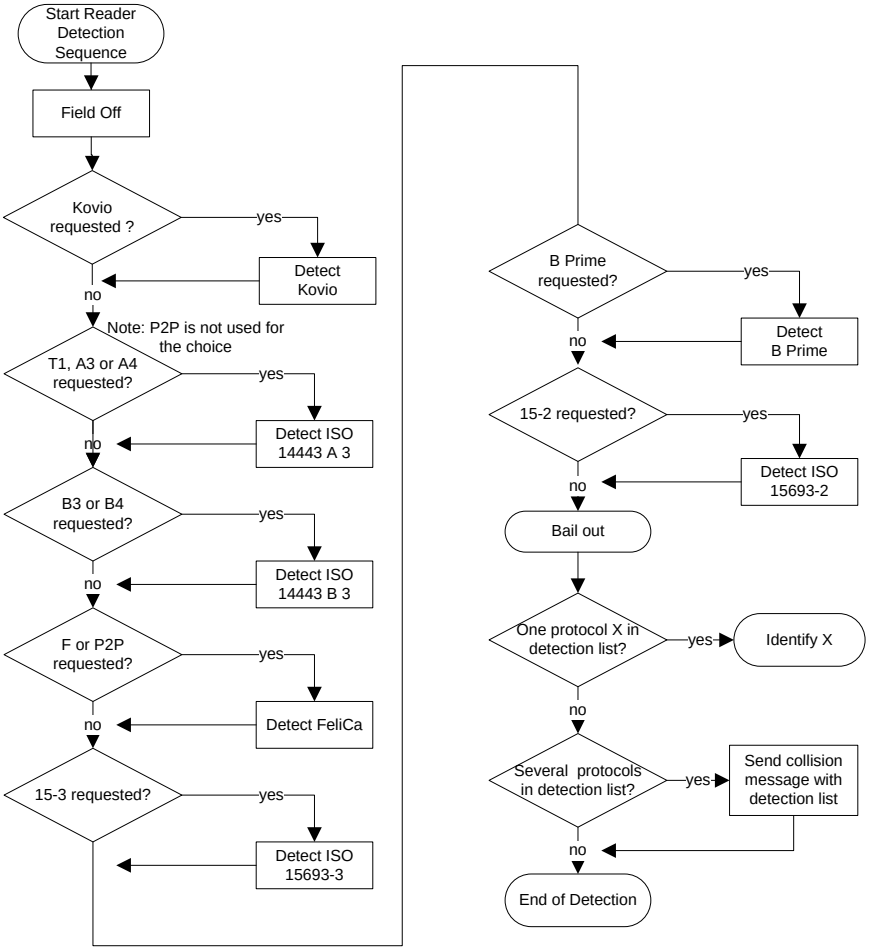

Figure 14 – The reader detection sequence (main loop)

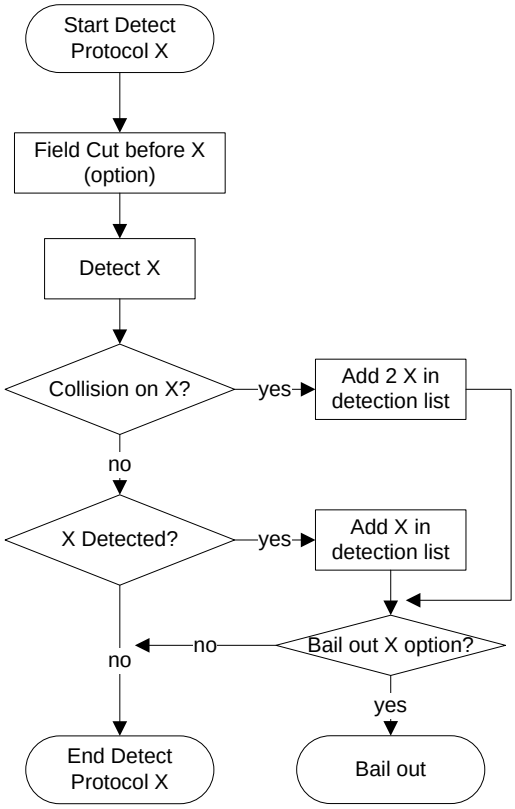

Figure 15 – The reader detection sequence (detection)

