NFC Controller Interface (NCI) Specification

Technical Specification

NFC Forum™

NCI 1.0

NFCForum-TS-NCI-1.0

2012-11-06

NFC Forum, Inc.
401 Edgewater Place, Suite 600
Wakefield, MA, USA 01880

Contents

Figures

Tables

1 Introduction

This document specifies a communication protocol called the NFC Controller Interface (NCI) between an NFC Controller (NFCC) and a Device Host (DH).

1.1 Objectives

NCI is defined to meet the following requirements:

- NCI is defined to be independent of a specific transport layer (a physical connection and any associated link protocol). Transport Mappings define the details on how to run NCI on top of different NCI Transport layers.
- NCI is specified to accommodate different levels of functionality in the NFCC with regard to RF communications. Different levels of functionality imply different splits of the NFC Forum protocol stack between the NFCC and the DH. To achieve this, NCI specifies RF Interfaces, each of which can be communicated with a DH and each of which interfaces to a defined functional block implemented on the NFCC. An NCI implementation on an NFCC will be tailored to include the RF Interfaces that match the functionalities of the NFCC.
- NCI provides features to allow DH to NFCEE communication. NCI therefore includes methods to discover and enumerate connected NFCEEs and the NFCEE Interfaces they support, and to establish connections between DH and NFCEE Interfaces. NCI also contains definitions of the data formats that can be exchanged over a connection to NFCEE Interfaces.
- NCI is specified to be extensible to allow future extensions and vendor specific functionality.

1.2 Scope

The following figure outlines a typical architecture of an NFC Forum Device.

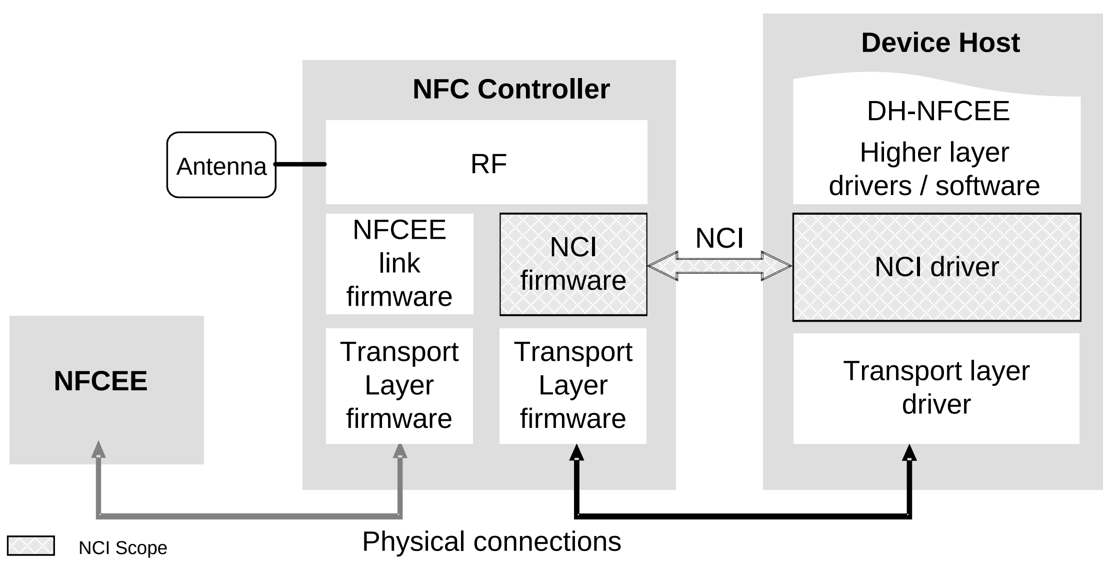

Figure 1: NCI Scope

The NFCC is connected to the Device Host, which is the main application processor in the Device.

The DH higher layer software may contain one or more NFC Execution Environments, or one or more NFC Execution Environments may be connected to the DH (e.g. on an SD card). All NFC Execution Environments on, or connected to, the DH are logically viewed as one entity, called a DH-NFCEE.

In addition one or more NFC Execution Environments may be integrated or connected to the NFCC. These are referred to as NFCEEs.

The scope of NCI is to define the communication between the DH and the NFCC. The communication between the NFCC and NFCEEs is out-of-scope of this specification.

1.3 Audience

This document is intended for use by manufacturers targeting to implement NFC Controllers, NFC Forum Devices, or NFC protocol stacks.

1.4 Applicable Documents or References

[ACTIVITY] NFC Forum Activity Technical Specification,
Version 1.0
NFC Forum

[DIGITAL] NFC Forum Digital Protocol Technical Specification,
Version 1.0
NFC Forum

[ISO/IEC_14443] Identification cards – Contactless integrated circuit cards – Proximity cards
Includes:

[ISO/IEC 14443-1:2008], Identification cards – Contactless integrated circuit cards – Proximity cards – Part 1: Physical characteristics

[ISO/IEC 14443-2:2010], Identification cards – Contactless integrated circuit cards – Proximity cards – Part 2: Radio frequency power and signal balance

[ISO/IEC 14443-3:2001], Identification cards – Contactless integrated circuit cards – Proximity cards – Part 3: Initialization and anticollision

[ISO/IEC_14443-3:2001/Amd.1], Identification cards -- Contactless integrated circuit(s) cards -- Proximity cards -- Part 3: Initialization and Anti-collision, 1 February 2001 with Amendment 1: Bit rates of fc/64, fc/32 and fc/16, 15 June 2005; Amendment 3: Handling of reserved fields and values, 22 March 2006; and Corrigendum 1: Amendment 1 - Corrigendum, 29 August 2006

[ISO/IEC 14443-4:2008], Identification cards – Contactless integrated circuit cards – Proximity cards – Part 4: Transmission protocol

ISO/IEC

[ETSI_102622]	ETSI TS 102 622, Smart Cards; UICC – Contactless Front-end (CLF) Interface; Host Controller Interface (HCI) Release 10 2011, ETSI
[ISO/IEC_7816-3]	ISO/IEC 7816-3, Identification cards - Integrated circuit cards - Part 3: Cards with contacts – Electrical interface and transmission protocols ISO/IEC
[ISO/IEC_7816-4]	ISO/IEC 7816-4, Identification cards - Integrated circuit cards. Organization, security, and commands for interchange, 2005, ISO/IEC
[ISO/IEC_28361]	ISO/IEC 28361 Information technology - Telecommunications and information exchange between systems - Near Field Communication Wired Interface (NFC-WI), 2007, ISO/IEC
[I2C]	I^2C-bus specification and user manual, Rev 03, June 2007, NXP
[LLCP]	NFC Logical Link Control Protocol (LLCP) Technical Specification Version 1.1, NFC Forum
[MANU]	Register of IC manufacturers, ISO/IEC JTC1/SC17, Standing Document 5
[RFC2119]	Key words for use in RFCs to Indicate Requirement Levels, RFC 2119, S. Bradner, March 1997, Internet Engineering Task Force
[T3TOP]	NFC Forum Type 3 Tag Operation Specification, Version 1.0 NFC Forum

1.5 Administration

The NFC Forum NFC Controller Interface (NCI) Specification is an open specification supported by the Near Field Communication Forum, Inc., located at:

401 Edgewater Place, Suite 600
Wakefield, MA, 01880

Tel.: +1 781-876-8955
Fax: +1 781-610-9864

http://www.nfc-forum.org/

The NFC Forum, Inc. maintains this specification. Comments, errors, and other feedback can be submitted at http://www.nfc-forum.org/apps/group_public/add_comment.php?document_id=13436.

1.6 Name and Logo Usage

The Near Field Communication Forum's policy regarding the use of the trademarks *NFC Forum* and the NFC Forum logo is as follows:

- Any company MAY claim compatibility with NFC Forum specifications, whether a member of the NFC Forum or not.
- Permission to use the NFC Forum logos is automatically granted to designated members only as stipulated on the most recent Membership Privileges document, during the period of time for which their membership dues are paid.
- Member's distributors and sales representatives MAY use the NFC Forum logo in promoting member's products sold under the name of the member.
- The logo SHALL be printed in black or in color as illustrated on the Logo Page that is available from the NFC Forum at the address above. The aspect ratio of the logo SHALL be maintained, but the size MAY be varied. Nothing MAY be added to or deleted from the logos.
- Since the NFC Forum name is a trademark of the Near Field Communication Forum, the following statement SHALL be included in all published literature and advertising material in which the name or logo appears:

  NFC Forum and the NFC Forum logo are trademarks of the Near Field Communication Forum.

1.7 Intellectual Property

This document conforms to the Intellectual Property guidelines specified in the NFC Forum's *Intellectual Property Rights Policy*, as outlined in the NFC Forum *Rules of Procedure*. These documents are available on the NFC Forum website.

1.8 Special Word Usage

The key words "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT" and "MAY" in this document are to be interpreted as described in [RFC2119].

1.9 Abbreviations

The abbreviations as used in this document are defined in Table 1.

Table 1: Abbreviations

Abbreviation	Description
AID	Application Identifier
CRC	Cyclic Redundancy Check
DH	Device Host
GID	Group Identifier
HCI	Host Controller Interface
HCP	Host Controller Protocol
ISO	International Organization for Standardization
MT	Message Type
NCI	NFC Controller Interface
NDEF	NFC Data Exchange Format
NFC	Near Field Communication
NFCC	NFC Controller
NFCEE	NFC Execution Environment
OID	Opcode Identifier
PBF	Packet Boundary Flag
RF	Radio Frequency
RFU	Reserved for Future Use
SAR	Segmentation and Reassembly
SDU	Service Data Unit

1.10 Glossary

Application ID (AID)

Defined in [ISO/IEC_7816-4], this is a specific type of Dedicated File (DF) name which is used in a SELECT command to identify applications.

Battery Off State

State where an internal battery or external power source is not available. For example, the battery is removed or empty, so the Device Host (DH) is switched off. The NFC Forum Device can only act in Listen Mode when the NFC Controller (NFCC) and certain NFC Execution Environments (NFCEEs) may be powered by a Remote NFC Device via magnetic coupling.

Command Message

A request sent from the Device Host (DH) to the NFC Controller (NFCC) for action by the NFCC.

Connection Identifier (Conn ID)

A unique 4-bit identifier for a Logical Connection.

Control Message

A generic name when referring to a Command, Response, or Notification Message, but not a Data Message.

NOTE The terms ‘Command’, ‘Response’, and ‘Notification’, as used in this document, are intended to mean the same as ‘Command Message’, ‘Response Message’, and ‘Notification Message’.

Cyclic Redundancy Check (CRC)

A checksum appended within the data segment before transmission, and verified afterward by the recipient to detect Transmission Errors.

Data Message

A message containing a data carried over a Logical Connection.

Destination Type

Identifies an entity (NFCC, NFCEE, or Remote NFC Endpoint) for which a Dynamic Logical Connection is intended.

Device Host (DH)

An Execution Environment responsible for the overall management of the NFC Forum Device and any peripherals. This includes the management (e.g., initialization, configuration, power management, etc.) of the NFC Controller peripheral.

DH-NFCEE

An environment residing on or connected to the DH, where NFC applications are executed. There is logically only one DH-NFCEE, but it may be composed of more than one environment (for example, applications on the DH and applications on a peripheral connected to the DH). The manner in which the DH manages the DH-NFCEE is implementation-specific.

Dynamic Logical Connection

A Logical Connection that is created and closed dynamically as needed.

HCI Network

A Network as described within [ETSI_102622] consisting of a host controller and one or more hosts.

ISO-DEP Protocol

Half-duplex block transmission protocol defined in [ISO/IEC_14443].

Listen Mode

The mode of an NFC Forum Device where it does not generate a carrier. In this mode, the NFC Forum Device listens for the RF field of another device.

Logical Connection

A communication channel between the Device Host (DH) and the NFC Controller (NFCC) that is used for data communication towards either the NFCC itself, an NFCEE, or a Remote NFC Endpoint.

Message

A generic term for a Command, Response, Notification, or Data object communicated between the DH and NFCC.

NCI

The logical interface between a Device Host (DH) and an NFC Controller (NFCC).

NCI Core

Defines the basic NCI functionality required between the Device Host (DH) and NFC Controller (NFCC).

NCI Transport

The physical connection (e.g., SPI, I2C, UART, USB, etc.) and any associated link level protocol between the DH and NFCC. Each supported NCI Transport has a Transport Mapping which defines the characteristics of the NCI Transport. An NCI Transport provides the ability to reliably transfer data without intimate knowledge of the data being transferred. The NCI specification defines multiple Transport Mappings.

NFC Controller (NFCC)

The logical entity responsible for the transmission of data over the NFC Radio Interface. The NFC Controller has a connection to the Device Host (DH) and may have connections to additional NFC Execution Environments (NFCEEs). Those connections are out of scope of this specification, but the impacts to the NCI are in scope.

NFC Execution Environment (NFCEE)

An environment, either built into the NFCC or connected to the NFCC, where NFC applications are executed. The NFCEE may be included in entities with various form factors, some of which can be removable or replaceable.

NFC Radio Interface

A contactless radio interface using NFC technology to communicate with a Remote NFC Endpoint.

NFC-DEP Initiator

The role of an NFC Forum Device reached when an NFC Forum Device in Poll Mode has gone through a number of Activities. In this mode, the NFC Forum Device communicates using the NFC-DEP protocol.

NFC-DEP Target

The role of an NFC Forum Device, reached when the NFC Forum Device in Listen Mode has gone through a number of Activities. In this mode, the NFC Forum Device communicates using the NFC-DEP protocols.

NFCEE Discovery Process

Functionality that allows detection of NFCEEs that are physically connected to the NFCC.

NFCEE Interface

A logical entity on the NFCC that communicates with the DH on one side and an NFCEE on the other side.

NFCEE Protocol

A protocol used in the communication between the NFCC and an NFCEE.

Notification Message

Can only be sent by an NFCC to the DH. It is sent asynchronously and typically contains informational parameters.

Packet

A structure that is used to transmit a Message over the NCI Transport. There are both Control Packets (for transporting Control Messages) and Data Packets (for transporting Data Messages).

Poll Mode

The mode of an NFC Forum Device when it generates a carrier and probes (“polls”) for other devices.

Response Message

Sent by the NFCC for each Command Message received from the DH. The Response Message may contain status information pertaining to the results of the Command Message.

Remote NFC Endpoint

Refers to a remote device, card, or tag, connected wirelessly over the NFC Radio Interface to the local NFC Forum Device.

RF Discovery Process

Functionality that allows detection of a Remote NFC Endpoint and detection by a Remote NFC Endpoint. The DH can configure the RF Discovery Process, which then runs autonomously within the NFCC.

RF Interface

Logical entities that may contain some protocol logic (e.g., an ISO-DEP RF Interface or an NFC-DEP RF Interface) or may be a transparent conduit (e.g., a Frame RF Interface). The DH can only communicate with a Remote NFC Endpoint via an RF Interface, designated as the “Active RF Interface”. The NFCC contains multiple RF Interfaces.

RF Protocol

A protocol used in the communication between the NFCC and a Remote NFC Endpoint.

Static RF Connection

A Logical Connection with a fixed Connection Identifier that always exists after NFCC initialization and is never closed. It is used by the DH to communicate with a Remote NFC Endpoint via an active RF Interface.

Switched On State

In this state, the DH, the NFCC, and all connected NFCEEs are switched on and powered either by internal battery or external power source. The NFC Forum device can act in both Poll and Listen Modes. NCI is only applicable in Switched On state.

Switched Off State

In this state, the DH is switched off, and the NFCC and all connected NFCEEs are powered either by internal battery or external power source. The NFC Forum Device can only act in Listen Mode.

UICC

A Smart Card that conforms to the specifications written and maintained by the TC ETSI Smart Card Platform. It is a platform to resident applications (e.g., USIM, CSIM, ISIM, banking, transport, etc.).

1.11 Coding Conventions

The following coding conventions apply in this document if not stated otherwise.

- Each octet is represented by bits b0 to b7, where b7 is the most significant bit (msb) and b0 the least significant bit (lsb). In all representations, the leftmost bit is the msb.
- All values greater than 1 octet are sent and received in Little Endian format.
- In representations of octet arrays, each octet is numbered, starting at 0. Octet numbered 0 is sent over the NCI Transport first.

This document uses the following notations for numbers:

- Values expressed in hexadecimal form are preceded by ‘0x’.
- Values expressed in binary form are followed by a lower case ‘b’.

In this document, the following rules apply for fields or values defined as Reserved for Future Use (RFU):

- For an NCI message including fields with a subset of octets or bits defined as RFU, the sender SHALL set these octets or bits to the value indicated in this document or to zero if no value is given.
- For an NCI message including fields with a subset of octets or bits defined as RFU, the receiver SHALL disregard these octets or bits and SHALL keep the same interpretation of any other field of the whole message, unless explicitly stated otherwise.

For fields that could contain values that are defined as RFU, the sender of an NCI message SHALL NOT set these fields to the RFU values.

If a parameter is defined as meaningless in certain conditions, the message sender SHALL include the parameter, but MAY set its value to any value. The message receiver SHALL ignore the parameter value.

Values defined as Proprietary MAY be used by implementations for extensions that are out of scope of this specification.

In this document, Control Messages are defined by using tables specifying the message parameters. Control Messages SHALL be coded in the order and length as specified by the corresponding definition table (first parameter being defined in the topmost table row).

The names of Parameters that can be present multiple times in a Control Message are followed by square brackets surrounding the number of times the parameter is to be included. This information is expressed by an index range (e.g., [1..10]). The upper bound of this range can be a variable defined by another parameter. The parameter instances SHALL follow each other sequentially. The corresponding parameter length indicates the length of a single parameter instance (not the length of the complete parameter set).

2 NCI Architecture

This section outlines some of the basic concepts used in NCI. It is an informal introduction to the documentation and normative statements later in this document.

2.1 Components

NCI can be split into the following logical components:

- NCI Core

  The NCI Core defines the basic functionality of the communication between a Device Host (DH) and an NFC Controller (NFCC). This enables Control Message (Command, Response, and Notification) and Data Message exchange between an NFCC and a DH.

- Transport Mappings

  Transport Mappings define how the NCI messaging is mapped to an underlying NCI Transport, which is a physical connection (and optional associated protocol) between the DH and the NFCC. Each Transport Mapping is associated with a specific NCI Transport.

- NCI modules

  NCI modules build on top of the functionality provided by the NCI Core. Each module provides a well-defined functionality to the DH. NCI modules provide the functionality to configure the NFCC and to discover and communicate with Remote NFC Endpoints or with local NFCEEs.

  Some NCI modules are mandatory parts of an NCI implementation, others are optional.

  There can also be dependencies between NCI modules in the sense that a module may only be useful if there are other modules implemented as well. For example, all modules that deal with communication with a Remote NFC Endpoint (the RF Interface modules) depend on the RF Discovery to be present.

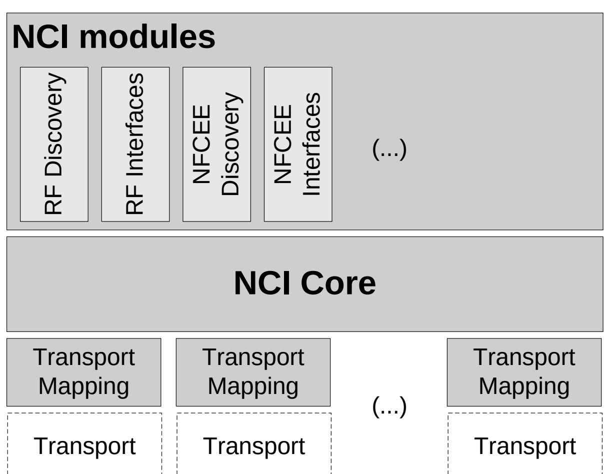

Figure 2: NCI Components

2.2 Concepts

This section outlines the basic concepts used in NCI.