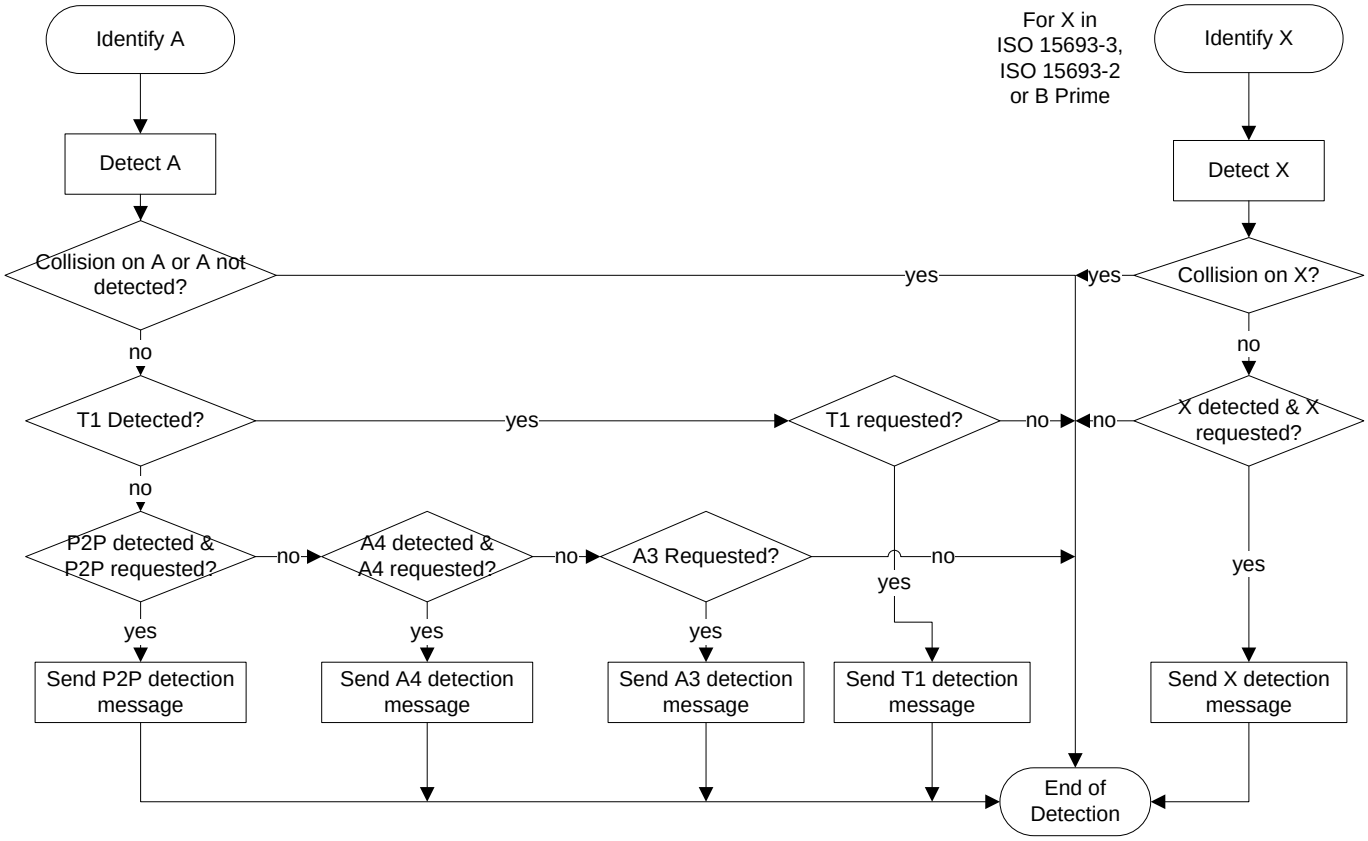

Figure 16 – The reader detection sequence (identification A, B Prime and 15)