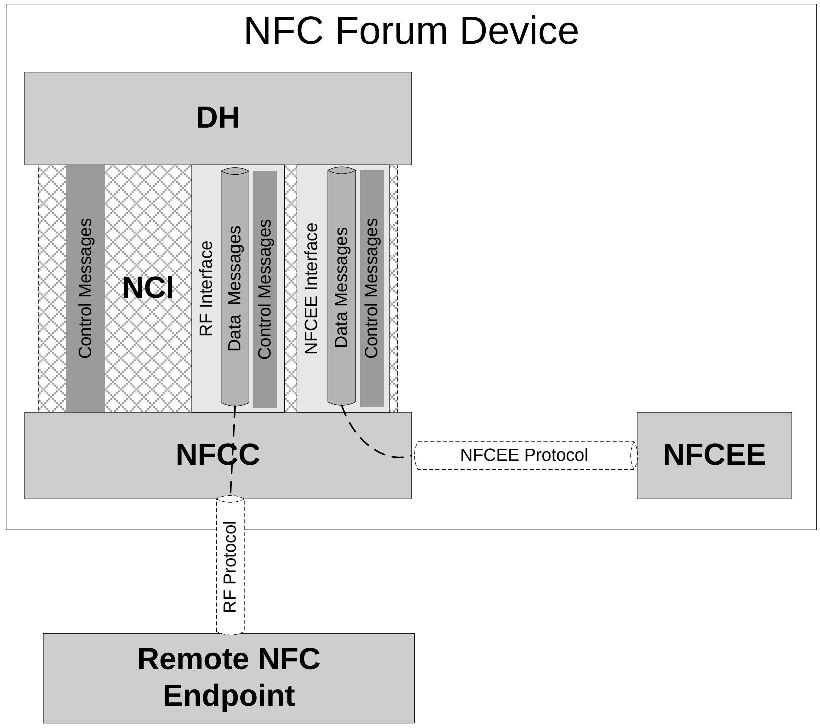

Figure 3: NCI concepts

2.2.1 Control Messages

A DH uses NCI Control Messages to control and configure an NFCC. Control Messages consist of Commands, Responses, and Notifications. Commands are only allowed to be sent in the direction from DH to NFCC, and Responses and Notifications are only allowed to be sent in the other direction. Control Messages are transmitted in NCI Control Packets, and NCI supports segmentation of Control Messages into multiple Packets.

The NCI Core defines a basic set of Control Messages (e.g., for setting and retrieving of NFCC configuration parameters). Additional Control Messages can be defined by NCI Modules.

2.2.2 Data Messages

Data Messages are used to transport data to either a Remote NFC Endpoint (named RF Communication in NCI) or to an NFCEE (named NFCEE Communication). NCI defines Data Packets enabling the segmentation of Data Messages into multiple Packets.

Data Messages can only be exchanged in the context of a Logical Connection. As a result, a Logical Connection must be established before any Data Messages can be sent. One Logical Connection, the Static RF Connection, is always established during initialization of NCI. The Static RF Connection is dedicated to be used for RF Communication. Additional Logical Connections can be created for RF and/or NFCEE Communication.

Logical Connections provide flow control for Data Messages in the direction from DH to NFCC.

2.2.3 Interfaces

An NCI Module may contain one Interface. An Interface defines how a DH can communicate via NCI with a Remote NFC Endpoint or NFCEE. Each Interface is defined to support specific protocols and can only be used for those protocols (the majority of Interfaces support exactly one protocol). NCI defines two types of Interfaces: RF Interfaces and NFCEE Interfaces.

Protocols used to communicate with a Remote NFC Endpoint are called RF Protocols. Protocols used to communicate with an NFCEE are called NFCEE Protocols.

An NFCEE Interface has a one-to-one relationship to an NFCEE Protocol, whereas there might be multiple RF Interfaces for one RF Protocol. The later allows NCI to support different splits of the protocol implementation between the NFCC and DH. An NCI implementation on an NFCC should include those RF Interfaces that match the functionality implemented on the NFCC.

Interfaces must be activated before they can be used and they must be deactivated when they are no longer used.

An Interface can define its own configuration parameters and Control Messages, but most importantly it must define how the payload of a Data Message maps to the payload of the respective RF or NFCEE Protocol and, in the case of RF Communication, whether the Static RF Connection and/or Dynamic Logical Connections are used to exchange those Data Messages between the DH and the NFCC.

2.2.4 RF Communication

RF Communication is started by configuring and running the RF Discovery process. The RF Discovery is an NCI module that discovers and enumerates Remote NFC Endpoints.

For each Remote NFC Endpoint, the RF Discovery Process provides the DH with the information about the Remote NFC Endpoint gathered during the RF Discovery Process. One part of this information is the RF Protocol that is used to communicate with the Remote NFC Endpoint. During RF Discovery configuration, the DH must configure a mapping that associates an RF Interface for each RF Protocol. If only a single Remote NFC Endpoint is detected during one discovery cycle, the RF Interface for this Endpoint is automatically activated. If there are multiple Remote NFC Endpoints detected in Poll Mode, the DH can select the Endpoint it wants to communicate with. This selection also triggers activation of the mapped Interface.

After an RF Interface has been activated, the DH can communicate with the Remote NFC Endpoint using the activated RF Interface. An activated RF Interface can be deactivated by either the DH or the NFCC (e.g., on behalf of the Remote NFC Endpoint). However each RF Interface can define which of those methods are allowed. There are different deactivation options depending on which part of the protocol stack is executed on the DH. For example if a protocol command to tear down the communication is handled on the DH, the DH will deactivate the RF Interface. If such a command is handled on the NFCC, the NFCC will deactivate the Interface.

This specification describes the possible Control Message sequences for RF Communication in the form of a state machine.

2.2.5 NFCEE Communication

The DH can learn about the NFCEEs connected to the NFCC by using the NFCEE Discovery module. During NFCEE Discovery, the NFCC assigns an identifier for each NFCEE. When the DH wants to communicate with an NFCEE, it needs to open a Logical Connection to the NFCEE using the corresponding identifier and specifying the NFCEE Protocol to be used.

Opening a Logical Connection to an NFCEE automatically activates the NFCEE Interface associated to the protocol specified. As there is always a one-to-one relationship between an NFCEE Protocol and Interface, there is no mapping step required (unlike RF Interface activation).

After the Interface has been activated, the DH can communicate with the NFCEE using the activated Interface.

Closing the connection to an NFCEE Interface deactivates that NFCEE Interface.

NCI also includes functionality to allow the DH to enable or disable the communication between an NFCEE and the NFCC.

2.2.6 Identifiers

NCI uses different Identifiers for Remote NFC Endpoints and NFCEEs. These identifiers are dynamically assigned by the NFCC. The DH gets to know them in the context of RF Discovery and NFCEE Discovery respectively. The identifiers for Remote NFC Endpoints are called RF Discovery IDs. They usually have a short lifetime as they are only valid for the time the DH wants to be able to communicate with the Remote NFC Endpoint. In contrast, the identifiers for NFCEEs have a longer lifetime as NFCEEs usually are not frequently added to or removed from a device. The identifiers for NFCEEs are called NFCEE IDs. There is one reserved and static NFCEE ID, value 0, which represents the DH-NFCEE.

Logical Connections take a third type of identifier, Destination Type, as a first parameter to identify the destination for the data. Depending on the Destination Type, there can be a second parameter for identifying the data destination. For example, if the Destination Type is ‘Remote NFC Endpoint’, the second parameter will be an RF Discovery ID.

2.2.7 NFCC as Shared Resource

The NFCC might only be used by the DH but also by the NFCEEs in the device (in such a case the NFCC is a shared resource). NFCEEs differ in the way they are connected to the NFCC, and the protocol used on such a link determines how an NFCEE can use the NFCC. For example, some protocols allow the NFCEE to provide its own configuration for RF parameters to the NFCC (similar to the NCI Configuration Parameters for RF Discovery). In other cases, the NFCEE might not provide such information.

NFCCs can have different implementation in how they deal with multiple configurations from DH and NFCEEs. For example, they might switch between those configurations so that only one is active at a time, or they might attempt to merge the different configurations. During initialization, NCI provides information for the DH as to whether the configuration it provides is the only one or if the NFCC supports configuration by NFCEEs as well.

NCI includes a module, called Listen Mode Routing, with which the DH can define where to route received data when the device has been activated in Listen Mode. The Listen Mode Routing allows the DH to maintain a routing table on the NFCC. Routing can be done based on the technology or protocol of the incoming traffic or based on application identifiers if 7816-4 APDU commands are used on top of ISO-DEP.

In addition, NCI enables the DH to get informed if communication between an NFCEE and a Remote NFC Endpoint occurs.

3 NCI Core Framework

3.1 Overview

The NCI Core includes the following required functionality:

- Packet formats to transmit Commands, Responses, Notification, and Data Messages over NCI.
- Definition of the Commands, Responses, and Notifications used for different operations as described in Section 4 between a Device Host and an NFC Controller. (Some subsequent sections in this specification define additional Commands, Responses, and Notifications that are not part of the NCI Core.)
- A flow control mechanism for Command/Response Message exchange.
- A Logical Connection concept for Data Messages.
- A credit-based flow control mechanism for Data Messages sent from the DH to the NFCC.
- Segmentation and reassembly for Control and Data Messages.
- An addressing scheme for NFC Execution Environments (NFCEE). The NCI Core supports communication between the DH and NFCEEs connected to the NFCC.

NOTE NCI only covers the link between a DH and an NFCC, so communication from a DH to an NFCEE will only be possible if it is also supported by the protocols used between the NFCC and the NFCEE.

- An addressing scheme for Remote NFC Endpoints. The NCI Core uses Logical Connections to support communication between the DH and targets (NFCEEs or Remote NFC Endpoints), which are discovered by the NFCC.
- Reset, initialization, and configuration of the NFCC.
- Exception handling, including Control Messages for indicating errors, and rules for how to use them.

3.2 NCI Control Messages

Control Messages contain Commands, Responses, and Notifications. They control the interaction between the DH and the NFCC. See Figure 4.

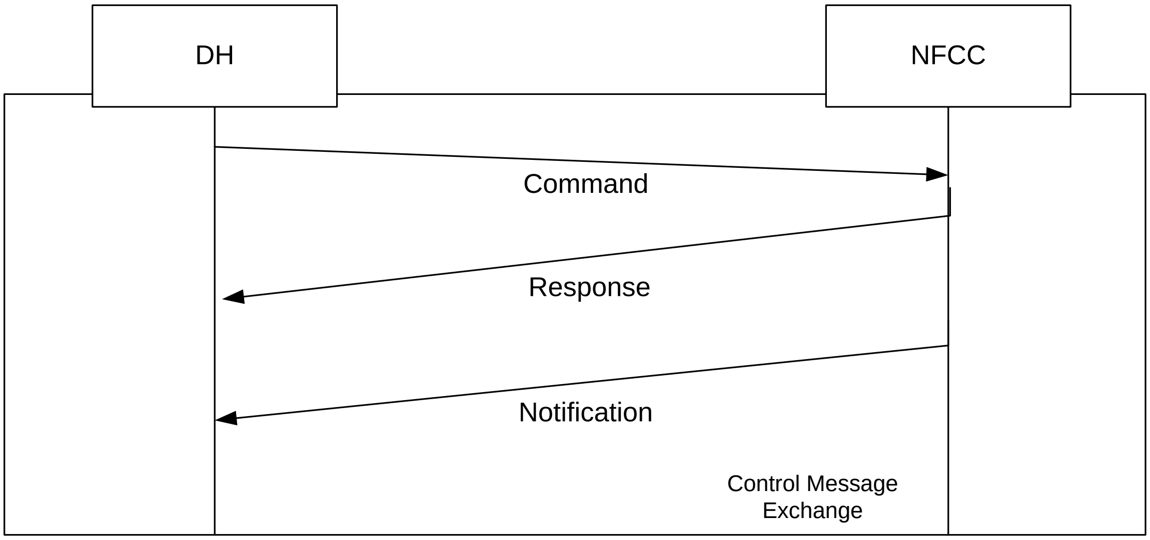

Figure 4: Control Message Exchange

A Command can be sent by the DH to instruct the NFCC to perform a specific action. For each Command received, the NFCC SHALL answer with a Response to acknowledge the receipt of the Command. The Response MAY also indicate changes that the Command caused at the NFCC.

Notifications SHALL only be sent from the NFCC to the DH. A Notification can be sent to deliver additional information related to a Command. A Notification can also be sent independently of any Command or Response, unless specified otherwise.

Control Messages are sent over the NCI as payload of Control Packets. A Control Packet payload contains a complete, or a segment of a, Control Message.

Both the DH and the NFCC SHALL be capable of supporting Control Messages with a payload of 255 octets, which is the maximum size of any Control Message.

The maximum payload length of a Control Packet is also 255 octets. The DH SHALL be capable of receiving Control Packets with 255 octet payloads. However, the NFCC MAY specify a smaller maximum Control Packet payload size defined by the parameter Max Control Packet Payload Size, see Section 4.2.

As a result, Control Messages can be segmented into multiple Control Packets when sent over the NCI, as described in Section 3.5.

3.2.1 Flow Control for Control Messages

The DH and NFCC are allowed to send a complete Control Message over the NCI in as many Packets as needed. There is no Packet-based flow control for Control Messages in NCI.

The following flow control rules apply to Control Messages:

- After sending a Command, the DH SHALL NOT send another Command until it receives a Response for that Command. Although the NCI Transport between the DH and NFCC is reliable, as a precaution the DH MAY maintain a timer, $\mathbf{T}_{\text{Request}}$, to check for a Response. If $\mathbf{T}_{\text{Request}}$ expires without a Response being received, it SHOULD be treated as an Exception and processed as in Section 3.2.2. The value of $\mathbf{T}_{\text{Request}}$ is left up to implementations, but SHOULD NOT be less than one second.

- After sending a Command, the DH SHALL be able to receive a Response.
- After sending a Response, the NFCC SHALL be ready to receive the next Command from the DH.
- The DH SHALL be able to receive a Notification from the NFCC at any time.

3.2.2 Exception Handling for Control Messages

The rules in this section define the exception processing to be performed by a receiver of an erroneous Control Message.

Any Command received by the DH SHALL be ignored. Any Response or Notification received by the NFCC SHALL be ignored.

In the case of a Control Message with syntax errors, meaning that the coding of the Control Packet is not consistent with this specification and where the receiver can still determine the type of the Control Message:

- If the Control Message is a Command, the NFCC SHALL ignore the content of the Command and send a Response with the same GID and OID as in the Command and with a Status value STATUS_SYNTAX_ERROR. The Response SHALL NOT contain any additional fields.
- If the Control Message is a Response, the DH SHALL ignore the content of the Response and is free to send another Command.
- If the Control Message is a Notification, the DH SHALL ignore the Notification.

In case of a Control Message with a semantic error, meaning that a Control Message with valid syntax is received when it is not expected:

- An unexpected Response or Notification SHALL be ignored by the DH.
- An unexpected Command SHALL NOT cause any action by the NFCC. Unless otherwise specified, the NFCC SHALL send a Response with a Status value of STATUS_SEMANTIC_ERROR and no additional fields.

The NFCC SHALL respond to an unknown Command (unknown GID or OID) with a Response having the same GID and OID as the Command, followed by a Status field with the value of STATUS_SYNTAX_ERROR and no additional fields.

The DH SHALL ignore any unknown Response or Notification (unknown GID or OID).

If the NFCC cannot perform the action requested in a valid Command, the NFCC SHALL inform the DH using a Response with one of the Status field values defined in Table 94. Allowed Status values are specified for each Response. Reasons for not being able to perform a Command could be buffer overflow, limited processing power, limited resources, etc.

If a Response for a corresponding Command is not received by the DH within the time set for $\mathbf{T}_{\text{Request}}$, the DH MAY resend the Command or send another Command.

3.3 NCI Data Messages

Data Messages are used to exchange data over Logical Connections between a DH and NFCC target (NFCEE or Remote NFC Endpoint). See Figure 5.

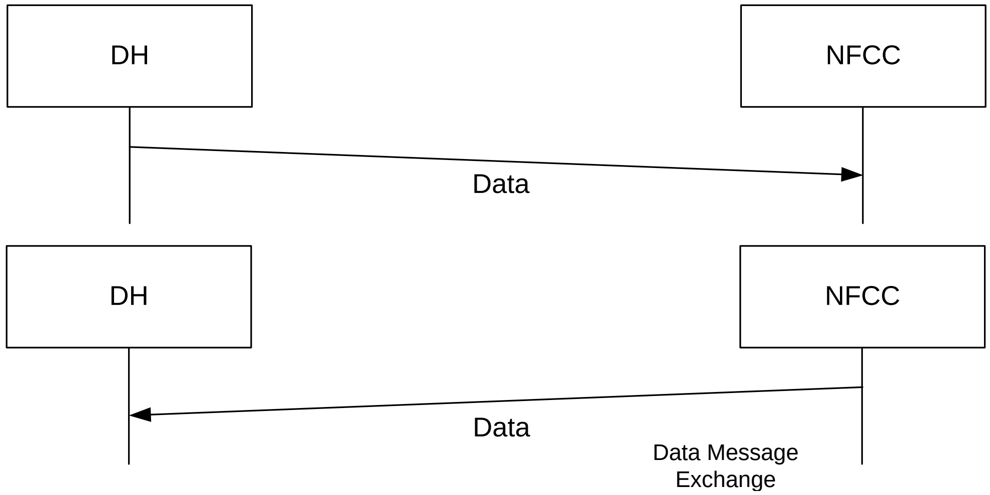

Figure 5: Data Exchange

Data Messages are sent over the NCI Transport as payloads of Data Packets. A Data Packet payload contains a complete, or a segment of a, Data Message.

Data Messages can be sent by either the DH (subject to flow control) or the NFCC at any time once a Logical Connection has been created.

The DH SHALL be capable of supporting any Data Message size sent by the NFCC (it is assumed that DH is able to handle any data received either from a local NFCEE or Remote NFC Endpoint). At any time, the maximum data size the NFCC is able to receive for a Logical Connection is defined by the Max Data Packet Payload Size announced during RF Interface activation in a case of Static RF Connection (see Section 7.3) or during Dynamic Logical Connection (see Section 4.4), times the number of unused credits given by the NFCC for that connection.

The maximum payload of a Data Packet is 255 octets. The DH SHALL be capable of receiving Data Packets with 255 octet payloads. However, the NFCC MAY specify a smaller maximum Data Packet payload size, on a Logical Connection basis.

The DH SHALL NOT send Data Packets with a payload length exceeding the Max Data Packet Payload Size that was announced during creation of the corresponding Logical Connection.

Data Messages can be segmented as described in Section 3.5.

3.3.1 Flow Control for Data Packets

A credit-based Data flow control mechanism is defined for data sent from the DH to the NFCC and MAY be invoked by the NFCC to eliminate buffer overflow conditions. It is assumed that the DH has sufficient buffering to handle all data sent from the NFCC, so credit-based flow control is not supported in that direction.

The credit-based flow control mechanism applies to Data Packets. Each Data Packet (that can contain either a complete Data Message or a segment of a Data Message) requires one credit.

Flow control is configured per Logical Connection during connection establishment (see Section 7.3 for Static RF Connection and Section 4.4 for Dynamic Logical Connection). It may be enabled or disabled differently for each Logical Connection and with different parameters.

It is mandatory for the DH to support credit-based flow control, though it is optional for the NFCC to request the DH to use flow control.

Please refer to Section 4.4.4 regarding the normative rules for the credit-based flow control mechanism.