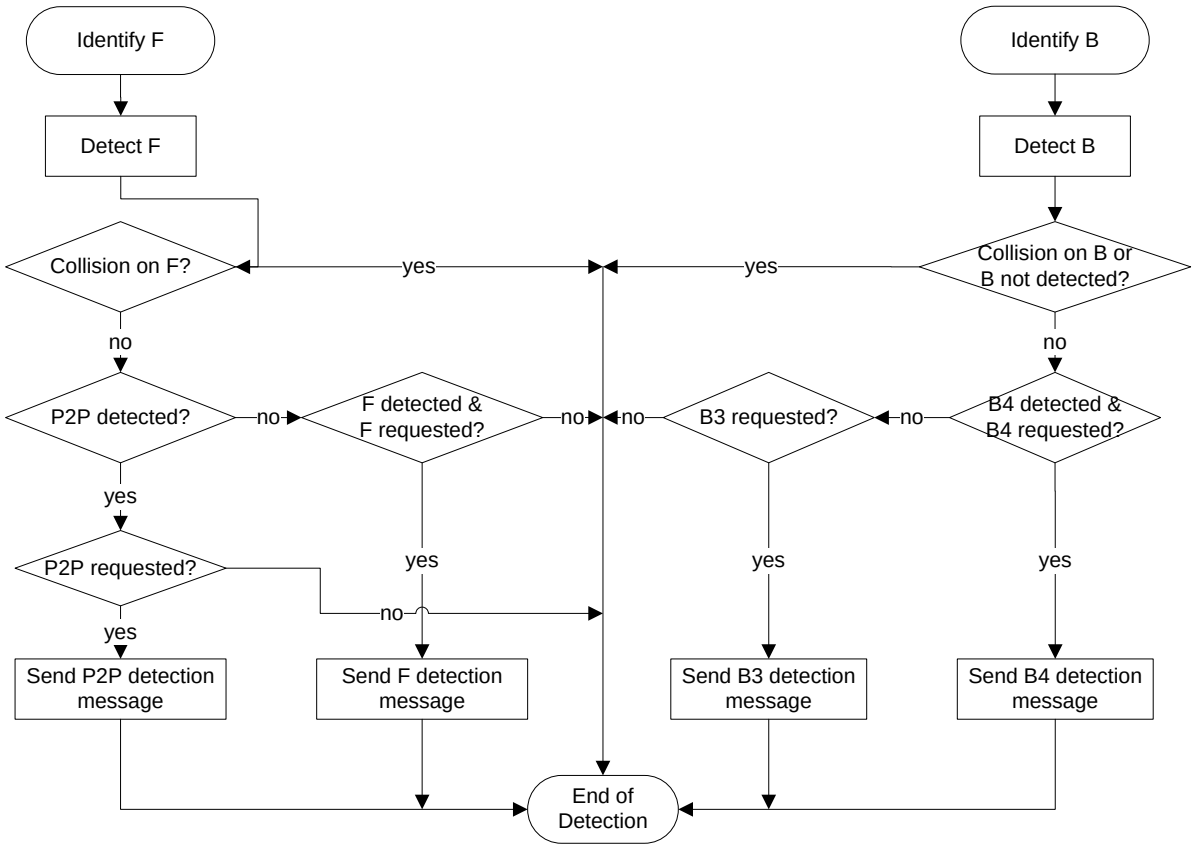

Figure 17 – The reader detection sequence (identification B and F)

11.2 Detection Operations

The NFC device uses only one command to control the detection sequence: NAL_CMD_DETECTION.

11.2.1 Command NAL_CMD_DETECTION

The command NAL_CMD_DETECTION configures the detection sequence for the card protocols and the reader protocols requested by the NFC Device. The encoding of the command is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_ADMIN
1	1	NAL_CMD_DETECTION
2	2	Card protocol descriptor
4	2	Reader protocol descriptor

The parameters of the command are two protocol bit field describing the card protocols and the reader protocols with the following meaning:

- If a bit is set, the corresponding protocol should be included in the detection sequence.
- If a bit is not set, the corresponding protocol should not be included in the detection sequence.

The parameter of the command described above specifies which protocol should be included in the detection sequence. The value is used to update the variables VAR_DEVICE_DETECTION-R and VAR_DEVICE_DETECTION-C. If all the bits of the parameter are set to zero, no protocol is requested.

The encoding of the answer is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_ADMIN
1	1	NAL_RES_OK

The behavior of the command depends on the current state of the NFC Controller:

Initial state	Actions
The NFC Controller is not performing a detection sequence, no card emulation is active and no reader is active.	• If no protocol is requested in the command: does nothing. • If at least one protocol is requested in the command, start the detection for the requested protocol(s).
The NFC Controller is performing a detection sequence.	• Remove from the detection sequence the protocol(s) no longer requested. • Add to the detection sequence the new protocol(s) requested. • Continue the detection sequence if at least one protocol is requested for the sequence. • Stop the detection sequence if no protocol is requested.

Initial state	Actions
A card protocol is active for the NFC Device	• If the active card protocol is requested by the command, does nothing. • If the active card protocol is not requested by the command, stop the card protocol and restart the detection sequence.
A reader protocol is active for the NFC Device	Stop the reader protocol, shut down the RF and restart the detection sequence with the requested protocols
A card protocol is active for the UICC	Does nothing

The following figure illustrates the behavior of the NFC Controller when the Device sends the command NAL_CMD_DETECTION.

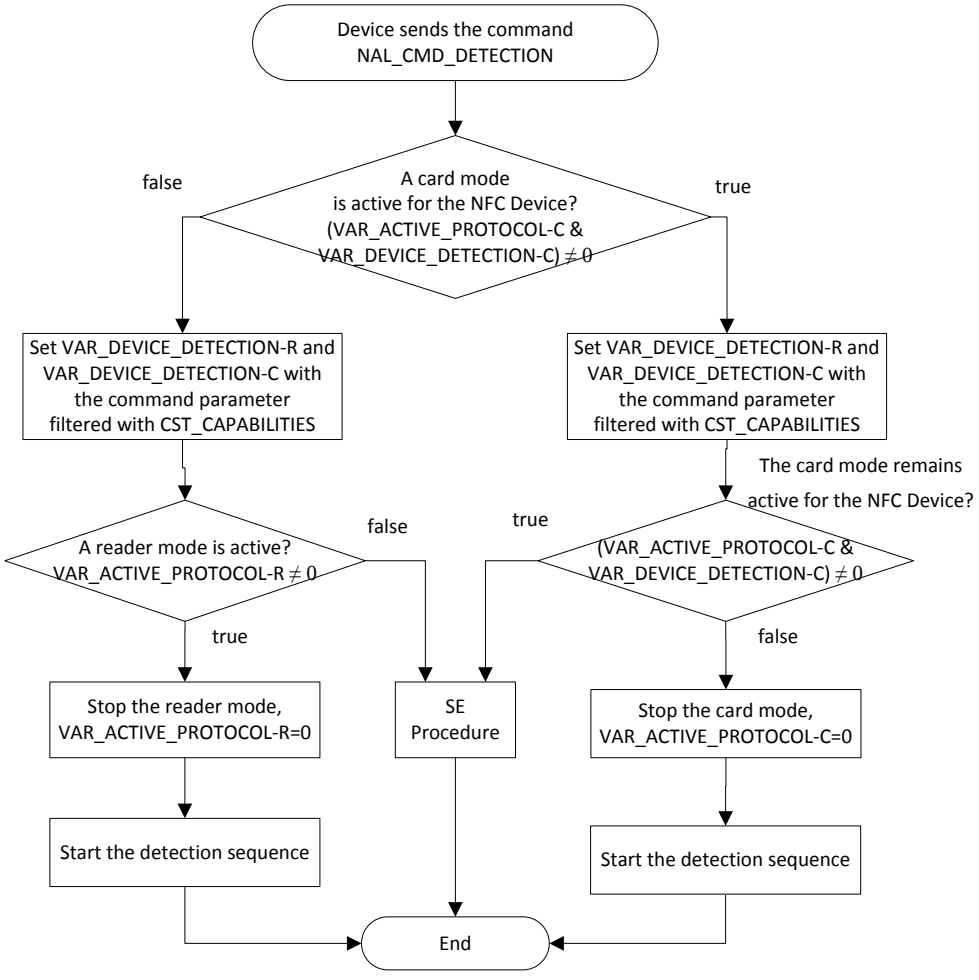

Figure 18 – The device sends the command NAL_CMD_DETECTION

11.2.2 Parameter NAL_PAR_DETECT_PULSE

This 16 bit value is the period of emission of the detection pulse used for the detection of cards. This value is only used if the reader protocol detection is implemented (See “firmware capabilities” in the parameter NAL_PAR_FIRMWARE_INFO).

Service	NAL_SERVICE_ADMIN
Code	NAL_PAR_DETECT_PULSE
Access	Write only
Storage	Volatile
Size	0x02 bytes
Default	0x300

Changing the value will take effect for the next delay between two pulses. A longer interval reduces the power consumption but increase the detection latency.