3.3.2 Exception Handling for Data Messages

The rules in this section define the exception processing to be performed by a receiver of an erroneous Data Message.

In case of a Data Message with syntax error, meaning that the coding of the Data Packet is not consistent with this specification:

- If the receiver is the NFCC, it SHALL send a CORE_INTERFACE_ERROR_NTF message with a Status value of STATUS_SYNTAX_ERROR.
- If the receiver is the DH, it SHALL ignore the Data Message.

3.4 Packet Formats

3.4.1 Common Packet Header

All Packets have a common header, consisting of an MT field and a PBF field, as shown in Figure 6.

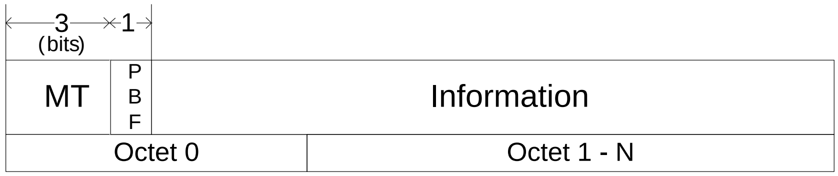

Figure 6: NCI Core Packet Format

Message Type (MT)

The MT field indicates the contents of the Packet and SHALL be a 3-bit field containing one of the values listed in Table 2. The content of the Information field is dependent on the value of the MT field. The receiver of an MT designated as RFU SHALL silently discard the packet.

Table 2: MT Values

MT	Description
000b	Data Packet- Section 3.4.3.
001b	Control Packet - Command Message as a payload- Section 3.4.2
010b	Control Packet - Response Message as a payload- Section 3.4.2
011b	Control Packet – Notification Message as a payload - Section 3.4.2
100b-111b	RFU

Packet Boundary Flag (PBF)

The Packet Boundary Flag (PBF) is used for Segmentation and Reassembly and SHALL be a 1 bit field containing one of the values listed in Table 3.

Table 3: PBF Values

PBF	Description
0b	The Packet contains a complete Message, or the Packet contains the last segment of a segmented Message
1b	The Packet contains a segment of a Message which is not the last segment.

The following rules apply to the PBF flag in Packets:

- If the Packet contains a complete Message, the PBF SHALL be set to 0b.
- If the Packet contains the last segment of a segmented Message, the PBF SHALL be set to 0b.

- If the packet does not contain the last segment of a segmented Message, the PBF SHALL be set to 1b.

See Section 3.5 for more details about Segmentation and Reassembly.

3.4.2 Format of Control Packets

The Control Packet structure is detailed in Figure 7.

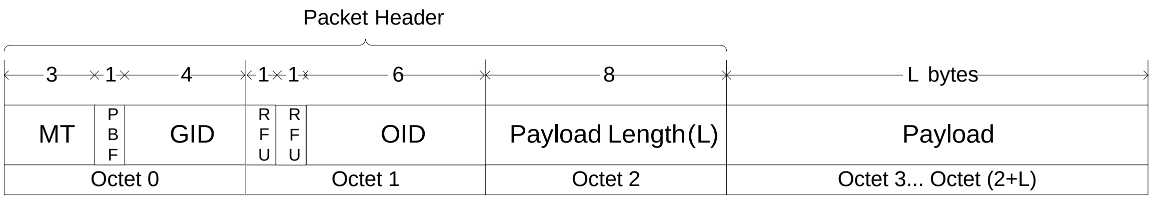

Figure 7: Control Packet Structure

Each Control Packet SHALL have a 3-octet Packet Header and MAY have additional payload for carrying a Control Message or a segment of Control Message.

NOTE In the case of an ‘empty’ Control Message, only the Packet Header is sent.

Message Type (MT)

Refer to Table 2 for details of the MT field.

Packet Boundary Flag (PBF)

Refer to Table 3 for details of the PBF field.

Group Identifier (GID)

The NCI supports Commands, Responses, and Notifications that are categorized according their individual groups. The Group Identifier (GID) indicates the categorization of the message and SHALL be a 4-bit field containing one of the values listed in Table 102.

All GID values not defined in Table 102 are RFU.

Opcode Identifier (OID)

The Opcode Identifier (OID) indicates the identification of the Control Message and SHALL be a 6-bit field that is a unique identification of a set of Command, Response, or Notification Messages within the group (GID). OID values are defined along with the definition of the respective Control Messages described in Table 102.

Payload Length (L)

The Payload Length SHALL indicate the number of octets present in the payload. The Payload Length field SHALL be an 8-bit field containing a value from 0 to 255.

3.4.3 Format of Data Packets

The Data Packet structure is detailed in Figure 8.

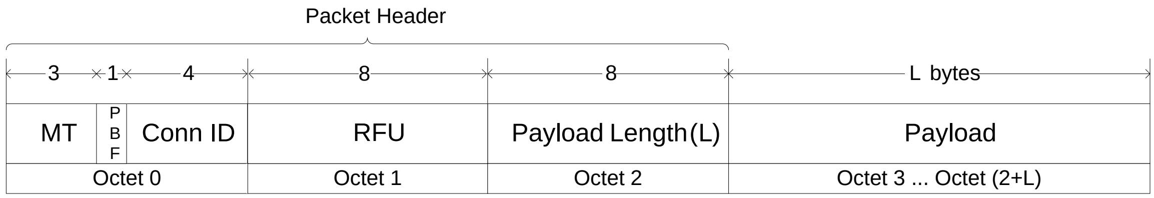

Figure 8: Data Packet Structure

Each Data Packet SHALL have a 3-octet Packet Header and MAY have additional Payload for carrying a Data Message or a segment of a Data Message.

NOTE In the case of an ‘empty’ Data Message, only the Packet Header is sent.

Message Type (MT)

Refer to Table 2 for details of the MT field.

NOTE MT always contains 000b to indicate a Data Packet, as defined in Table 2.

Packet Boundary Flag (PBF)

Refer to Table 3 for details of the PBF field.

Connection Identifier (Conn ID)

The Connection Identifier (Conn ID) SHALL be used to indicate the previously setup Logical Connection to which this data belongs. Refer to Section 4.4 for details on setting up a Logical Connection and the assignment of the Conn ID. The Conn ID is a 4-bit field containing a value from 0 to 15.

Payload Length (L)

The Payload Length field indicates the number of Payload octets present. The Payload Length field is an 8-bit field containing a value from 0 to 255.

3.5 Segmentation and Reassembly

The Segmentation and Reassembly functionality SHALL be supported by both the DH and the NFCC.

Segmentation and Reassembly of Messages SHALL be performed independently for Control Packets and Data Packets of each Logical Connection.

Any NCI Transport Mapping is allowed to define a fixed Maximum Transmission Unit (MTU) size in octets. If such a Mapping is defined and used, then if either DH or NFCC needs to transmit a Message (either Control or Data Message) that would generate a Packet (including Packet Header) larger than the MTU, the Segmentation and Reassembly (SAR) feature SHALL be used on the Message.

The following rules apply to segmenting Control Messages:

- For each segment of a Control Message, the header of the Control Packet SHALL contain the same MT, GID and OID values.
- **From DH to NFCC:** The Segmentation and Reassembly feature SHALL be used when sending a Command Message from the DH to the NFCC that would generate a Control Packet with a payload larger than the “Max Control Packet Payload Size” reported by the NFCC at initialization (see Section 4.2). Each segment of a Command Message except for the last SHALL contain a payload with the length of “Max Control Packet Payload Size”.
- **From NFCC to DH:** When an NFCC sends a Control Message to the DH, regardless of the length, it MAY segment the Control Message into smaller Control Packets if needed for internal optimization purposes.

The following rules apply to segmenting Data Messages:

- For each segment of a Data Message, the header of the Data Packet SHALL contain the same MT and Conn ID.
- **From DH to NFCC:** If a Data Message payload size exceeds the Max Data Packet Payload Size of the connection, then the Segmentation and Reassembly feature SHALL be used on the Data Message.
- **From NFCC to DH:** When an NFCC sends a Data Message to the DH, regardless of the payload length, it MAY segment the Data Message into smaller Data Packets for any internal reason; for example, for transmission buffer optimization.

3.6 Logical Connections

Logical Connections are used to exchange Data Messages between the DH and the NFCC. Logical Connections provide a common context for related Data Messages. Depending on the information exchanged during Logical Connection establishment, the NFCC can be the endpoint of the Data communication, or it has to forward the Payload of the Data Messages to or from a Remote NFC Endpoint or NFCEE (the later two being the main use cases for Logical Connections).

NOTE The communication method between the NFCC and the Remote Endpoint or between the NFCC and an NFCEE is out of scope of NCI.

A Logical Connection is set up through negotiation between the NFCC and the DH as described in Section 4.4. The following is an overview of the Logical Connection concept:

Dynamic Logical Connection:

- The DH MAY create a Dynamic Logical Connection.
- The NFCC MAY reject an incoming connection request.
- An identifier (Conn ID) will be assigned by the NFCC to identify the Dynamic Logical Connection, and this remains valid for the life of the Dynamic Logical Connection. The Conn ID is released when the Dynamic Logical Connection is closed.
- The DH can close a Dynamic Logical Connection.
- Data can be transported only after the Dynamic Logical Connection has been successfully created.
- Both the DH and the NFCC SHALL ignore Data Packets with unassigned Conn IDs.

Static RF Connection:

- The Static RF Connection exists after NFCC Initialization without needing to be created using the connection Control Messages defined in Section 4.4.2 and is never closed,
- The Initial Number of Credits and Max Data Packet Payload Size of the Static RF Connection are (re)established by the NFCC each time it sends RF_INTF_ACTIVATED_NTF
- The DH SHALL NOT send data over the Static RF Connection if there is no active RF Interface. See Section 7.3.
- If there is no active RF Interface, both the DH and the NFCC SHALL ignore Data Packets with the Conn ID of the Static RF Connection.

NOTE All RF Interfaces included in this version of the specification use only the Static RF Connection. However, Dynamic Logical Connections may be used by RF Interfaces included in future versions of this specification.

The Conn ID has a range of 0 to 15 (see Table 4). The Conn ID of 0 is reserved for the Static RF Connection used for RF communication, which exists following NFCC Initialization but that is not available for use unless an RF Interface is activated. Each RF Interface defines whether the Static RF Connection is used and whether Dynamic Logical Connections are permitted.

Table 4: Conn ID

Conn ID	Description
0000b	Static RF Connection between the DH and a Remote NFC Endpoint
0001b-1111b	Dynamically assigned by the NFCC

4 NCI Core Control Messages

Following are the descriptions of the Commands, Responses, and Notifications that are part of the NCI Core.

4.1 Reset of NFCC

These Control Messages are used to reset the NFCC.

Table 5: Control Messages to Reset the NFCC

CORE_RESET_CMD			
Payload Field(s)	**Length**	**Value/Description**	
Reset Type	1 Octet	0x00	Keep Configuration Reset the NFCC and keep the NCI RF Configuration (if supported).
		0x01	Reset Configuration Reset the NFCC including the NCI RF Configuration.
		0x02 – 0xFF	RFU

CORE_RESET_RSP		
Payload Field(s)	**Length**	**Value/Description**
Status	1 Octet	See Table 94.
NCI Version	1 Octet	See Table 6.
Configuration Status	1 Octet	See Table 7.

CORE_RESET_NTF			
Payload Field(s)	**Length**	**Value/Description**	
Reason Code	1 Octet	0x00	Unspecified reason
		0x01-0x9F	RFU
		0xA0-0xFF	For proprietary use
Configuration Status	1 Octet	See Table 7.	

Table 6: NCI Version

NCI Version Identifier	Definition
0x10	NCI Version 1.0
Other values	RFU

Table 7: Configuration Status

Value	Definition
0x00	NCI RF Configuration has been kept
0x01	NCI RF Configuration has been reset
0x02-0xFF	RFU

The NCI Version parameter SHALL be encoded as an 8-bit field consisting of two 4-bit unsigned values representing the major and minor release levels of this specification. The most significant 4 bits SHALL denote the major release level. The least significant 4 bits SHALL denote the minor release level of this specification.

The DH SHALL continue the communication if it supports the major version reported by the NFCC and it SHALL NOT use Commands, RFU values, or RFU fields from a greater minor version than reported by the NFCC.

The CORE_RESET_CMD is issued by the DH to reset the NFCC. This Command MAY be issued anytime following power-up of the NFCC. After a reset, the NCI initialization as described in Section 4.2 SHALL be performed.

On completion of the reset procedure, the NFCC SHALL send a CORE_RESET_RSP informing the DH that the NFCC has been reset. The Status SHALL be STATUS_OK.

The CORE_RESET_CMD allows defining different reset types using the Reset Type parameter. The Configuration Status parameters in CORE_RESET_RSP and CORE_RESET_NTF inform the DH about the status of the NCI RF configuration after the reset.

NOTE This allows different NFCC implementations: Some NFCCs may have persistent memory for the NCI RF Configuration and therefore do not require the DH to re-configure after a reset. Others may not have persistent memory for the NCI RF Configuration. The DH can force a reset of the configuration by using the Reset Type parameter. The DH knows, based on the Configuration Status value, whether it needs to configure the NFCC after the reset or not.

If Reset Type has been set to 0x00, the Configuration Status in CORE_RESET_RSP SHALL be set to either 0x00 or 0x01.

If Reset Type has been set to 0x01, the Configuration Status in CORE_RESET_RSP SHALL be set to 0x01.

For all Configuration Status values, all data in Buffers used for NCI Data and Control Packet exchange SHALL be deleted and the Buffer SHALL be freed.

In this context NCI RF Configuration SHALL comprise:

- The Listen Mode Routing Table (see Section 6.3)
- All Configuration Parameters (see Table 101 for a list of Configuration Parameters)
- RF Interface Mapping configuration (see Section 6.2)

If Configuration Status is set to 0x01 (in either CORE_RESET_RSP or CORE_RESET_NTF), the NCI RF Configuration SHALL have been reset, which includes:

- Removing all entries of the Listen Mode Routing Table
- Reverting all Configuration Parameter to their default values

- Erasing the RF Interface Mapping configuration

If Configuration Status is set to 0x00, the NCI RF Configuration SHALL be the same as before the reset. In this case, the NFCC internal mapping of NFCEEs to NFCEE IDs SHALL also be unchanged (as otherwise the Listen Mode Routing Table would be corrupted).

The NFCC MAY also reset itself (without having received a CORE_RESET_CMD); e.g., in the case of an internal error. In these cases, the NFCC SHALL inform the DH with the CORE_RESET_NTF. The Reason code SHALL reflect the internal reset reason and the Configuration Status the status of the NCI RF Configuration.

4.2 Initialization of NFCC

These Control Messages are used to initialize the NFCC.

Table 8: Control Messages to Initialize the NFCC

<table>
<tr><th colspan="3">CORE_INIT_CMD</th></tr>
<tr><th>Payload Field(s)</th><th>Length</th><th>Value/Description</th></tr>
<tr><td colspan="3">Empty payload</td></tr>
</table>

<table>
<tr><th colspan="4">CORE_INIT_RSP</th></tr>
<tr><th>Payload Field(s)</th><th>Length</th><th colspan="2">Value/Description</th></tr>
<tr><td>Status</td><td>1 Octet</td><td colspan="2">See Table 94.</td></tr>
<tr><td>NFCC Features</td><td>4 Octets</td><td colspan="2">See Table 9.</td></tr>
<tr><td>Number of Supported RF Interfaces</td><td>1 Octet</td><td colspan="2">Number of Supported RF Interface fields to follow (n).</td></tr>
<tr><td>Supported RF Interface [1..n]</td><td>1 Octet</td><td colspan="2">See Table 99.
NOTE If supported, the pseudo interface NFCEE Direct RF Interface has to be reported as well.</td></tr>
<tr><td rowspan="2">Max Logical Connections</td><td rowspan="2">1 Octet</td><td>0x00 – 0x0E</td><td>Maximum number of Dynamic Logical Connections supported by the NFCC.</td></tr>
<tr><td>0x0F – 0xFF</td><td>RFU</td></tr>
<tr><td>Max Routing Table Size</td><td>2 Octets</td><td colspan="2">Indicates the maximum amount of data in Octets that are possible in a routing configuration (see Section 6.3). If Listen Mode Routing is not supported, then the value SHALL be 0x0000.</td></tr>
<tr><td>Max Control Packet Payload Size</td><td>1 Octet</td><td colspan="2">Indicates the maximum payload length of a NCI Control Packet that the NFCC is able to receive.
Valid range is 32 to 255.
NOTE All Control Messages exchanged prior to this have a length that is smaller than 32 octets.</td></tr>
<tr><td>Max Size for Large Parameters</td><td>2 Octets</td><td colspan="2">The maximum size in octets for the sum of the sizes of PB_H_INFO and LB_H_INFO_RESP parameter values.</td></tr>
</table>

CORE_INIT_RSP		
Payload Field(s)	**Length**	**Value/Description**
Manufacturer ID	1 Octet	IC Manufacturer ID, as defined in [MANU]. If this information is not available, the NFCC SHALL return 0x00.
Manufacturer Specific Information	4 Octets	This field contains NFCC manufacturer specific information like chip version, firmware version, etc., encoded in a manufacturer-specific mode. If this information is not available or if the Manufacturer ID is set to 0x00, the NFCC SHALL return all octets containing 0x00.

The CORE_INIT_CMD is used by the DH to initialize the NFCC. This Command SHALL be issued following indication that the NFCC has been successfully reset (see Section 4.1), and at no other time.

On execution of the Command, the NFCC SHALL send a CORE_INIT_RSP to inform the DH that the NFCC has executed the Command. If the initialization was successful, the Status SHALL be STATUS_OK. If the NFCC is unable to execute the Command, the Status SHALL be set to STATUS_FAILED (see Table 94), and the other parameters of the CORE_INIT_RSP SHALL be ignored by the DH.

No other Command that is defined in this specification except for CORE_RESET_CMD or CORE_INIT_CMD SHALL be sent prior to the NFCC being successfully initialized.

Prior to initialization, if the DH knows the manufacturer ID of the NFCC by some means outside the scope of NCI, then proprietary Commands MAY be sent, but otherwise they SHALL NOT be sent.

Following successful initialization, proprietary Commands MAY be sent by the DH, and proprietary Responses and Notifications MAY be sent by the NFCC.

After sending a CORE_INIT_CMD, the DH SHALL discard any information that may have been previously provided by the NFCC concerning discovery requests from NFCEE(s).

Table 9: NFCC Features

	Bit Mask								**Description**
	b7	**b6**	**b5**	**b4**	**b3**	**b2**	**b1**	**b0**	
Octet 0	0	0	0	0	0				RFU
						X	X		Discovery Configuration Mode: This parameter informs the DH how the NFCC uses the RF Configuration provided by the DH if multiple NFCEEs exist in the NFC Forum Device. If set to 00b, the DH is the only entity that configures the NFCC. If set to 01b, the NFCC can receive configurations from the DH and other NFCEEs. This implies that the NFCC can manage or merge multiple sets of RF configuration parameters. Other values are RFU. See Section 7.1 for more details on the use of this parameter.
								X	Discovery Frequency configuration in RF_DISCOVER_CMD supported, if the bit is set to 1b. If set to 0b the Discovery Frequency value is ignored and the value of 0x01 SHALL be used by the NFCC.
Octet 1	0	0	0	0					RFU
					X				AID based routing; Supported if the bit is set to 1b (NFCC supports 7816-4 Command parsing of SELECT command).
						X			Protocol based routing; Supported if bit is set to 1b.
							X		Technology based Routing; Supported if the bit is set to 1b.
								0	RFU
Octet 2	0	0	0	0	0	0			RFU
							X		Switched Off state; Supported if the bit is set to 1b.
								X	Battery Off state; Supported if the bit is set to 1b.
Octet 3	0	0	0	0	0	0	0	0	Octet 3 is reserved for proprietary capabilities

If no routing type is supported in Octet 1 of the NFCC Features, then the NFCC does not support Listen Mode Routing. In that case the DH SHALL NOT use any Command defined in Section 6.3.