The payload contains duration of the delay between two pulses, expressed in milliseconds. Depending of the NFC controller capabilities, the range of the allowed values may be restricted to a subset of valid values. In this case, the value will be rounded to the nearest value supported by the NFC controller.

Some typical values: 100 ms, 200 ms, 400 ms and 700 ms.

12 Test Functions

The NFC Controller may execute two types of test. The production tests are reserved for the production line of the NFC Device in laboratory. The production test commands may not return directly the result of the test. The result is obtained with the physical measurement of electrical or RF events using external measurement devices. They are required to validate the physical integration of the NFC Controller into the NFC Device. The self tests are a set of internal tests of the NFC Controller. The self tests are executed in the NFC Device without external measurement devices. The self test commands return the result of the test.

12.1 Production Tests

This section describes the production tests executed by the NFC Controller.

12.1.1 Command NAL_CMD_PRODUCTION_TEST

The encoding of the production test command is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_ADMIN
1	1	NAL_CMD_PRODUCTION_TEST
2	N ≠ 0	The test parameters

The encoding of the answer is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_ADMIN
1	N	The result data if any. The length of the data may be zero.

Specific parameter values stop the current test. Only one test can be executed at a given time.

See the release notes of the NFC HAL Binding for each NFC Controller for the detailed description of the test command.

12.2 Self Tests

This section describes the self tests executed by the NFC Controller.

12.2.1 Command NAL_CMD_SELF_TEST

The self test is a set of tests which are launched with the command NAL_CMD_SELF_TEST.

The encoding of the command is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_ADMIN
1	1	NAL_CMD_SELF_TEST

The encoding of the answer is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_ADMIN
1	1	• NAL_RES_OK Success, the test was executed. See the result value. • NAL_RES_BAD_STATE Command rejected
2	1	A value representing the result of the test. This value is meaningful only when the result code is NAL_RES_OK. • 0x00 The test passed • Another value: the test failed, see the documentation of the NFC Controller. The result value is 0x00 if the result code is not NAL_RES_OK.

Sending the command NAL_CMD_SELF_TEST when one of the following conditions is true may have unspecified effects:

- A communication is ongoing on SWP,
- A detection procedure is pending,
- A protocol is active on the RF, or
- another test is ongoing.

In these cases, the NFC Controller may reject the test command with the result code NAL_RES_BAD_STATE. The NFC Device shall not send the test command when one of the conditions is true.

See the data sheet of the NFC Controller for the detailed description of the test command.

12.3 Raw Mode

The raw mode is a specific mode of the NFC Controller used to exchange raw messages with the NFC Controller. This mode is used to execute some specific tests requiring a proprietary behavior of the NFC Controller. Switching to raw mode is definitive. The NFC Controller should be rebooted to come back to normal mode.

12.3.1 Parameter NAL_PAR_RAW_MODE

This parameter is used to switch to raw mode by setting the value to 1.

Service	NAL_SERVICE_ADMIN
Code	NAL_PAR_RAW_MODE
Access	Write only
Storage	Volatile
Size	0x01 byte
Default	0x00

12.3.2 Event NAL_EVT_RAW_MESSAGE

The NFC Device and the NFC Controller send the event NAL_EVT_RAW_MESSAGE to exchange raw messages. This event is only used in raw mode.

The encoding of the event is the following:

Offset (in bytes)	Length (in bytes)	Description
0	1	NAL_SERVICE_ADMIN
1	1	NAL_EVT_RAW_MESSAGE.
2	N	The raw message.

13 Appendix A: Protocol Constant Values

This appendix describes the constant values of the protocol.

13.1 Global Constants

Message Type	Value
NAL_VERSION	0x08

13.2 Message Type

Message Type	Value
NAL_MESSAGE_TYPE_COMMAND	0x00
NAL_MESSAGE_TYPE_ANSWER	0x40
NAL_MESSAGE_TYPE_EVENT	0x80

13.3 Services

Service Code	Value
NAL_SERVICE_ADMIN	0x00
NAL_SERVICE_READER_14_A_4	0x01
NAL_SERVICE_READER_14_B_4	0x02
NAL_SERVICE_READER_14_A_3	0x03
NAL_SERVICE_READER_14_B_3	0x04
NAL_SERVICE_READER_TYPE_1	0x05
NAL_SERVICE_READER_FELICA	0x06
NAL_SERVICE_READER_15_3	0x07
NAL_SERVICE_CARD_14_A_4	0x08
NAL_SERVICE_CARD_14_B_4	0x09
NAL_SERVICE_P2P_INITIATOR	0x0A
NAL_SERVICE_P2P_TARGET	0x0B
NAL_SERVICE_UICC	0x0C
NAL_SERVICE_SECURE_ELEMENT	0x0D
NAL_SERVICE_READER_15_2	0x0E
NAL_SERVICE_READER_B_PRIME	0x0F
NAL_SERVICE_READER_KOVIO	0x10