4.3 NFCC Configuration

4.3.1 Setting the Configuration

These Control Messages are used to set configuration parameters on the NFCC.

Table 10: Control Messages for Setting Configuration Parameters

CORE_SET_CONFIG_CMD				
Payload Field(s)	**Length**	**Value/Description**		
Number of Parameters	1 Octet	The number of Parameter fields to follow (n).		
Parameter [1..n]	m+2 Octets	ID	1 Octet	The identifier of the configuration parameter. See Table 101 for a list of IDs.
		Len	1 Octet	The length of Val (m). If Len is set to 0x00, then the Val field is omitted, and the NFCC SHALL set the configuration parameter to its default value.
		Val	m Octets	The value of the configuration parameter.

CORE_SET_CONFIG_RSP		
Payload Field(s)	**Length**	**Value/Description**
Status	1 Octet	See Table 94
Number of Parameters	1 Octet	The number of Parameter ID fields to follow (n). Value SHALL be 0x00 and no Parameter IDs listed unless Status = STATUS_INVALID_PARAM.
Parameter ID [0..n]	1 Octet	The identifier of the invalid configuration parameter. See Table 101 for a list of IDs.

All configuration parameters within the NFCC are set to default values, but the DH MAY use CORE_SET_CONFIG_CMD to change these values. The NFCC responds with CORE_SET_CONFIG_RSP, and the Status indicates whether the setting of these configuration parameters was successful or not. A Status of STATUS_OK SHALL indicate that all configuration parameters have been set to these new values within the NFCC.

If the DH tries to set a parameter that is not applicable for the NFCC, the NFCC SHALL respond with a CORE_SET_CONFIG_RSP with a Status field of STATUS_INVALID_PARAM and including one or more invalid Parameter ID(s). All other configuration parameters SHALL have been set to the new values within the NFCC.

Based on the maximum size of the payload of a Control Message and on the length of the Payload Fields in CORE_GET_CONFIG_RSP, the maximum length of a parameter is limited to 251 octets. The parameter length fields (Parameter Len [n]) SHALL be set to a value between 0x00 and 0xFB. The values 0xFC - 0xFF are RFU.

4.3.2 Retrieve the Configuration

These Control Messages are used by the DH to retrieve current configuration parameters of the NFCC.

Table 11: Control Messages for Reading Current Configuration

CORE_GET_CONFIG_CMD		
Payload Field(s)	**Length**	**Value/Description**
Number of Parameters	1 Octet	The number of Parameter ID fields to follow (n).
Parameter ID [1..n]	1 Octet	The identifier of the configuration parameter. See Table 101 for a list of IDs.

<table>
<tr><th colspan="5">CORE_GET_CONFIG_RSP</th></tr>
<tr><th>Payload Field(s)</th><th>Length</th><th colspan="3">Value/Description</th></tr>
<tr><td>Status</td><td>1 Octet</td><td colspan="3">See Table 94.</td></tr>
<tr><td>Number of Parameters</td><td>1 Octet</td><td colspan="3">The number of Parameter fields to follow (n).</td></tr>
<tr><td rowspan="3">Parameter [1..n]</td><td rowspan="3">m+2 Octets</td><td>ID</td><td>1 Octet</td><td>The identifier of the configuration parameter.
See Table 101 for a list of IDs.</td></tr>
<tr><td>Len</td><td>1 Octet</td><td>Length of Val (m).
If Len = 0x00, then Val field is omitted.</td></tr>
<tr><td>Val</td><td>m Octets</td><td>Value of the configuration parameter.</td></tr>
</table>

The DH MAY use CORE_GET_CONFIG_CMD to retrieve the current configuration parameters of the NFCC. If the NFCC is able to respond with all requested parameters, the NFCC SHALL respond with the CORE_GET_CONFIG_RSP with a Status of STATUS_OK.

NOTE In the STATUS_OK case, parameter values can be 'empty' and therefore have a Len of 0x00 and no Val field (e.g., if the default value of the parameter is 'empty').

If the DH tries to retrieve any parameter(s) that are not available in the NFCC, the NFCC SHALL respond with a CORE_GET_CONFIG_RSP with a Status field of STATUS_INVALID_PARAM, containing each unavailable Parameter ID with a Parameter Len field of value zero. In this case, the CORE_GET_CONFIG_RSP SHALL NOT include any parameter(s) that are available on the NFCC.

NOTE This failure case (STATUS_INVALID_PARAM) is intended for when the DH tries to retrieve a Parameter that is not supported at all by the NFCC.

NOTE After receiving the list of unavailable Parameters, the DH can assume that the other Parameters requested in the CORE_GET_CONFIG_CMD are available, and the DH may initiate another CORE_GET_CONFIG_CMD to retrieve those Parameters.

If CORE_GET_CONFIG_RSP message payload size with all requested parameters would exceed the maximum Control Message payload size (255 octets), then the NFCC SHALL only send the limited set of parameters that will not exceed the maximum payload size. In this case, the Status field SHALL have a value of STATUS_MESSAGE_SIZE_EXCEEDED. The DH MAY retrieve any unreturned parameters by sending another CORE_GET_CONFIG_CMD requesting only those specific IDs.

If both of these conditions happen: 1) the DH tries to retrieve unavailable Parameters and 2) Response would exceed maximum Control Message payload size, then case (1) has higher priority; i.e., the NFCC SHALL return a status of STATUS_INVALID_PARAM to the DH.

4.4 Logical Connection Management

4.4.1 Destination Type

Destination Types are used during Logical Connection creation. They identify the type of an entity in the system: NFCC, NFCEE, or Remote NFC Endpoint.

Destination Types may also identify specific functionality within an entity, like the loop-back function within the NFCC.

Valid Destination Types are defined in Table 12.

Table 12: Destination Types

Destination Type	Description
0x00	RFU
0x01	NFCC - Loop-back This is used to create a Logical Connection between the DH and the NFCC for loop-back mode. See Section 12.1.
0x02	Remote NFC Endpoint This is used to create a Dynamic Logical Connection for communicating with a Remote NFC Endpoint.
0x03	NFCEE This is used to create a Dynamic Logical Connection for communicating with an NFCEE.
0x04 – 0xC0	RFU
0xC1 – 0xFE	Proprietary This is for proprietary use; e.g., to create a Logical Connection between the DH and the NFCC for firmware update purposes.
0xFF	RFU

4.4.2 Connection Creation

These Control Messages are used when the DH creates a Dynamic Logical Connection to the NFCC.

Table 13: Control Messages for DH Connection Creation

<table>
<tr><th colspan="5">CORE_CONN_CREATE_CMD</th></tr>
<tr><th>Payload Field(s)</th><th>Length</th><th colspan="3">Value/Description</th></tr>
<tr><td>Destination Type</td><td>1 Octet</td><td colspan="3">See Table 12.</td></tr>
<tr><td>Number of Destination-specific Parameters</td><td>1 Octet</td><td colspan="3">Based on the Destination Type of the connection, the number of Destination-specific Parameter fields following (n).</td></tr>
<tr><td rowspan="3">Destination-specific Parameter [0..n]</td><td rowspan="3">m+2 Octets</td><td>Type</td><td>1 Octet</td><td>Type of the Destination-specific Parameter See Table 15.</td></tr>
<tr><td>Length</td><td>1 Octet</td><td>Length of Value (m)</td></tr>
<tr><td>Value</td><td>m Octets</td><td>Value of the Destination-specific Parameter</td></tr>
</table>

CORE_CONN_CREATE_RSP		
Payload Field(s)	**Length**	**Value/Description**
Status	1 Octet	See Table 94.
Max Data Packet Payload Size	1 Octet	A number from 1 – 255
Initial Number of Credits	1 Octet	See Table 14.
Conn ID	1 Octet	The four least significant bits of the octet SHALL contain the Conn ID as defined in Table 4. The four most significant bits are RFU.

Table 14: Initial Number of Credits

Value	Description
0x00– 0xFE	Number of credits
0xFF	Data flow control is not used

Table 15: Destination-specific Parameters

Type	Length	Value
0x00	2 Octets	First octet: An RF Discovery ID (see Table 53). Second octet: RF Protocol (see Table 98).
0x01	2 Octets	First octet: NFCEE ID as defined in Table 84. The value of 0x00 (DH-NFCEE ID) SHALL NOT be used. Second octet: NFCEE Interface Protocol. See Table 100.
0x02-0x9F		RFU
0xA0-0xFF		For proprietary use

To create a Logical Connection, the DH sends the CORE_CONN_CREATE_CMD to the NFCC and identifies the Destination Type (see Section 4.4.1) to which this Logical Connection will apply.

The combination of Destination Type and Destination Specific Parameters SHALL uniquely identify a single destination for the Logical Connection.

If the Destination Type is 0x01, then no Destination-specific Parameters are allowed.

If the Destination Type is that of a Remote NFC Endpoint (0x02), then only the Destination-specific Parameter with Type 0x00 or proprietary parameters as defined in Table 15 SHALL be present.

NOTE This version of the specification does not actually use Dynamic Logical Connections for communication with Remote NFC Endpoints.

If the Destination Type is that of an NFCEE (0x03), then only the Destination-specific Parameter with Type 0x01 or proprietary parameters as defined in Table 15 SHALL be present.

After receipt of a CON_CREATE_CMD, the NFCC must determine whether to accept the request and, if so, continue with the actual creation of the Logical Connection. The NFCC SHALL respond with a CORE_CONN_CREATE_RSP where the Status SHALL indicate whether the Logical Connection has been established (STATUS_OK) or failed. In failure case and if no different Status value is specified for the concrete case, the Status value SHALL be STATUS_REJECTED.

NOTE For example, NFCEE Interface activation (Section 10) defines a specific Status value for the NFCEE Interface activation failure.

If the Logical Connection has been established, the Conn ID SHALL indicate the Connection Identifier for this Logical Connection

The CORE_CONN_CREATE_RSP SHALL contain the actual Conn ID used for the Logical Connection.

If data flow control is requested by the NFCC, the NFCC also indicates the Initial Number of Credits that the NFCC has allocated for this connection’s receive path. If data flow control is not requested, the NFCC SHALL set the parameter value for Initial Number of Credits to 0xFF.

4.4.3 Connection Closure

These Control Messages are used to close a Dynamic Logical Connection.

Table 16: Control Messages for Connection Closure

CORE_CONN_CLOSE_CMD		
Payload Field(s)	**Length**	**Value/Description**
Conn ID	1 Octet	The identifier of the connection to be closed. The four least significant bits of the octet SHALL contain the Conn ID as defined in Table 4. The four most significant bits are RFU.

CORE_CONN_CLOSE_RSP		
Payload Field(s)	**Length**	**Value/Description**
Status	1 Octet	See Table 94.

To close a Logical Connection, the DH sends the CORE_CONN_CLOSE_CMD to the NFCC indicating the Conn ID to be closed.

On receiving a CORE_CONN_CLOSE_CMD for an existing connection, the NFCC SHALL accept the connection closure request by sending a CORE_CONN_CLOSE_RSP with a Status of STATUS_OK, and the Logical Connection is closed.

If there is no connection associated to the Conn ID in the CORE_CONN_CLOSE_CMD, the NFCC SHALL reject the connection closure request by sending a CORE_CONN_CLOSE_RSP with a Status of STATUS_REJECTED. When a DH receives this status, it should assume that the Conn ID is unknown, and therefore the connection no longer exists on the NFCC and SHOULD proceed as in a normal closing.

The DH SHALL clean up all resources associated with the Conn ID after it sends the CORE_CONN_CLOSE_CMD. The NFCC SHALL clean up all resources associated with the Conn ID after receiving the CORE_CONN_CLOSE_CMD.

NOTE ‘Clean up all resources’ means deleting all data from buffers related to the closed Logical Connection, setting *NFCC_Credits_Avail* to initial value, and releasing the Conn ID for later use.

NOTE The DH should make sure that the last Data Message has arrived at the NFCC before sending the CORE_CONN_CLOSE_CMD. The NFCC should attempt to send any pending data to a Remote NFC Endpoint or NFCEE even though the corresponding connection is being closed.

4.4.4 Connection Credit Management

These Control Messages are used to manage the credits on a Logical Connection that uses credit-based flow control.

Table 17: Control Messages for Connection Credit Management

<table>
<tr><th colspan="5">CORE_CONN_CREDITS_NTF</th></tr>
<tr><th>Payload Field(s)</th><th>Length</th><th colspan="3">Value/Description</th></tr>
<tr><td>Number of Entries</td><td>1 Octet</td><td colspan="3">Number of Entry fields to follow (n).</td></tr>
<tr><td rowspan="2">Entry [1..n]</td><td rowspan="2">2 Octet</td><td>Conn ID</td><td>1 Octet</td><td>The identifier of the Logical Connection for which credits are given.
The four least significant bits of the octet SHALL contain the Conn ID as defined in Table 4. The four most significant bits are RFU.</td></tr>
<tr><td>Credits</td><td>1 Octet</td><td>The number of credits given.</td></tr>
</table>

For each Logical Connection, the DH stores the initial credits value received as part of the connection setup (Initial Number of Credits parameter) in a variable, namely *NFCC_Credits_Avail*, which is used to track the number of Data Packets that can be sent to the NFCC.

NOTE If there are not enough credits to send the whole Data Message, the DH is allowed to send as many Data Packets as number of credits available.

When the DH wants to send a Data Packet on a Logical Connection and flow control is enabled, the DH SHALL check that the *NFCC_Credits_Avail* variable for that connection is greater than 0. If so, the DH SHALL reduce the *NFCC_Credits_Avail* by 1 and transfer the Data Packet over the Logical Connection. The DH SHALL NOT send any Data Packet on a connection when the corresponding *NFCC_Credits_Avail* is 0.

If the NFCC receives a Data Packet that was subject to credit-based flow control, it needs to tell the DH when the buffer is again available for use. It does this by sending CORE_CONN_CREDITS_NTF. When the DH receives this Notification, it SHALL add the credits to variable *NFCC_Credits_Avail* for the appropriate Logical Connection(s).

The DH SHALL set the *NFCC_Credits_Avail* variable to the initial value on the following scenarios:

- A Dynamic Logical Connection is created. See Sections 4.4.2.
- For the Static RF Connection, if an RF Interface Activated Notification is received, see Section 7.3.1.

4.5 Generic Error

This Notification is used to inform the DH about a generic error situation.

Table 18: Control Messages for Generic Error

CORE_GENERIC_ERROR_NTF		
Payload Field(s)	**Length**	**Value/Description**
Status	1 Octet	See Table 94.

This Notification is used in error situations when the error can not be notified using an error status in a Response Message. This Notification SHALL NOT be used to report error scenarios related to NFCEE or RF Interface communication using Logical Connections.

To notify a generic error situation, the NFCC SHALL send CORE_GENERIC_ERROR_NTF to the DH with the Status code identifying the error case.

4.6 Interface Error

These Control Messages are used to inform the DH about an RF or NFCEE Interface communication specific error situation.

Table 19: Control Messages for Interface Error

CORE_INTERFACE_ERROR_NTF		
Payload Field(s)	**Length**	**Value/Description**
Status	1 Octet	See Table 94.
Conn ID	1 Octet	The identifier of the Logical Connection where the error occurred. The four least significant bits of the octet SHALL contain the Conn ID as defined in Table 4. The four most significant bits are RFU.

This Notification is used in error situations when the error can not be notified using an error status in a Response Message.

This Notifcation SHALL only be used to notify error scenarios related to NFCEE or RF Interface communication using Logical Connections.

To notify a Logical Connection or RF / NFCEE Interface specific error situation, the NFCC SHALL send CORE_INTERFACE_ERROR_NTF to the DH with the Status code identifying the error and the affected Conn ID.

5 RF Communication

Communication with a Remote NFC Endpoint is performed by the use of RF Interfaces. RF Interfaces are logical entities on the NFCC which allow the DH to use specific layers in the protocol stack implemented on the NFCC. The DH can only communicate with a Remote NFC Endpoint via an RF Interface that has been activated during the Discovery process.

Section 5.1 gives an overview about the available RF Interfaces.

Section 5.2 defines the NCI state machine for RF Communication.

Section 5.3 defines a mechanism to get information about an external RF field.

5.1 RF Interface Architecture

All data exchanged between the DH and a Remote NFC Endpoint flows through an RF Interface, designated as the “Active RF Interface”.

Figure 9 shows an overview of the RF Interfaces.

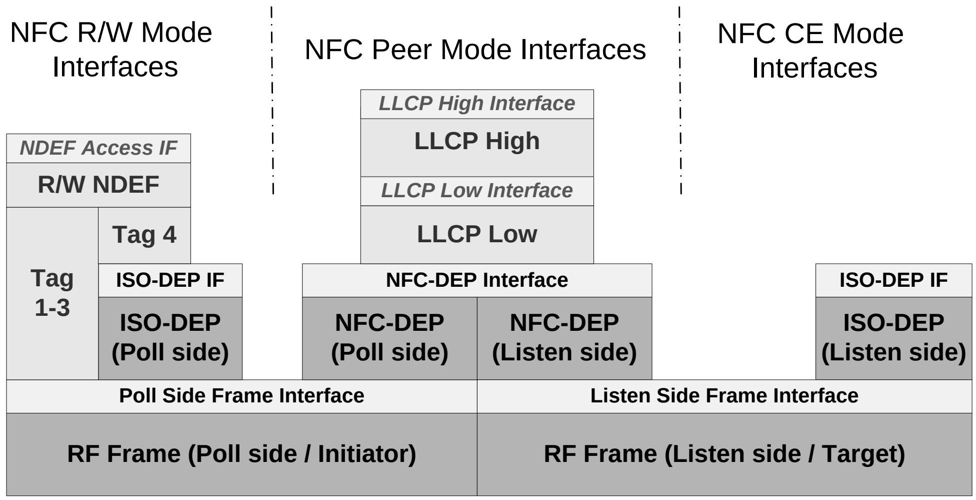

Figure 9: RF Interface Architecture

NOTE The NDEF Access, LLCP High and LLCP Low RF Interfaces do not exist in the NCI specification version 1.0. They are anticipated to be added in a future version.

The following should be noted:

- The ISO-DEP RF Interface is applicable for both Reader/Writer Mode and Card Emulation Mode.
- Specific to the Frame RF Interface
    - The Poll Side Frame RF Interface is applicable for both Reader/Writer Mode and the NFC-DEP Initiator side of the Peer Mode.
    - Listen Side Frame RF Interface is applicable for both Card Emulation Mode and the NFC-DEP Target side of the Peer Mode.

An RF Interface is automatically activated by the NFCC in the following cases:

- When in Poll Mode, a single Remote NFC Endpoint supporting a single protocol has been discovered.

- In Listen Mode, the NFCC has been discovered/selected by a Remote NFC Endpoint.

In both cases, the RF Interface to activate is determined by the current mapping between RF Protocols and RF Interfaces that can be configured using the RF_DISCOVER_MAP_CMD Command defined in Section 6.2.

If, in Poll Mode, multiple Remote NFC Endpoints or a Remote NFC Endpoint supporting more than one RF Protocol have been discovered, the DH has to select a Remote NFC Endpoint and RF Protocol using the RF_DISCOVER_SELECT_CMD Command. After receipt of this Command, the NFCC activates the RF Interface specified in the RF_DISCOVER_SELECT_CMD Command.

The activation of an RF Interface depends on the Mode:

- For Poll Mode, one Remote NFC Endpoint must be selected (if only one Remote NFC Endpoint is detected, it is selected automatically) and, depending on the RF Interface to activate, the NFCC may need to first establish one or more lower-level protocol(s) before activating an RF interface. In future extensions of NCI, the NFCC may even have to start exchanging some data (for instance, to check if the remote NFC Endpoint stores NDEF-compliant data) before activating the RF Interface.
- For Listen Mode, the NFCC may only have to be detected/selected by a Remote NFC Endpoint before activating the RF interface. Or, it may need to first wait for the Remote NFC Endpoint to establish one or more lower-level protocol(s).

Section 8 describes dependencies and explains how the RF Interfaces are used.

5.2 State Machine

The NCI process for RF Communication is described using a state machine shown in Figure 10.

It is entirely optional for any DH or NFCC to implement the state machine explicitly. However all DH and NFCC implementations SHALL follow NCI behavior exactly *as if* they had implemented the state machine.

Each transition from one state to another is accompanied by an appropriate NCI Command, Response, or Notification, so both the DH and NFCC can always unambiguously know the current RF Communication state.

The following Control Messages SHALL NOT be sent unless permitted by the rules of the state machine:

- RF_DISCOVER_CMD, RF_DISCOVER_RSP, RF_DISCOVER_NTF
- RF_DISCOVER_SELECT_CMD, RF_DISCOVER_SELECT_RSP
- RF_INTF_ACTIVATED_NTF
- RF_DEACTIVATE_CMD, RF_DEACTIVATE_RSP, RF_DEACTIVATE_NTF
- CORE_INTERFACE_ERROR_NTF with a Conn ID of a Logical Connection that is used in the context of an RF Interface.

In addition, the following Control Messages, as specified in the corresponding sections, can only be sent in certain states:

- RF_NFCEE_ACTION_NTF (see Section 7.5)
- RF_PARAMETER_UPDATE_CMD/RSP (see Section 8.2.2.1)

- RF_T3T_POLLING_CMD/RSP/NTF (see Section 8.2.2.2)

If the NFCC or DH receives a Control Message in a state where that Control Message is not allowed, it SHALL treat this as a semantic error, as defined in 3.2.2.

Other Commands, Responses, and Notifications that are not explicitly forbidden by this specification MAY be exchanged in any state.

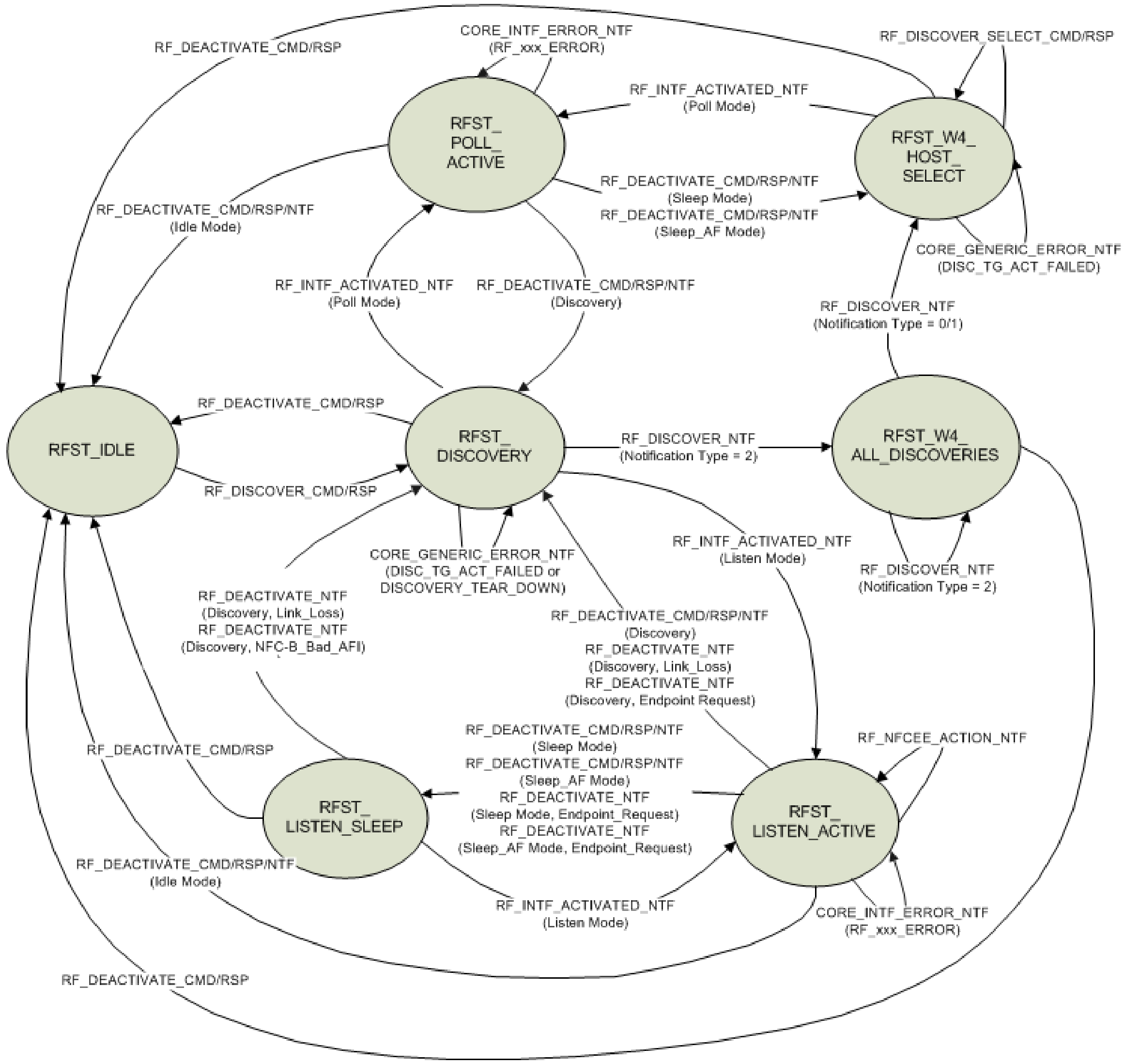

Figure 10: RF Communication State Machine

Upon successful DH and NFCC initialization (see Section 4.2), the RF Communication state machine SHALL be in **RFST_IDLE** state. Before the first transition out of **RFST_IDLE** state, the DH SHALL set the RF Communication Configuration as described in Section 6.

For Poll Mode, the RF Communication state machine relates to the Activities defined in [ACTIVITY] in the following way:

- Technology Detection is handled in **RFST_DISCOVERY** and **RFST_W4_ALL_DISCOVERIES.**
- Collision Resolution is handled in **RFST_DISCOVERY** and **RFST_W4_HOST_SELECT.**

- Device Activation is handled in **RFST_DISCOVERY**, **RFST_W4_ALL_DISCOVERIES**, and **RFST_W4_HOST_SELECT**, depending on the number of Remote NFC Endpoints discovered and the RF Interface mapping. Depending on the RF Interface, part of the Device Activation Activity is handled by the DH in **RFST_POLL_ACTIVE**.
- RF Data Exchange is handled in state **RFST_POLL_ACTIVE**
- Depending on the RF Interface, the Device Deactivation Activity is fully handled by the NFCC or split between the DH and the NFCC. The DH is responsible for handling its part in **RFST_POLL_ACTIVE**. The NFCC handles Device Deactivation when moving from state **RFST_POLL_ACTIVE** to either **RFST_DISCOVERY, RFST_IDLE**, or **RFST_W4_HOST_SELECT**.

The Listen Mode state machine defined in [ACTIVITY] is handled in states **RFST_DISCOVERY**, **RFST_LISTEN_ACTIVE**, and **RFST_LISTEN_SLEEP**. The IDLE state defined in [ACTIVITY] is hosted inside the **RFST_DISCOVERY** NCI state.

5.2.1 State RFST_IDLE

Discovery related configuration defined in Sections 6.1, 6.2, and 6.3 SHALL only be set while in this state.

NOTE Although CORE_SET_CONFIG_CMD may be sent in any state, parameters that relate to RF Discovery may only be included in state **RFST_IDLE**.

The NFCC SHALL turn the RF field OFF in this state. The greedy collection [ACTIVITY] is cleared in this state.

When the DH issues a valid RF_DISCOVER_CMD Command and the NFCC returns RF_DISCOVER_RSP with status STATUS_OK, the state is changed to **RFST_DISCOVERY**.

5.2.2 State RFST_DISCOVERY

In this state, the NFCC stays in Poll Mode and/or Listen Mode (based on the discovery configuration) until at least one Remote NFC Endpoint is detected or the RF Discovery Process is stopped by the DH.

The parameters and duty cycle for Poll Mode and Listen Mode are as configured by the DH.

The NFCC is expected to start filling up the greedy collection [ACTIVITY] in this state.

If discovered by a Remote NFC Endpoint in Listen mode, once the Remote NFC Endpoint has established any underlying protocol(s) needed by the configured RF Interface, the NFCC SHALL send RF_INTF_ACTIVATED_NTF (Listen Mode) to the DH and the state is changed to **RFST_LISTEN_ACTIVE**.

While polling, if the NFCC discovers more than one Remote NFC Endpoint, or a Remote NFC Endpoint that supports more than one RF Protocol, it SHALL start sending RF_DISCOVER_NTF messages to the DH. At this point, the state is changed to **RFST_W4_ALL_DISCOVERIES**.

While polling, if the NFCC discovers just one Remote NFC Endpoint that supports just one Protocol, the NFCC SHALL try to automatically activate it. The NFCC SHALL first establish any underlying protocol(s) with the Remote NFC Endpoint that are needed by the configured RF Interface. On completion, the NFCC SHALL activate the RF Interface and send RF_INTF_ACTIVATED_NTF (Poll Mode) to the DH. At this point, the state is changed to **RFST_POLL_ACTIVE**. If the protocol activation is not successful, the NFCC SHALL send CORE_GENERIC_ERROR_NTF to the DH with status DISCOVERY_TARGET_ACTIVATION_FAILED and stay in **RFST_DISCOVERY**.

In this state, the NFCC MAY detect a command during the RF communication, which forces it to come back to the IDLE state, as defined in the [ACTIVITY] Listen Mode state machine. If the RF Protocol deactivation is completed, the NFCC SHALL send CORE_GENERIC_ERROR_NTF (DISCOVERY_TEAR_DOWN), and the state will remain **RFST_DISCOVERY**.

NOTE RF Protocol deactivation while in state **RFST_DISCOVERY** can happen if during protocol activation in Listen Mode, a teardown command is received (e.g., an NFC-DEP RLS_REQ when waiting for a potential PSL_REQ).

If the DH sends RF_DEACTIVATE_CMD, the NFCC SHALL ignore the Deactivation Type parameter, stop the RF Discovery Process, and send RF_DEACTIVATE_RSP. The state will then change to **RFST_IDLE**.

5.2.3 State RFST_W4_ALL_DISCOVERIES

In this state, the NFCC has discovered more than one Remote NFC Endpoint or a Remote NFC Endpoint that supports more than one RF Protocol, while polling.

The NFCC SHOULD keep the RF field ON while in this state.

Discover notifications with Notification type set to 2 SHALL NOT change the state.

When the NFCC sends the last RF_DISCOVER_NTF (Notification Type not equal to 2) to the DH, the state is changed to **RFST_W4_HOST_SELECT**.

If the DH sends RF_DEACTIVATE_CMD, the NFCC SHALL ignore the Deactivation Type parameter, stop the RF Discovery Process and send RF_DEACTIVATE_RSP. The state will then change to **RFST_IDLE**.

5.2.4 State RFST_W4_HOST_SELECT

The NFCC SHOULD keep the RF field ON while in this state.

In this state the NFCC has its greedy collection intact and is waiting for the DH to select one of the Remote NFC Endpoints from the greedy collection [ACTIVITY] to activate. The greedy collection [ACTIVITY] may contain Remote NFC Endpoints that are in sleep state.

When the DH sends RF_DISCOVER_SELECT_CMD with a valid RF Discovery ID, RF Protocol and RF Interface, the NFCC SHALL try to activate the associated Remote NFC Endpoint (depending on the state of the Remote NFC Endpoint). The NFCC SHALL first establish any underlying protocol(s) with the Remote NFC Endpoint that are needed by the configured RF Interface. On completion, the NFCC SHALL activate the RF Interface and send RF_INTF_ACTIVATED_NTF (Poll Mode) to the DH. At this point, the state is changed to **RFST_POLL_ACTIVE**.

If the activation was not successful, the NFCC SHALL send CORE_GENERIC_ERROR_NTF to the DH with a Status of DISCOVERY_TARGET_ACTIVATION_FAILED and the state will remain as **RFST_W4_HOST_SELECT**.

If the DH sends RF_DEACTIVATE_CMD, the NFCC SHALL ignore the Deactivation Type parameter, stop the RF Discovery Process and send RF_DEACTIVATE_RSP. The state will then change to **RFST_IDLE**.

5.2.5 State RFST_POLL_ACTIVE

In this state the NFCC device is activated in Poll Mode.

The NFCC SHALL keep the RF field ON while in this state.

In this state an RF Interface is activated, which allows the NFCC to communicate with a Remote NFC Endpoint. The NFCC SHALL accept and send Data Messages to or from the DH as specified by the active RF Interface.

In this state the DH MAY send RF_DEACTIVATE_CMD (Sleep Mode or Sleep_AF Mode) to deactivate communication with the Remote NFC Endpoint. Depending on the activated interface and protocol, the NFCC will issue RF commands to put the Remote NFC Endpoint into sleep state. The NFCC internally marks the state of that particular RF Discovery ID device as “sleeping”, and SHALL send RF_DEACTIVATE_RSP followed by RF_DEACTIVATE_NTF (Sleep Mode or Sleep_AF Mode, DH Request) upon successful completion. The state will change to **RFST_W4_HOST_SELECT**.

If the DH sends RF_DEACTIVATE_CMD (Idle Mode), the NFCC SHALL send RF_DEACTIVATE_RSP followed by RF_DEACTIVATE_NTF (Idle Mode, DH Request) upon successful deactivation. The state will then change to **RFST_IDLE**.

If the DH sends RF_DEACTIVATE_CMD (Discovery), the NFCC SHALL send RF_DEACTIVATE_RSP followed by RF_DEACTIVATE_NTF (Discovery, DH Request) upon successful deactivation. The state will then change to **RFST_DISCOVERY** where the NFCC SHALL either restart or continue the Polling discovery activity.

In both cases above, depending on the activated interface and protocol, the NFCC may issue RF commands to deactivate the Remote NFC Endpoint before sending RF_DEACTIVATE_NTF.

When using the ISO-DEP or NFC-DEP RF interface, and the NFCC detects an error during the RF communication, it SHALL notify the DH sending CORE_INTERFACE_ERROR_NTF, using the appropriate status out of RF_TRANSMISSION_ERROR, RF_PROTOCOL_ERROR and RF_TIMEOUT_ERROR (see Table 94). The state will then remain **RFST_POLL_ACTIVE.**

5.2.6 State RFST_LISTEN_ACTIVE

In this state the NFCC is activated in Listen Mode.

In this state an RF Interface is activated, which allows the NFCC to communicate with a Remote NFC Endpoint as specified by the active RF Interface. The NFCC SHALL accept and send Data Messages to or from the DH or NFCEE based on the routing tables.

NOTE By configuring the NFCC to send RF_NFCEE_ACTION_NTF notifications, the DH will get informed if a decision is made by the NFCC based on the routing algorithm to route to a different NFCEE (see 7.4). The local routing destination can change when using AID based routing (see 6.3.1) if the Remote NFC Endpoint selects an AID where the corresponding application is hosted on a different NFCEE than the one previously selected. Routing can also change if the NFC Forum Device appears to the Remote NFC Endpoint as not one but multiple endpoints and the Remote NFC Endpoint switches communication between these endpoints.

In this state the NFCC MAY be put to sleep mode by the DH sending RF_DEACTIVATE_CMD (Sleep Mode or Sleep_AF Mode) (e.g., if ISO-DEP or NFC-DEP implementation on the DH use Frame RF Interface) or by the Remote NFC Endpoint. If the deactivation to sleep is successful, the NFCC SHALL send RF_DEACTIVATE_NTF (Sleep Mode or Sleep_AF Mode, DH Request, or Endpoint Request) to the DH. The state will then change to **RFST_LISTEN_SLEEP**.

If the DH sends RF_DEACTIVATE_CMD (Idle Mode), the NFCC SHALL send RF_DEACTIVATE_RSP followed by RF_DEACTIVATE_NTF (Idle, DH Request) upon successful deactivation. The state will then change to **RFST_IDLE**.

On detection of remote RF field off, the NFCC SHALL move to the NO REMOTE FIELD state in the Listen Mode State Machine defined in [ACTIVITY] and send RF_DEACTIVATE_NTF (Discovery, RF Link Loss) to the DH. The RF Communication state will then change to **RFST_DISCOVERY.**

When using the RF Frame Interface and the DH detects a command or an error during the RF communication which forces returning to the IDLE state, as defined in the [ACTIVITY] Listen Mode state machine, the DH SHALL send RF_DEACTIVATE_CMD (Discovery) to the NFCC, which SHALL answer RF_DEACTIVATE_RSP followed by RF_DEACTIVATE_NTF (Discovery, DH Request). The state will then change to **RFST_DISCOVERY.**

When using the NFC-DEP RF interface and the NFCC detects a RLS_REQ command during the RF communication, which forces it returning to the IDLE state, as defined in the [ACTIVITY] Listen Mode state machine, the DH SHALL send RF_DEACTIVATE_NTF (Discovery, Endpoint_Request) to the DH. The state will then change to **RFST_DISCOVERY**.

When using the ISO-DEP or NFC-DEP RF interface, and the NFCC detects an error during the RF communication, which does not require returning to the IDLE state, as defined in the [ACTIVITY] Listen Mode state machine, the NFCC SHALL send CORE_INTERFACE_ERROR_NTF, using the appropriate status out of RF_TRANSMISSION_ERROR, RF_PROTOCOL_ERROR and RF_TIMEOUT_ERROR (see table 88). The state will then remain **RFST_LISTEN_ACTIVE**.

5.2.7 State RFST_LISTEN_SLEEP

In this state the NFCC is not supposed to respond to any RF commands until it gets a valid RF wake up command (for details refer [DIGITAL]).

If the NFCC receives a valid RF wake up command(s) followed by successful activation procedure, the NFCC SHALL send RF_INTF_ACTIVATED_NTF (Listen mode) to the DH. At that point, the state is changed back to **RFST_LISTEN_ACTIVE**.

On detection of remote RF field off, the NFCC SHALL send RF_DEACTIVATE_NTF (Discovery, RF Link Loss) to the DH. The RF Communication state will then change to **RFST_DISCOVERY.**

For NFC-B Technology, when the NFCC detects a command during the RF communication, which forces returning to the IDLE state, as defined in the [ACTIVITY] Listen Mode state machine, the NFCC SHALL send RF_DEACTIVATE_NTF (Discovery, NFC-B_Bad_AFI) to the DH. The state will then change to **RFST_DISCOVERY**.

If the DH sends RF_DEACTIVATE_CMD (Idle Mode), the NFCC SHALL send RF_DEACTIVATE_RSP. The state will then change to **RFST_IDLE**.

5.3 RF Field Information

This Notification is used to inform the DH about operating fields generated by Remote NFC Endpoints.

Table 20: Notification for RF Field information

RF_FIELD_INFO_NTF		
Payload Field(s)	**Length**	**Value/Description**
RF Field Status	1 Octet	See Table 21.