13.4 Parameter Codes

Parameter Code	Value
NAL_PAR_PERSISTENT_POLICY	0x00
NAL_PAR_POLICY	0x01
NAL_PAR_HARDWARE_INFO	0x02
NAL_PAR_FIRMWARE_INFO	0x03
NAL_PAR_DETECT_PULSE	0x04
NAL_PAR_PERSISTENT_MEMORY	0x05
NAL_PAR_READER_CONFIG	0x06
NAL_PAR_CARD_CONFIG	0x07
NAL_PAR_P2P_INITIATOR_LINK_PARAMETERS	0x08
NAL_PAR_UICC_SWP	0x09
NAL_PAR_UICC_READER_PROTOCOLS	0x0A
NAL_PAR_UICC_CARD_PROTOCOLS	0x0B

Parameter Code	Value
RFU	0x0C
NAL_PAR_RAW_MODE	0x0D
NAL_PAR_LIST_CARDS	0x0E
NAL_PAR_ROUTING_TABLE_CONFIG	0x0F
NAL_PAR_ROUTING_TABLE_ENTRIES	0x10

13.5 Command Codes

Command Code	Value
NAL_CMD_SET_PARAMETER	NAL_MESSAGE_TYPE_COMMAND | 0x00
NAL_CMD_GET_PARAMETER	NAL_MESSAGE_TYPE_COMMAND | 0x01
NAL_CMD_DETECTION	NAL_MESSAGE_TYPE_COMMAND | 0x02
NAL_CMD_READER_XCHG_DATA	NAL_MESSAGE_TYPE_COMMAND | 0x03
NAL_CMD_COM_TRANSFER	NAL_MESSAGE_TYPE_COMMAND | 0x04
NAL_CMD_PRODUCTION_TEST	NAL_MESSAGE_TYPE_COMMAND | 0x06
NAL_CMD_SELF_TEST	NAL_MESSAGE_TYPE_COMMAND | 0x07
NAL_CMD_UPDATE_FIRMWARE	NAL_MESSAGE_TYPE_COMMAND | 0x08
NAL_CMD_UICC_START_SWP	NAL_MESSAGE_TYPE_COMMAND | 0x09
NAL_CMD_SE_RESET	NAL_MESSAGE_TYPE_COMMAND | 0x0A

13.6 Result Codes

Result Code	Value
NAL_RES_OK	NAL_MESSAGE_TYPE_ANSWER | 0x00
NAL_RES_TIMEOUT	NAL_MESSAGE_TYPE_ANSWER | 0x01
NAL_RES_UNKNOWN_COMMAND	NAL_MESSAGE_TYPE_ANSWER | 0x02
NAL_RES_UNKNOWN_PARAM	NAL_MESSAGE_TYPE_ANSWER | 0x03
NAL_RES_BAD_LENGTH	NAL_MESSAGE_TYPE_ANSWER | 0x04
NAL_RES_BAD_DATA	NAL_MESSAGE_TYPE_ANSWER | 0x05
NAL_RES_BAD_STATE	NAL_MESSAGE_TYPE_ANSWER | 0x06
NAL_RES_PROTOCOL_ERROR	NAL_MESSAGE_TYPE_ANSWER | 0x07
NAL_RES_BAD_VERSION	NAL_MESSAGE_TYPE_ANSWER | 0x08
NAL_RES_FEATURE_NOT_SUPPORTED	NAL_MESSAGE_TYPE_ANSWER | 0x09