Table 21: RF Field Status

<table>
<tr><th></th><th colspan="8">Bit Mask</th><th>Description</th></tr>
<tr><td></td><td>b7</td><td>b6</td><td>b5</td><td>b4</td><td>b3</td><td>b2</td><td>b1</td><td>b0</td><td></td></tr>
<tr><td rowspan="2">Octet 0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td></td><td>RFU</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>X</td><td>1b: Operating field generated by Remote NFC Endpoint
0b: No Operating field generated by Remote NFC Endpoint.</td></tr>
</table>

The DH can configure whether the NFCC is allowed to send RF Field Information Notifications by setting the following configuration parameter:

Table 22: RF Field Information Configuration Parameter

ID	Length	Value	Description
RF_FIELD_INFO	1 Octet	0x00 (default)	The NFCC is not allowed to send RF Field Information Notifications to the DH.
		0x01	The NFCC is allowed to send RF Field Information Notifications to the DH.
		0x02-0xFF	RFU

If RF_FIELD_INFO is set to 0x01, the following rules apply:

- The NFCC SHALL send RF_FIELD_INFO_NTF immediately if the RF_FIELD_INFO is set to 0x01. This allows the DH to retrieve the current RF field status.
- If an operating field from a Remote NFC Endpoint has been detected (bit 0 of RF Field Status is set to 1b) and if not specified differently by the used Transport Mapping, the NFCC SHALL send RF_FIELD_INFO_NTF.

  Transport Mappings MAY restrict the sending of RF_FIELD_INFO_NTF if the transport requires time to become operational after detecting an external field and if the DH can use the start of this transport activation as an indication of the presence of an external field. Transport Mappings SHALL NOT restrict the sending of RF_FIELD_INFO_NTF for any other reason.
- If the loss of an operating field from a Remote NFC Endpoint has been detected (bit 0 of RF Field Status is set to 0b), the NFCC SHALL send RF_FIELD_INFO_NTF.

If RF_FIELD_INFO is set to 0x00, the NFCC SHALL NOT send RF_FIELD_INFO_NTF notifications.

6 RF Communication Configuration

Before starting the actual RF Discovery Process by moving to the **RFST_DISCOVERY** state described in Section 5.2, the DH SHALL have first configured:

- Any non-default Poll Mode and Listen Mode parameters
- The mapping between protocols and interfaces
- Any CE routing that is needed

The above steps need to be followed before the first time the RF Discovery Process is started, and after that, only when something changes. They are described in detail in the following sections.

6.1 Configuration Parameters

These are configuration parameters related to discovery. The Commands used to set and get these parameters are specified in Section 4.3.

The sub-sections below describe all configurable RF Discovery parameters.

Table 101 contains a list of all parameters with their Parameter Tags. All parameters have default values, so the DH is not required to configure any RF Discovery parameter.

Also, some parameters are relevant only for either Listen or Poll Mode or only to specific RF Interface(s). These parameters are described in the corresponding RF Interface sections.

If the DH has changed any parameter, the DH MAY reset the parameter back to its default value by sending a CORE_SET_CONFIG_CMD with a Parameter field containing the ID of the parameter, a length of 0x00, and no value field.

If the DH retrieves a Parameter whose current value is a default defined by the NFCC, the NFCC SHALL send the actual value of the parameter, as set by the NFCC)

In the particular case where the NFCC combines configuration parameters from the DH and some NFCEEs (as reported in CORE_INIT_RSP; see Table 9), the NFCC may modify the configuration parameters set by the DH before they are visible on the RF or read back by the DH. The way they are modified is out of the scope of this document.

6.1.1 Poll A Parameters

Table 23: Discovery Configuration Parameters for Poll A

ID	Length	Value	Description
PA_BAIL_OUT	1 Octet	0x00 (default)	No bail out during Poll Mode in Discovery activity, as defined in [ACTIVITY].
		0x01	Bail out when NFC-A Technology has been detected during Poll Mode in Discovery activity, as defined in [ACTIVITY].
		0x02-0xFF	RFU

6.1.2 Poll B Parameters

Table 24: Discovery Configuration Parameters for Poll B

ID	Length	Value	Description
PB_AFI	1 Octet	variable	Application family identifier as defined in [DIGITAL]. Default: 0x00 (all application families) [DIGITAL] mandates a value of 0x00.
PB_BAIL_OUT	1 Octet	0x00 (default)	No bail out during Poll Mode in Discovery activity, as defined in [ACTIVITY].
		0x01	Bail out when NFC-B Technology has been detected during Poll Mode in Discovery activity, as defined in [ACTIVITY].
		0x02-0xFF	RFU
PB_ATTRIB_PARAM1	1 Octet	variable	The values and coding of this parameter SHALL be as defined in [DIGITAL] for Param 1 of the ATTRIB command. There is no default because this is a read only parameter. The DH SHALL NOT attempt to write this parameter
PB_SENSB_REQ_PARAM	1 Octet	variable	Control of what is sent in the PARAM byte of an ALLB_REQ or SENSB_REQ as defined in [DIGITAL]. See Table 25 Default: 0x00

NOTE The information provided by PB_ATTRIB_PARAM1 can be used by the DH to construct a valid ATTRIB command when using the Frame RF Interface.

Table 25: Values for PB_SENSB_REQ_PARAM

	Bit Mask								Description
	b7	**b6**	**b5**	**b4**	**b3**	**b2**	**b1**	**b0**	
Octet 0	0	0							RFU
			X						If set to 1b, the NFCC SHALL indicate support for advanced features in an ALLB_REQ or SENSB_REQ, otherwise it SHALL NOT indicate support
				X					If set to 1b, the NFCC SHALL indicate support for extended SENSB_RES in an ALLB_REQ or SENSB_REQ, otherwise it SHALL NOT indicate support
					0	0	0	0	Will be set by the NFCC independently of what is configured by DH.

6.1.3 Poll F Parameters

Table 26: Discovery Configuration Parameters for Poll F

ID	Length	Value	Description
PF_BIT_RATE	1 Octet	1 – 2	The initial bit rate. Value coding, refer to Table 97 Default: 0x01
PF_RC_CODE	1 Octet	0-2	Value of RC to use in SENSF_REQ, as defined in [DIGITAL]. Default: 0x00

6.1.4 Poll ISO-DEP Parameters

Table 27: Discovery Configuration Parameters for ISO-DEP

ID	Length	Value	Description
PB_H_INFO	0-n Octets		Higher layer INF field of the ATTRIB Command as defined in [DIGITAL]. Default: empty (NFCC will send ATTRIB Command without Higher Layer – INF field).
PI_BIT_RATE	1 Octet	0-3	Maximum allowed bit rate Default: 0x00 (106 Kbit/s) For value coding, refer to Table 97. Depending on the capabilities of NFCC, the NFCC MAY adjust a lower bit rate than specified by this field even if a higher bit rate would be supported by the Remote NFC Endpoint.
PA_ADV_FEAT	1 Octet	0x00 (default)	The NFCC SHALL NOT indicate support for advanced protocol features in the RATS command as defined in [DIGITAL]
		0x01	The NFCC SHALL indicate support for advanced protocol features in the RATS command as defined in [DIGITAL].
		0x02-0xFF	RFU

6.1.5 Poll NFC-DEP Parameters

Those parameters MAY be configured if polling for NFC-A and/or NFC-F or P2P-Active.

Table 28: Discovery Configuration Parameters for Poll NFC-DEP

ID	Length	Value	Description
PN_NFC_DEP_SPEED	1 Octet	0x00 (Default)	Highest Available Bit Rates. The NFCC SHOULD use the highest supported Bit Rates for Data Exchange. The NFCC is responsible for determining the highest Bit Rate available for Data Exchange. If the Bit Rate change is possible, a PSL_REQ is sent by the NFCC.
		0x01	Maintain the Bit Rates. The NFCC SHALL use the same bit rates for Data Exchange as were used for Device Activation, even if higher bit rates are supported. This means that a PSL_REQ is not sent by the NFCC.
		0x02-0xFF	RFU
PN_ATR_REQ_GEN_BYTES	0-n Octets		General Bytes for ATR_REQ. Default: empty (no General Bytes SHALL be sent in ATR_REQ).
PN_ATR_REQ_CONFIG	1 Octet		Configuration to be used in the Optional Parameters ($\mathbf{PP}_I$) within ATR_REQ. See Table 29. Default: 0x30

Table 29: Values for PN_ATR_REQ_CONFIG

	Bit Mask								Description
	b7	**b6**	**b5**	**b4**	**b3**	**b2**	**b1**	**b0**	
Octet 0	0	0			0		0	0	RFU
						X			If set to 1b, the DID MAY be used, otherwise DID SHALL NOT be used NOTE Needs to be 0b for LLCP.
			X	X					Value for **LR**$_{I}$ as defined in [DIGITAL] NOTE Needs to be always set to 11b for LLCP

6.1.6 Listen A Parameters

Table 30: Discovery Configuration Parameters for Listen A

ID	Length	Description
LA_BIT_FRAME_SDD	1 Octet	Bit Frame SDD value to be sent in Byte 1 of SENS_RES. This is a 5-bit value that SHALL be contained in the 5 least significant bits of the octet. Default: NFCC decides (the NFCC SHALL set the value as defined in [DIGITAL]).
LA_PLATFORM_CONFIG	1 Octet	Platform Configuration value to be sent in Byte 2 of SENS_RES. This is a 4-bit value that SHALL be contained in the 4 least significant bits of the octet. Default: NFCC decides (the NFCC SHALL set the value as defined in [DIGITAL]).
LA_SEL_INFO	1 Octet	This value is used to generate SEL_RES according [DIGITAL]. Bits set in this field SHALL be set in the SEL_RES sent by the NFCC. See Table 31. Default: NFCC decides (the NFCC SHALL set the default value corresponding to the RF Interfaces implemented on the NFCC).
LA_NFCID1	4, 7, or 10 Octets	NFCID1 as defined in [DIGITAL]. Default: NFCC decides.

Table 31: LA_SEL_INFO coding

	Bit Mask								Description
	b7	**b6**	**b5**	**b4**	**b3**	**b2**	**b1**	**b0**	
Octet 0	X			X	X		X	X	For proprietary use
						0			Will be set by the NFCC independently of what is configured by DH.
		X							If set to 1b, NFC-DEP Protocol is supported by the NFC Forum Device in Listen Mode.
			X						If set to 1b, ISO-DEP Protocol is supported by the NFC Forum Device in Listen Mode.

6.1.7 Listen B Parameters

Table 32: Discovery Configuration Parameters for Listen B

ID	Length	Description
LB_SENSB_INFO	1 Octet	Used to generate Byte 2 of Protocol Info within SENSB_RES according to [DIGITAL]. See Table 33. Default: NFCC decides. The NFCC SHALL set the default value corresponding to the protocols implemented on the NFCC.
LB_NFCID0	4 Octets	NFCID0 as defined in [DIGITAL]. Default: NFCC decides.
LB_APPLICATION_DATA	4 Octets	Application Data (Bytes 6-9) of SENSB_RES as defined in [DIGITAL]. Default: All octets are set to 0x00.
LB_SFGI	1 Octet	Start-Up Frame Guard Time as defined in [DIGITAL]. Default: NFCC decides.
LB_ADC_FO	1 Octet	Byte 3 of Protocol Info within SENSB_RES according [DIGITAL]. See Table 34. Default: 0x05

Table 33: LB_SENSB_INFO values

	Bit Mask								Description
	b7	**b6**	**b5**	**b4**	**b3**	**b2**	**b1**	**b0**	
Octet 0	X	X							For proprietary use
			0	0	0	0	0		RFU
								X	If set to 1b, ISO-DEP Protocol is supported by the NFC Forum Device in Listen Mode.

Table 34: LB_ADC_FO values

	Bit Mask								Description
	b7	**b6**	**b5**	**b4**	**b3**	**b2**	**b1**	**b0**	
Octet 0	0	0	0	0			0		Will be set by the NFCC independently of what is configured by DH
					X	X			b3 and b4 of ADC Coding field of SENSB_RES (Byte 12) as defined in [DIGITAL]
								X	If set to 1b DID MAY be used, otherwise it SHALL NOT be used.

6.1.8 Listen F Parameters

The NFC-F listen configuration supports the configuration for the NFC Forum Type 3 Tag Platform, as well as for NFC-DEP Target functionality. The DH may configure none, one, or both of these capabilities.

The names of configuration parameters in Table 35, which are only relevant for the Type 3 Tag Platform, start with the prefix ‘LF_T3T_’. The other parameters are relevant to configure the NFC-DEP functionality.

Using the configuration for the NFC-DEP functionality, the DH can provide the information required by the Listen Mode state machine described in [ACTIVITY] for moving from the IDLE state to the READY_F Sub-state. The NFC-DEP configuration SHALL be used by the NFCC for this purpose.

Using the configuration for the Type 3 Tag Platform, the DH can provide the information required by the Listen Mode state machine described in [ACTIVITY] for answering a SENSF_REQ Command in IDLE state or READY_F Sub-state . The Type 3 Tag Platform configuration SHALL be used by the NFCC for this purpose.

The Type 3 Tag Platform configuration uses a set of parameters whose names start with LF_T3T_IDENTIFIERS_and a following number. This document refers to the trailing number as the index and uses the term LF_T3T_IDENTIFIERS without index to refer to the whole parameter set.

Table 35: Discovery Configuration Parameters for Listen F

ID	Length	Value	Description
LF_CON_BITR_F	1 Octet	Configures the bit rates to listen for (as defined in [ACTIVITY]). See Table 37 Default: 110b (listen for both speeds)	
LF_PROTOCOL_TYPE	1 Octet	Protocols supported by the NFC Forum Device in Listen Mode for NFC-F. See Table 36. Default: NFCC decides. The NFCC SHALL set the default value corresponding to the protocols implemented on the NFCC.	
LF_T3T_MAX	1 Octet	0 – 16	The maximum index of LF_T3T_IDENTIFIERS supported by the NFCC. There is no default because this is a read only parameter. The DH SHALL NOT attempt to write this parameter.
		17 – 255	RFU
LF_T3T_IDENTIFIERS_1	10 Octets	For each identifier: Octet 0 and Octet 1 indicate the System Code of a Type 3 Tag Emulation occurring on the DH. Octet 2 – Octet 9 indicates NFCID2 for the Type 3 Tag Platform. Default: Octet 0 and 1 SHALL be set to 0xFF. Octet 2 SHALL be set to 0x02. Octet 3 SHALL be set to 0xFE. Octets 4-9 SHALL be set to 0x00.	
LF_T3T_IDENTIFIERS_2	10 Octets		
…			
LF_T3T_IDENTIFIERS_16	10 Octets		
LF_T3T_PMM	8 Octets	PAD0, PAD1, MRTI_check, MRTI_update and PAD2 of SENSF_RES as defined in [DIGITAL]. Default: all octets set to 0xFF	
LF_T3T_FLAGS	2 Octets	A bit field indicating which LF_T3TIDENTIFIERS are enabled in the process to create responses to a SENSF_REQ. Detailed definitions as defined below. Default: 0x0000.	
LF_ADV_FEAT	1 Octet	0x00 (default)	The NFCC SHALL NOT include RD bytes in its SENSF_RES if it receives a SENSF_REQ with RC set to 0x02
		0x01	The NFCC MAY include RD bytes in its SENSF_RES if it receives a SENSF_REQ with RC set to 0x02
		0x02-0xFF	RFU

If the DH is interested in NFC-DEP based communication it SHALL set the b1 of LF_PROTOCOL_TYPE to 1, which will enables the generation of SENSF_RES indicating NFC-DEP capabilities as a response to a SENSF_REQ having a System Code of FFFFh. Otherwise the DH SHALL set the b1 of this field to 0.

The bit at position X of LF_T3T_FLAGS determines whether the NFCC is allowed to send a SENSF_RES with the identifiers specified in LF_T3T_IDENTIFIERS with index X. If the bit at position X is set to 0, the contents of LF_T3T_IDENTIFIERS with index X SHALL be ignored in the process to create responses to a SENSF_REQ, if set to 1 they are enabled to be used in this process. The bits having a higher number than the value of LF_T3T_MAX SHALL be ignored by the NFCC.

If LF_T3T_FLAGS is set to the default 0x0000, sending of SENSF_RES for Type 3 Tag platform is disabled.

Table 36: Supported Protocols for Listen F

	Bit Mask								**Description**
	b7	**b6**	**b5**	**b4**	**b3**	**b2**	**b1**	**b0**	
Octet 0	X	X							For proprietary use
			0	0	0	0		0	RFU
							X		If set to 1b, NFC-DEP Protocol is supported by the NFC Forum Device in Listen Mode

Table 37: LF_CON_BITR_F Values

	Bit Mask								**Description**
	b7	**b6**	**b5**	**b4**	**b3**	**b2**	**b1**	**b0**	
Octet 0	0	0	0	0	0			0	RFU
						X			If set to 1b, listen for 424 kbps
							X		If set to 1b, listen for 212 kbps
	At least one of the bits b1 or b2 SHALL be set to 1b.								

The Octets 0 and 1 of an LF_T3T_IDENTIFIERS value (representing a Type 3 Tag Platform System Code) SHALL NOT be configured to be equal to Octets 0 and 1 of any other LF_T3T_IDENTIFIERS value.

The NFCC SHALL answer by sending CORE_SET_CONFIG_RSP with a Status of STATUS_REJECTED, if the CORE_SET_CONFIG_CMD contains either :

- the parameter LF_T3T_MAX.
- a parameter of type LF_T3T_IDENTIFIERS with a higher index than the value of LF_T3T_MAX.

- a LF_T3T_IDENTIFIERS parameter with values for Octets 0 and 1 which are equal to Octet 0 and 1 of another LF_T3T_IDENTIFIERS parameters value.

The NFCC SHALL answer by sending CORE_GET_CONFIG_RSP with a Status of STATUS_REJECTED, if the CORE_GET_CONFIG_CMD contains

- a parameter of type LF_T3T_IDENTIFIERS with a higher index than the value provided by LF_T3T_MAX.

NOTE If the value of LF_T3T_MAX is set to 0, the NFCC does not support generating SENSF responses for the Type 3 Tag Platform.

6.1.9 Listen ISO-DEP Parameters

These parameters MAY be configured if listening for NFC-A or NFC-B.

Table 38: Discovery Configuration Parameters for Listen ISO-DEP

ID	Length	Description
LI_FWI	1 Octet	Frame Waiting time Integer as defined in [DIGITAL]. Default: 4.
LA_HIST_BY	0 – n Octets	Historical Bytes (only applicable for Type 4A Tag) as defined in [DIGITAL]. Default: empty (do not send historical bytes).
LB_H_INFO_RESP	0 – n Octets	Higher Layer – Response field of the ATTRIB response as defined in [DIGITAL]. Default: empty (send ATTRIB response without Higher Layer – Response field).
LI_BIT_RATE	1 Octet	Maximum supported bit rate. Default: 0x00 (106 Kbit/s) For value coding, refer to Table 97. Depending on the capabilities of NFCC, the NFCC MAY reduce the maximum supported bit rate reported to the RF reader.

6.1.10 Listen NFC-DEP Parameters

These parameters MAY be configured if listening for NFC-A and/or NFC-F.

Table 39: Discovery Configuration Parameters for Listen NFC-DEP

ID	Length	Value	Description
LN_WT	1 Octet	Waiting Time defined in [DIGITAL]. Default: Target value for WT_{MAX} as defined in [DIGITAL].	
LN_ATR_RES_GEN_BYTES	0-n Octets	General Bytes in ATR_RES as defined in [DIGITAL]. Default: empty (no General Bytes SHALL be sent in ATR_RES).	
LN_ATR_RES_CONFIG	1 Octet	Used to generate the Optional parameters (**PP**$_T$) in ATR_RES. See Table 40. Default: 0x30 (NFCC indicates a maximum payload size of 254 bytes).	

Table 40: Values for LN_ATR_RES_CONFIG

	Bit Mask								Description
	b7	**b6**	**b5**	**b4**	**b3**	**b2**	**b1**	**b0**	
Octet 0	0	0			0	0			RFU
							X	X	Will be set by the NFCC independently of what is configured by DH
			X	X					Value for **LR**$_T$ as defined in [DIGITAL]. Default: 11b NOTE Need to be set to 11b for LLCP

6.1.11 Common Parameters

Table 41: Common Parameters for Discovery Configuration

ID	Length	Description
TOTAL_DURATION	2 Octets	0x0000 – 0xFFFF defines the Total Duration of the single discovery period in [ms]. This time definition provides only a rough target value for the NFCC, but NFCC may adjust the duration time due to the current limitation of active RF Protocols and hardware limitations. Default: NFCC decides.
CON_DEVICES_LIMIT	1 Octet	As defined in [ACTIVITY] for the Collision Resolution Activity Default: NFCC decides (based on its capabilities).

6.2 RF Interface Mapping Configuration

These Control Messages are used to configure the mapping between RF Protocols and RF Interfaces.

Table 42: Control Messages for RF Interface Mapping Configuration

RF_DISCOVER_MAP_CMD				
Payload Field(s)	**Length**	**Value/Description**		
Number of Mapping Configurations	1 Octet	The number of Mapping Configuration fields to follow (n).		
Mapping Configuration [1..n]	3 Octets	RF Protocol	1 Octet	See Table 98.
		Mode	1 Octet	See Table 43.
		RF Interface	1 Octet	See Table 99. The value 0 (NFCEE Direct RF Interface) SHALL NOT be used.

RF_DISCOVER_MAP_RSP		
Payload Field(s)	**Length**	**Value/Description**
Status	1 Octet	See Table 94.

Table 43: Value Field for Mode

	Bit Mask								**Description**
	b7	**b6**	**b5**	**b4**	**b3**	**b2**	**b1**	**b0**	
Octet 0	0	0	0	0	0	0			RFU
							X		If set to 1b, the RF Interface is mapped to the RF Protocol in Listen Mode.
								X	If set to 1b, the RF Interface is mapped to the RF Protocol in Poll Mode.
	At least one of the bits b0 or b1 SHALL be set to 1b.								

A Mapping Configuration defines the RF Interface which SHALL be used for the communication from the DH to a Remote NFC Endpoint using the specified RF Protocol and Mode, when the NFCC autonomously transfers to the **RFST_POLL_ACTIVE** or **RFST_LISTEN_ACTIVE** states.

If the DH initiates the transfer to the **RFST_POLL_ACTIVE** state by sending an RF_DISCOVER_SELECT_CMD, the DH SHALL specify the interface to use as a parameter to the RF_DISCOVER_SELECT_CMD which overruless any values which may be in the RF Interface Mapping configuration.

Only one RF Interface SHALL be mapped to each RF Protocol.

To perform RF Interface mapping, the DH SHALL send a RF_DISCOVER_MAP_CMD to the NFCC.

The NFCC SHALL set the RF Interface for all RF Protocols / Modes not included in the RF_DISCOVER_MAP_CMD to the value Frame RF Interface.

If the NFCC accepts the RF Interface mapping configuration, it SHALL respond using a RF_DISCOVER_MAP_RSP with a Status of STATUS_OK.

If the RF Interface mapping is invalid, the NFCC SHALL reject the mapping configuration by sending a RF_DISCOVER_MAP_RSP with a Status of STATUS_REJECTED. In this case the DH MAY attempt an alternative configuration.

If the mapping configuration is rejected, the state of the RF Interface Mapping configuration in the NFCC SHALL be invalid. As long as the table is invalid, starting of the RF Discovery Process is not possible (see Section 6).

The DH SHALL only use RF Interface values that are supported by the NFCC. The RF Interfaces supported by the NFCC are made known to the DH in the CORE_INIT_RSP.

6.3 Listen Mode Routing Configuration

If, as part of the Discovery Process, the DH wants the NFCC to enter Listen Mode and the NFCC has indicated support for listen mode routing, the DH SHALL configure the Listen Mode Routing Table. This is required to provide the NFCC the information on where to route received data when activated in Listen Mode.

If an NFCEE for which a routing entry exists becomes disabled, the NFCC SHALL stop the routing to that NFCEE until the NFCEE is enabled again. The corresponding routing entries SHALL be ignored for the time the NFCEE is disabled.

NOTE The DH should evaluate whether to reconfigure the Listen Mode Routing Table if an NFCEE is disabled for which protocol or technology based routing entries exist that can be handled by other NFCEEs.

If an NFCEE is removed, then the corresponding NFCEE ID becomes invalid and the DH SHALL remove all Routing Entries for that NFCEE ID by reconfiguring the Listen Mode Routing Table.

6.3.1 Listen Mode Routing Table Design

The Listen Mode Routing Table consists of three types of routing entries. These are:

- AID-based routing entries
- Protocol-based routing entries
- Technology-based routing entries

AID-based routing is only possible if the NFCC terminates the ISO-DEP protocol and understands at least the SELECT command defined in [ISO/IEC_7816-4]. The DH knows if the NFCC supports this from the NFCC capabilities returned in the CORE_INIT_RSP.

Each AID SHALL be unique in the NFCC's routing table.

For each protocol value except for the T3T protocol and proprietary protocols, there SHALL be at most one protocol-based route be configured for each supported Power State (Switched On, Switched Off and Battery Off).

For T3T protocol, as the Remote NFC Endpoint can manage responses from multiple NFCEEs, the DH MAY configure more than one NFCEE ID for each supported Power State. In this case, data received from the Remote NFC Endpoint will be forwarded to all configured NFCEE IDs if the NFCC cannot find the route for it. The corresponding rules for routing entries for proprietary protocols are out-of-scope of this specification.

NOTE Transitions between Power States are implementation specific issues, thus out-of-scope of this specification.

For each technology there SHALL be at most one technology-based route configured for each supported Power State.

The order of searching through the routing table SHALL be (1) AID-based, (2) Protocol-based and (3) Technology-based. For each received frame from RF, the NFCC SHALL follow the steps below until a route is found:

1. If the Listen-side ISO-DEP RF Interface is used and there is at least one AID-based routing entry for the current power state, then:

   a. Monitor for the SELECT command as defined in [ISO/IEC_7816-4]. If a SELECT command is found for type AID, match the AID in the routing table, based on the current power state. If matched, route to that target (including the SELECT command), otherwise continue with step 1b.

   b. If it is the first APDU after ISO-DEP activation, continue with step 2. Otherwise, route to the same target as the previous command.

2. Check if there is a target configured in the routing table for the currently active protocol, based on the current power state. If yes, route to the target. If T3T, check for and route to multiple targets, otherwise continue with step 3.
3. Check if there is target configured for the currently active NFC Technology, based on the current power state. If yes, route to that target, otherwise continue with step 4.
4. If this step is reached, no route was found for the received frame. The NFCC handles the frame in an implementation-specific manner.

6.3.2 Configure Listen Mode Routing

These Control Messages are used to configure the Listen Mode Routing Table.

Table 44: Control Messages to Configure Listen Mode Routing

<table>
<tr><th colspan="5">RF_SET_LISTEN_MODE_ROUTING_CMD</th></tr>
<tr><th>Payload Field(s)</th><th>Length</th><th colspan="3">Value/Description</th></tr>
<tr><td>More</td><td>1 Octet</td><td colspan="3">See Table 45.</td></tr>
<tr><td>Number of Routing Entries</td><td>1 Octet</td><td colspan="3">The number of Routing Entry fields to follow (n).
This Control Message SHALL be at least one Routing Entry.</td></tr>
<tr><td rowspan="3">Routing Entry [1..n]</td><td rowspan="3">x+2 Octets</td><td>Type</td><td>1 Octet</td><td>One of the types defined in Table 46.</td></tr>
<tr><td>Length</td><td>1 Octet</td><td>The length of Value (x).</td></tr>
<tr><td>Value</td><td>x Octets</td><td>Value of the Routing TLV.</td></tr>
</table>

<table>
<tr><th colspan="3">RF_SET_LISTEN_MODE_ROUTING_RSP</th></tr>
<tr><th>Payload Field(s)</th><th>Length</th><th>Value/Description</th></tr>
<tr><td>Status</td><td>1 Octet</td><td>See Table 94.</td></tr>
</table>

Table 45: More field values

Value	Description
0x00	Last Message
0x01	More Message(s) to follow
0x02 – 0xFF	RFU

Table 46: TLV Coding for Listen Mode Routing

Type	Length	Value
0x00	3 Octets	Technology-based routing entry, value field coded according to Table 47.
0x01	3 Octets	Protocol-based routing entry, value field coded according to Table 48.
0x02	2+n Octets	AID-based routing entry, value field coded according to Table 49.
0x03-0x9F		RFU
0xA0-0xFF		For proprietary use

Table 47: Value Field for Technology-based Routing

Payload Field(s)	Length	Value/Description
Route	1 Octet	An NFCEE ID as defined in Table 84
Power State	1 Octet	See Table 50.
Technology	1 Octet	A valid RF Technology as defined in Table 95.

Table 48: Value Field for Protocol-based Routing

Payload Field(s)	Length	Value/Description
Route	1 Octet	An NFCEE ID as defined in Table 84
Power State	1 Octet	See Table 50.
Protocol	1 Octet	A valid RF Protocol as defined in Table 98

Table 49: Value Field for AID-based Routing

Payload Field(s)	Length	Value/Description
Route	1 Octet	An NFCEE ID as defined in Table 84
Power State	1 Octet	See Table 50.
AID	5-16 Octets	Application Identifier

Table 50: Value Field for Power State

	Bit Mask								Description
	b7	b6	b5	b4	b3	b2	b1	b0	
	0	0	0	0	0				RFU
						X			Battery off
							X		Switched off
								X	Switched on

When configuration of the NFCC for Listen Mode communication is necessary, the DH SHALL always send the complete Listen Mode Routing Table.

The NFCC uses the More field to determine if it has received all the Commands necessary to configure the routing. The new routing SHALL only become effective following receipt of all configuration information.

The DH SHALL keep the total size of the routing configuration information smaller than the 'Max Routing Table Size' indicated during Initialization (see Section 4.2).

All parameters except 'More' and 'Number of Routing Entries' are included in the calculation to determine if the routing configuration size exceeds the Max Routing Table Size.

The DH SHALL NOT try to configure routing of a specific type unless the NFCC has indicated support for routing that type in the NFCC Features sent in the CORE_INIT_RSP. Also, the DH SHALL NOT try to configure routing for a specific power state unless the NFCC has indicated support for that power state in the NFCC Features sent in the CORE_INIT_RSP.

On receipt of the RF_SET_LISTEN_MODE_ROUTING_CMD with a valid routing configuration, the NFCC SHALL respond with the RF_SET_LISTEN_MODE_ROUTING_RSP with a Status of STATUS_OK.

In case of an error the NFCC SHALL respond with the RF_SET_LISTEN_MODE_ROUTING_RSP with a Status of STATUS_FAILED and the routing table SHALL be emptied.

Also if a new routing configuration is comprised of several Commands, and any one of these Commands fail, then the new routing configurations SHALL be ignored and the routing table SHALL be emptied

After above failure cases, the DH SHALL retry to configure the routing table until the NFCC accepts the routing table.

NOTE Failure in routing table configuration may lead to an empty routing table. As the routing table can only be configured in **RFST_IDLE** state this can not happen during ongoing RF communication.

6.3.3 Read Listen Mode Routing

These Control Messages are used by the DH to read the NFCC’s Listen Mode Routing Table.

Table 51: Control Messages to Read the NFCC’s Listen Mode Routing

RF_GET_LISTEN_MODE_ROUTING_CMD		
Payload Field(s)	**Length**	**Value/Description**
Empty Payload		

RF_GET_LISTEN_MODE_ROUTING_RSP		
Payload Field(s)	**Length**	**Value/Description**
Status	1 Octet	See Table 94.

<table>
<tr><th colspan="5">RF_GET_LISTEN_MODE_ROUTING_NTF</th></tr>
<tr><th>Payload Field(s)</th><th>Length</th><th colspan="3">Value/Description</th></tr>
<tr><td>More</td><td>1 Octet</td><td colspan="3">See Table 45.</td></tr>
<tr><td>Number of Routing Entries</td><td>1 Octet</td><td colspan="3">The number of Routing Entry fields to follow (n).
If the Listen Mode Routing Table is empty, the value of this field is 0x00 and there are no Routing Entries following.</td></tr>
<tr><td rowspan="3">Routing Entry [0..n]</td><td rowspan="3">x+2 Octets</td><td>Type</td><td>1 Octet</td><td>One of the types defined in Table 46.</td></tr>
<tr><td>Length</td><td>1 Octet</td><td>The length of Value (x).</td></tr>
<tr><td>Value</td><td>x Octets</td><td>Value of the Routing TLV.</td></tr>
</table>

To retrieve the current routing information from the NFCC, the DH sends the RF_GET_LISTEN_MODE_ROUTING_CMD to the NFCC.

The NFCC SHOULD respond with the RF_GET_LISTEN_MODE_ROUTING_RSP with a Status of STATUS_OK followed by one or more RF_GET_LISTEN_MODE_ROUTING_NTF(s) containing the current routing information.

All but the last RF_GET_LISTEN_MODE_ROUTING_NTF SHALL have the More Parameter set to 1. The last RF_GET_LISTEN_MODE_ROUTING_NTF SHALL have the More Parameter set to 0.

Routing Entry fields will only be present if the value of Number of Routing Entries is greater than zero.

In case of an error the NFCC SHALL respond with RF_GET_LISTEN_MODE_ROUTING_RSP with a Status indicating the failure reason and the RF_GET_LISTEN_MODE_ROUTING_NTF SHALL NOT be sent.

7 RF Discovery

This section describes the Control Messages required for moving through the state machine defined in Section 5.2.

7.1 Starting RF Discovery

RF Discovery Process is a periodic activity consisting of poll and listen cycles, configured by different discovery configurations. The TOTAL_DURATION configuration parameter in Table 41 specifies the duration of one discovery period (=Total Duration). This parameter is used to define the length of listen or idle cycles.

The following rules are applied for the Total Duration parameter:

- If both Poll and Listen Modes are configured, then Listen Duration length is Total Duration minus Poll Duration, where Poll Duration is the time required to execute the Technology Detection Activity (see [ACTIVITY]) for the configured technologies when no Remote NFC Endpoint is detected.
- If only Poll Mode is configured, then Idle Duration length is Total Duration minus Poll Duration.
- If only Listen Mode is configured, then this parameter is not applicable.
- The NFCC MAY extend Total Duration if it is not long enough to cover the Configurations provided in the RF_DISCOVER_CMD.

NOTE [ACTIVITY] defines some limits to the length of polling for each technology, but there may be NFCC implementation specific variations. Also the minimum length of listening for each technology may vary per implementation. Thus the DH may not know the exact minimum value to be used for Total Duration.

If the Discovery Configuration Mode in Octet 0 of the NFCC Features as returned in CORE_INIT_RSP is set to 00b, the DH is responsible for defining not just the configuration parameters, but the Listen Mode Routing Table, and the RF Technology and Mode list in the RF_DISCOVER_CMD. The NFCC is unable to modify these items so the entire process of RF Discovery is as defined by the DH.

If the Discovery Configuration Mode in Octet 0 of the NFCC Features as returned in CORE_INIT_RSP is set to 01b, the DH may choose to set configuration parameters, Listen Mode Routing Table, and the RF Technology and Mode list in the RF_DISCOVER_CMD for its own purposes. However, the NFCC may modify any or all of these items to suit its own RF Discovery needs. If the DH has no needs for RF Discovery, the RF Technology and Mode list in RF_DISCOVER_CMD may be empty, and the NFCC may use just its own values.

The following Control Messages are used to initiate the RF Discovery Process.

Table 52: Control Messages to Start Discovery

<table>
<tr><th colspan="6">RF_DISCOVER_CMD</th></tr>
<tr><th>Payload Field(s)</th><th>Length</th><th colspan="4">Value/Description</th></tr>
<tr><td>Number of Configurations</td><td>1 Octet</td><td colspan="4">The number of Configuration fields to follow (n).
n can be 0 if the RF Discovery is enabled for NFCEE(s) only (e.g. in response to a RF_NFCEE_DISCOVERY_REQ_NTF and an NFCEE(s) providing configuration settings directly to the NFCC). If n is 0, this Command contains no Configuration fields.</td></tr>
<tr><td rowspan="5">Configuration [0..n]</td><td rowspan="5">2 Octets</td><td>RF Technology and Mode</td><td>1 Octet</td><td colspan="2">RF Technology and Mode of the local device. See Table 96.</td></tr>
<tr><td rowspan="4">Discovery Frequency</td><td rowspan="4">1 Octet</td><td>0x00</td><td>RFU</td></tr>
<tr><td>0x01</td><td>RF Technology and Mode will be executed in every discovery period.</td></tr>
<tr><td>0x02-0x0A</td><td>These values are allowed for Poll Mode RF Technology and Mode values. This value specifies how often the Poll period of the specific RF Technology will be executed. For example, a value of 10 indicates that this Polling will be executed in every 10^{th} discovery period.</td></tr>
<tr><td>0x0B-0xFF</td><td>RFU</td></tr>
</table>

RF_DISCOVER_RSP		
Payload Field(s)	**Length**	**Value/Description**
Status	1 Octet	See Table 94.

<table>
<tr><th colspan="4">RF_DISCOVER_NTF</th></tr>
<tr><th>Payload Field(s)</th><th>Length</th><th colspan="2">Value/Description</th></tr>
<tr><td>RF Discovery ID</td><td>1 Octet</td><td colspan="2">See Table 53.</td></tr>
<tr><td>RF Protocol</td><td>1 Octet</td><td colspan="2">RF Protocol supported by the Remote NFC Endpoint. See Table 98.</td></tr>
<tr><td>RF Technology and Mode</td><td>1 Octet</td><td colspan="2">RF Technology and Mode of the local device. See Table 96.</td></tr>
<tr><td>Length of RF Technology Specific Parameters</td><td>1 Octet</td><td colspan="2">The length of RF Technology Specific Parameters field to follow.</td></tr>
<tr><td>RF Technology Specific Parameters</td><td>0 – n Octets</td><td colspan="2">One of the below tables depending on the value of the RF Technology and Mode;
See Table 54 for NFC-A Poll Mode.
See Table 56 for NFC-B Poll Mode.
See Table 58 for NFC-F Poll Mode.
Proprietary parameters if the value of RF Technology and Mode is reserved for a proprietary technology.</td></tr>
<tr><td rowspan="4">Notification Type</td><td rowspan="4">1 Octet</td><td>0</td><td>Last Notification</td></tr>
<tr><td>1</td><td>Last Notification (due to NFCC reaching it’s resource limit)</td></tr>
<tr><td>2</td><td>More Notification to follow</td></tr>
<tr><td>3-255</td><td>RFU</td></tr>
</table>

Table 53: RF Discovery ID

Value	Description
0	RFU
1 – 254	Dynamically assigned by the NFCC
255	RFU

The DH requests that the NFCC starts Discovery activity by sending the RF_DISCOVER_CMD. The parameters RF Technology and Mode and Discovery Frequency are provided by the DH to configure the manner in which the NFCC performs the RF Discovery Process. If the parameters are acceptable for the NFCC, the NFCC SHALL return the RF_DISCOVER_RSP with a Status of STATUS_OK and will start the RF Discovery Process accordingly.