13.7 Event Codes

Event Code	Value
NAL_EVT_STANDBY_MODE	NAL_MESSAGE_TYPE_EVENT | 0x00
NAL_EVT_READER_TARGET_COLLISION	NAL_MESSAGE_TYPE_EVENT | 0x01
NAL_EVT_READER_TARGET_DISCOVERED	NAL_MESSAGE_TYPE_EVENT | 0x02
NAL_EVT_UICC_DETECTION_REQUEST	NAL_MESSAGE_TYPE_EVENT | 0x03
NAL_EVT_CARD_SELECTED	NAL_MESSAGE_TYPE_EVENT | 0x04
NAL_EVT_CARD_SEND_DATA	NAL_MESSAGE_TYPE_EVENT | 0x05
NAL_EVT_CARD_END_OF_TRANSACTION	NAL_MESSAGE_TYPE_EVENT | 0x06
NAL_EVT_SE_CARD_EOT	NAL_MESSAGE_TYPE_EVENT | 0x08
NAL_EVT_P2P_TARGET_DISCOVERED	NAL_MESSAGE_TYPE_EVENT | 0x09
NAL_EVT_P2P_INITIATOR_DISCOVERED	NAL_MESSAGE_TYPE_EVENT | 0x0A
NAL_EVT_P2P_SEND_DATA	NAL_MESSAGE_TYPE_EVENT | 0x0B
NAL_EVT_UICC_CONNECTIVITY	NAL_MESSAGE_TYPE_EVENT | 0x0C
NAL_EVT_RF_FIELD	NAL_MESSAGE_TYPE_EVENT | 0x0D
RFU	NAL_MESSAGE_TYPE_EVENT | 0x0E

Event Code	Value
NAL_EVT_RAW_MESSAGE	NAL_MESSAGE_TYPE_EVENT | 0x0F
NAL_EVT_NFCC_ERROR	NAL_MESSAGE_TYPE_EVENT | 0x10

13.8 Reader Protocol Codes

Constant	Value	Description
NAL_PROTOCOL_READER_ISO_14443_4_A	0x0001	Reader ISO 14443 A level 4
NAL_PROTOCOL_READER_ISO_14443_4_B	0x0002	Reader ISO 14443 B level 4
NAL_PROTOCOL_READER_ISO_14443_3_A	0x0004	Reader ISO 14443 A level 3
NAL_PROTOCOL_READER_ISO_14443_3_B	0x0008	Reader ISO 14443 B level 3
NAL_PROTOCOL_READER_ISO_15693_3	0x0010	Reader ISO 15693 level 3
NAL_PROTOCOL_READER_ISO_15693_2	0x0020	Reader ISO 15693 level 2
NAL_PROTOCOL_READER_FELICA	0x0040	Reader FeliCa
NAL_PROTOCOL_READER_P2P_INITIATOR	0x0080	Reader P2P Initiator
NAL_PROTOCOL_READER_TYPE_1_CHIP	0x0100	Reader Type 1
NAL_PROTOCOL_READER_MIFARE_CLASSIC	0x0200	Reader Mifare Classic
NAL_PROTOCOL_READER_BPRIME	0x0400	Reader B Prime
NAL_PROTOCOL_READER_KOVIO	0x0800	Reader Kovio RF barcode ROM ID
NAL_PROTOCOL_READER_MIFARE_PLUS	0x1000	Reader Mifare Plus

13.9 Card Protocol Codes

Constant	Value	Description
NAL_PROTOCOL_CARD_ISO_14443_4_A	0x0001	Card ISO 14443 A level 4
NAL_PROTOCOL_CARD_ISO_14443_4_B	0x0002	Card ISO 14443 B level 4
NAL_PROTOCOL_CARD_ISO_14443_3_A	0x0004	Card ISO 14443 A level 3
NAL_PROTOCOL_CARD_ISO_14443_3_B	0x0008	Card ISO 14443 B level 3
NAL_PROTOCOL_CARD_ISO_15693_3	0x0010	Card ISO 15693 level 3
NAL_PROTOCOL_CARD_ISO_15693_2	0x0020	Card ISO 15693 level 2
NAL_PROTOCOL_CARD_FELICA	0x0040	Card FeliCa
NAL_PROTOCOL_CARD_P2P_TARGET	0x0080	Card P2P Target
NAL_PROTOCOL_CARD_TYPE_1_CHIP	0x0100	Card Type 1
NAL_PROTOCOL_CARD_MIFARE_CLASSIC	0x0200	Card Mifare Classic
NAL_PROTOCOL_CARD_BPRIME	0x0400	Card B Prime
NAL_PROTOCOL_CARD_KOVIO	0x0800	Card Kovio RF barcode ROM ID
NAL_PROTOCOL_CARD_MIFARE_PLUS	0x1000	Card Mifare Plus