In the case where the RF Communication State Machine is not in the state **RFST_IDLE**, the NFCC SHALL return RF_DISCOVER_RSP with a Status of DISCOVERY_ALREADY_STARTED. In this error case, the current ongoing RF Discovery Process SHALL continue without any changes.

In Poll Mode, if there are multiple Remote NFC Endpoints detected, or if a Remote NFC Endpoint supports multiple RF Protocols, the NFCC SHALL send RF_DISCOVER_NTF to the DH for each combination of Remote NFC Endpoint and RF Protocol detected during the RF Discovery Process.

- The NFCC SHALL assign a unique RF Discovery ID to each detected Remote NFC Endpoint. The combination of RF Discovery ID and RF Protocol SHALL be unique across all RF_DISCOVER_NTFs sent within a series of RF Discovery Notifications. The NFCC SHALL assign the same RF Discovery ID in all notifications send for a single Remote NFC Endpoint supporting multiple protocols.

NOTE If a Remote NFC Endpoint supports multiple protocols, the NFCC uses the same RF Discovery ID for each Notification, but different RF Protocol values. If a Remote NFC Endpoint uses separate polling responses to indicate support of multiple protocols, the NFCC can not know that the responses came from a single Remote NFC Endpoint. In that case, the NFCC assigns different RF Discovery ID.

- All assigned RF Discovery IDs are released when the RF State machine defined in Section 5.2 moves into state **RFST_IDLE**.
- The Notification Type field SHALL be set to 0 or 1 if the current RF_DISCOVER_NTF is the last notification being sent or set to 2 if there is another RF_DISCOVER_NTF to follow (called series of RF Discovery Notifications).

NOTE Since RF_DISCOVER_NTF is only sent when there are multiple Remote NFC Endpoints in the field or if a Remote NFC Endpoint supports multiple RF Protocols, the first notification sent after starting the RF Discovery Process will always have the Notification Type set to 0x02 (more notifications to follow)

- A value of 0x00 SHALL be used for the Notification Type if the NFCC has completed the collision resolution process and no further Remote NFC Endpoints have been identified. A value of 1 SHALL be used if the NFCC has aborted the collision resolution process due to internal restrictions and therefore further Remote NFC Endpoints might not have been detected.

After receiving RF_DISCOVER_NTF with a Notification Type field set to 0x00 or 0x01, the DH SHALL either select the Remote NFC Endpoint by sending RF_DISCOVER_SELECT_CMD or stop the RF Discovery Process by sending RF_DEACTIVATE_CMD.

Table 54: Specific Parameters for NFC-A Poll Mode

Parameter	Length	Description
SENS_RES Response	2 Octets	Defined in [DIGITAL].
NFCID1 Length	1 Octet	Length of NFCID1 Parameter. If an NFC-Forum Type 1 Tag is detected then no NFCID1 Parameter is available and the value of this parameter is set to 0x00. In all other cases NFCID1 Length value SHALL be either 0x04, 0x07 or 0x0A. Other values are RFU.
NFCID1	0, 4, 7, or 10 Octets	Defined in [DIGITAL].
SEL_RES Response Length	1 Octets	Length of SEL_RES Response Parameter. If a NFC-Forum Type 1 Tag is detected then no SEL_RES Response is available and the value of this parameter is set to 0x00. In all other cases the value of the SEL_RES Response Length SHALL be 0x01. Other values are RFU.
SEL_RES Response	0, 1 Octet	Defined in [DIGITAL].

Table 55: Specific Parameters for NFC-A Listen Mode

Parameter	Length	Description
No parameters are currently defined.		

Table 56: Specific Parameters for NFC-B Poll Mode

Parameter	Length	Description
SENSB_RES Response Length	1 Octet	Length of SENSB_RES Response Parameter. Allowed values SHALL be 0x0B and 0x0C. Other values are RFU.
SENSB_RES Response	11 or 12 Octets	Byte 2 – Byte 12 or 13 of SENSB_RES as defined in [DIGITAL].

Table 57: Specific Parameters for NFC-B Listen Mode

Parameter	Length	Description
No parameters are currently defined.		

Table 58: Specific Parameters for NFC-F Poll Mode

Parameter	Length	Description	
Bit Rate	1 Octet	1	212 kbps
		2	424 kbps
		0 and 3 to 255	RFU
SENSF_RES Response Length	1 Octet	Length of SENSF_RES Response Parameter Allowed values SHALL be 0x10 and 0x12. Other values are RFU.	
SENSF_RES Response	16 or 18 Octets	Byte 2 – Byte 17 or 19 of SENSF_RES as defined in [DIGITAL].	

Table 59: Specific Parameters for NFC-F Listen Mode

Parameter	Length	Description
Local NFCID2 Length	1 Octet	Length of Local NFCID2 Parameter. If NFCID2 is available, then Local NFCID2 Length SHALL be 0x08. If NFCID2 is not available, then Local NFCID2 Length SHALL be 0x00, and Local NFCID2 field is not present. Other values are RFU.
Local NFCID2	0 or 8 Octets	NFCID2 generated by the Local NFCC for NFC-DEP Protocol Available for Frame RF Interface

7.2 Select Discovered Target

These Control Messages are used to select an RF Discovery ID and RF Protocol (identifying a Remote NFC Endpoint as reported by a previous RF_DISCOVER_NTF) and the RF Interface to use for communicating with the Remote NFC Endpoint, which might differ from the one defined in the RF_DISCOVER_MAP_CMD.

Table 60: Control Messages to select a Discovered Target

RF_DISCOVER_SELECT_CMD		
Payload Field(s)	**Length**	**Value/Description**
RF Discovery ID	1 Octet	See Table 53.
RF Protocol	1 Octet	See Table 98.
RF Interface	1 Octet	See Table 99. The value 0x00 (NFCEE Direct RF Interface) SHALL NOT be used.

RF_DISCOVER_SELECT_RSP		
Payload Field(s)	**Length**	**Value/Description**
Status	1 Octet	See Table 94.

The DH SHALL send RF_DISCOVER_SELECT_CMD to the NFCC to inform the NFCC which RF Discovery ID, RF Protocol and RF Interface are to be used for subsequent communication.

In case the RF Discovery ID, RF Protocol or RF Interface is not valid the NFCC SHALL respond with RF_DISCOVER_SELECT_RSP with a Status of STATUS_REJECTED.

Otherwise, the NFCC SHALL respond with RF_DISCOVER_SELECT_RSP with a Status of STATUS_OK. After that, the NFCC SHALL perform activation for the RF Protocol, depending on the RF Technology or RF Interface associated with the parameters of the RF_DISCOVER_SELECT_CMD.

The specified RF Interface parameter value is only valid for the following RF Interface activation and does not cause any change to the RF Interface Mapping configuration (see Section 6.2).

7.3 RF Interface Activation and Deactivation

The NFCC can activate an RF Interface either in **RFST_DISCOVERY** or **RFST_W4_HOST_SELECT** state by sending RF_INTF_ACTIVATED_NTF. The RF_INTF_ACTIVATED_NTF SHALL cause the RF communication state to change to either **RFST_POLL_ACTIVE** or **RFST_LISTEN_ACTIVE**, as described in Section 5.2.

At most one RF Interface SHALL be active at any time.

7.3.1 RF Interface Activation Notification

This Notification is used by the NFCC to inform the DH that a specific RF Interface has been activated.

Table 61: Notification for RF Interface activation

RF_INTF_ACTIVATED_NTF		
Payload Field(s)	**Length**	**Value/Description**
RF Discovery ID	1 Octet	See Table 53.
RF Interface	1 Octet	See Table 99. If this contains a value of 0x00 (NFCEE Direct RF Interface) then all following parameters SHALL contain a value of 0 and SHALL be ignored.
RF Protocol	1 Octet	See Table 98.
Activation RF Technology and Mode	1 Octet	RF Technology and Mode of the local device that were used for the collection of the RF Technology Specific Parameters below. See Table 96.
Max Data Packet Payload Size	1 Octet	Max Data Packet Payload Size the NFCC can receive for the Static RF Connection. A number from 1 – 255.
Initial Number of Credits	1 Octet	Initial Number of Credits given by the NFCC to the DH for the Static RF Connection, as defined in Table 14.
Length of RF Technology Specific Parameters	1 Octet	The length of RF Technology Specific Parameters field to follow.
RF Technology Specific Parameters	0 – n Octets	One of the below tables depending on the value of the RF Technology and Mode; Depends on RF Technology and Mode. See Table 54 for NFC-A Poll Mode. See Table 55 for NFC-A Listen Mode. See Table 56 for NFC-B Poll Mode. See Table 57 for NFC-B Listen Mode. See Table 58 for NFC-F Poll Mode. See Table 59 for NFC-F Listen Mode. Proprietary parameters if the value of RF Technology and Mode is reserved for a proprietary technology.

RF_INTF_ACTIVATED_NTF		
Payload Field(s)	**Length**	**Value/Description**
Data Exchange RF Technology and Mode	1 Octet	RF Technology and Mode that will be used for future Data Exchange. See Table 96.
Data Exchange Transmit Bit Rate	1 Octet	Bit rate that will be used for future Data Exchange in the transmit direction. For a polling device this is the bit rate from poll to listen, and for a listening device this is the bit rate from listen to poll. See Table 97.
Data Exchange Receive Bit Rate	1 Octet	Bit Rate that will be used for future Data Exchange in the receive direction. For a polling device this is the bit rate from listen to poll, and for a listening device this is the bit rate from poll to listen. See Table 97.
Length of Activation Parameters	1 Octet	The length of Activation Parameters field to follow.
Activation Parameters	0 – n Octets	Activation Parameters are defined in the RF Interface section that corresponds to the RF Interface value. If a proprietary interface is activated, proprietary parameters MAY be used. For a list of possible ISO-DEP RF Interface Activation Parameters, see Table 76, Table 77, Table 78, and Table 79. For a list of possible NFC-DEP RF Interface Activation Parameters, see Table 82 and Table 83. There are no Activation Parameters defined for the Frame RF Interface.

The RF Interface to be activated is selected based on the current RF Interface Mapping configuration (see Section 6.2) or the parameter in the RF_DISCOVER_SELECT_CMD (see Section 7.2).

Depending on the selected RF Discovery ID and RF Protocol, the NFCC performs protocol activation procedures before activating the RF Interface. The protocol activation is described in each RF Interface section.

When all phases before RF Interface activation are performed successfully, the NFCC SHALL send RF_INTF_ACTIVATED_NTF with information about the activated RF Interface (RF Interface field).

After the NFCC sends RF_INTF_ACTIVATED_NTF, the Static RF Connection may be used. The Max Data Packet Payload Size and Initial Number of Credits parameters of the RF_INTF_ACTIVATED_NTF apply to the Static RF Connection.

NOTE The NFCC may choose any permitted values for Max Data Packet Payload Size and Initial Number of Credits when activating an RF Interface, regardless of any values that may have been used in previous activations.

The Data Exchange RF Technology and Mode, Data Exchange Transmit Bit Rate and Data Exchange Receive Bit Rate parameters inform the DH about the RF Technology and Bit Rates used in the further data exchange with the Remote NFC Endpoint.

RF_INTF_ACTIVATED_NTF can include Activation Parameters. Activation Parameters depend on the RF Interface field. Each RF Interface section defines the parameters to be included.

Other parameters in the RF_INTF_ACTIVATED_NTF are the same as in the RF_DISCOVER_NTF.

The RF Discovery ID value communicated in an RF_INTF_ACTIVATED_NTF SHALL be valid until the state is changed to **RFST_IDLE**.

7.3.2 RF Interface Deactivation

These Control Messages are used to deactivate an active RF Interface (to stop communication between the DH or NFCEE with a Remote NFC Endpoint) or to stop the Discovery process.

Table 62: Control Messages for RF Interface Deactivation

RF_DEACTIVATE_CMD		
Payload Field(s)	**Length**	**Value/Description**
Deactivation Type	1 Octet	See Table 63.

RF_DEACTIVATE_RSP		
Payload Field(s)	**Length**	**Value/Description**
Status	1 Octet	See Table 94

RF_DEACTIVATE_NTF		
Payload Field(s)	**Length**	**Value/Description**
Deactivation Type	1 Octet	See Table 63.
Deactivation Reason	1 Octet	See Table 64

Table 63: Deactivation Types

Deactivation Type	Description
0x00	Idle Mode
0x01	Sleep Mode
0x02	Sleep_AF Mode
0x03	Discovery
0x04 – 0xFF	RFU

NOTE ‘Sleep Mode’ and Sleep_AF Mode’ refer to the sleep states of Listen Mode state machine defined in [ACTIVITY]. Depending on the technology, ‘Sleep Mode’ refers to SLEEP_A for NFC-A, SLEEP_B for NFC-B or IDLE for NFC-F. ‘Sleep_AF refers to the SLEEP_AF state.

Table 64: Deactivation Reasons

Deactivation Reason	Description
0x00	DH_Request
0x01	Endpoint_Request
0x02	RF_Link_Loss
0x03	NFC-B_Bad_AFI
0x04 – 0xFF	RFU

The RF State Machine in Section 5.2 specifies for each state the possible deactivation cases.

The following rules apply in addition to the definitions in Section 5.2:

- For the RF_DEACTIVATE_CMD, the Deactivation Type values ‘Sleep Mode’ and ‘Sleep_AF Mode’ are not allowed for all RF Interfaces. Each RF Interface specifies whether it supports these deactivation cases. If an RF_DEACTIVATE_CMD with Deactivation Type set to ‘Sleep Mode’ or ‘Sleep_AF Mode’ is received when using an RF Interface where those values are not allowed, the NFCC SHALL send RF_DEACTIVATE_RSP with a Status of STATUS_REJECTED. The NFCC SHALL NOT send RF_DEACTIVATE_NTF in this case.
- If the Deactivation Type in the RF_DEACTIVATE_CMD was set to ‘Sleep Mode’ or ‘Sleep_AF Mode’ and there was an error in executing the protocol deactivation procedures defined by the RF Interface, the RF_DEACTIVATE_NTF SHALL have the Deactivation Type set to ‘Idle Mode’ and the state SHALL change to **RFST_IDLE**.
- In all other cases and if not defined otherwise in the corresponding interface section, the value of the Deactivation Type parameter of the RF_DEACTIVATE_NTF SHALL be the same as in the RF_DEACTIVATE_CMD. If the RF_DEACTIVATE_NTF is send following a RF_DEACTIVATE_CMD/RSP the Deactivation Reason SHALL be set to ‘DH Request’.
- If the activated RF Interface defines that the NFCC has to perform protocol deactivation procedures, the NFCC SHALL perform those deactivation procedures before sending the RF_DEACTIVATE_NTF.
- Prior to sending an RF_DEACTIVATE_NTF, the NFCC SHOULD send all data pending to be sent to the Remote NFC Endpoint and send all completely received Data Messages to the DH.
- After sending a RF_DEACTIVATE_NTF, the NFCC SHALL stop sending any Data Messages related to the RF Interface.
- Upon receipt of a RF_DEACTIVATE_NTF, the DH SHALL NOT send any Data Messages related to the RF Interface.

After an RF Interface has been deactivated:

- no further communication operations defined by the RF Interface (including Data Messages) SHALL be performed.
- all remaining data in the NFCC and DH buffers that was exchanged in the context of the RF Interface SHALL be removed.

7.4 NFCEE Discovery Request

This notification informs the DH of changes to the list of RF Discovery tasks requested on behalf of NFCEE(s) which are attached to the NFCC. Information is sent in this notification as a list of TLVs as defined in Table 66.

- If an attached NFCEE wishes to start using RF communication for a given combination of RF Protocol / RF Technology and Mode, a TLV is included in which the Type field is set to 0x00, and the value is set according to Table 67.
- If an attached NFCEE wishes to stop using RF communication for a given combination of RF Protocol / RF Technology and Mode, a TLV is included in which the Type field is set to 0x01, and the value is set according to Table 67.

This notification can be sent at any time after the first NFCEE Discovery has been run, even if RF Discovery is already in progress.

The NFCC SHALL consider any previously transmitted RF_NFCEE_DISCOVERY_REQ_NTF with a given NFCEE ID as discarded by the DH when the NFCEE with this NFCEE ID is disabled.

If the NFCEE uses the HCI Protocol as defined in [ETSI_102622] the NFCC may determine the NFCEE RF Discovery requirements based on the opened pipes. Otherwise, the method the NFCC uses to find out the NFCEE’s RF Discovery requirements is implementation specific.

The DH SHOULD take the NFCEE’s request under consideration, based on its own requirements and requests for other NFCEEs. Therefore the action taken by the DH on receipt of this notification is implementation specific.

Table 65: Notification for NFCEE Discovery Request

<table>
<tr><th colspan="5">RF_NFCEE_DISCOVERY_REQ_NTF</th></tr>
<tr><th>Payload Field(s)</th><th>Length</th><th colspan="3">Value/Description</th></tr>
<tr><td>Number Of Information Entries</td><td>1 Octet</td><td colspan="3">The number of information entries to follow (n).</td></tr>
<tr><td rowspan="3">Information Entry [1..n]</td><td rowspan="3">x+2 Octets</td><td>Type</td><td>1 Octet</td><td>One of the types defined in Table 66.</td></tr>
<tr><td>Length</td><td>1 Octet</td><td>The length of Value (x).</td></tr>
<tr><td>Value</td><td>x Octets</td><td>Value of the Information Entry.</td></tr>
</table>

Table 66: TLV Coding for RF NFCEE Discovery Request

Type	Length	Value
0x00	3 Octets	Information about the particular discovery request in this TLV is to be added to the list, value field coded according to Table 67.
0x01	3 Octets	Information about the particular discovery request in this TLV is to be removed from the list, value field coded according to Table 67.
0x02-0x7F		RFU
0x80-0xFF		For proprietary use

Table 67: Value Field for NFCEE Discovery Request Information

Payload Field(s)	Length	Value/Description
NFCEE	1 Octet	An NFCEE ID as defined in Table 84
RF Technology and Mode	1 Octet	An RF Technology and Mode as defined in Table 96.
RF Protocol	1 Octet	An RF Protocol as defined in Table 98.

7.5 RF NFCEE Action

RF NFCEE Action is the mechanism used to indicate that an action involving one of the NFCEEs in the device has occurred that may be of interest to the DH.

Actions could be routing decisions made by the NFCC or the availability of application-level information about a transaction with a Remote NFC Endpoint.

For example, this indication could provide a mechanism for a User Interface application on the DH to perform any application specific behaviors based on the knowledge that a specific NFCEE or application on an NFCEE has been accessed by a Remote NFC Endpoint. For example, the User Interface application could display branding and/or request an action from a consumer.

This notification SHALL NOT be sent in states other than **RFST_LISTEN_ACTIVE**.

Table 68: Notification to Report an NFCEE Action

RF_NFCEE_ACTION_NTF		
Payload Field(s)	**Length**	**Value/Description**
NFCEE ID	1 Octet	An NFCEE ID as defined in Table 84.
Trigger	1 Octet	See Table 69.
Supporting Data Length	1 Octet	The length of Supporting Data field to follow (n)
Supporting Data	n Octets	Depends on Trigger

The Trigger value indicates the type of trigger that has caused this notification to be sent (as defined in Table 69).

Table 69: Trigger in NFCEE Action Notification

Trigger	Description	Supporting Data
0x00	[ISO/IEC_7816-4] SELECT command with an AID	The AID in the SELECT command
0x01	RF Protocol based routing decision	The RF Protocol, see Table 98
0x02	RF Technology based routing decision	The RF Technology, see Table 95
0x03-0x0F	RFU	
0x10	Application initiation	Application specific. Could be an AID for an [ISO/IEC_7816-4] type application
0x11-0xFF	RFU	

If RF_NFCEE_ACTION is set to 0x01, the following triggers apply:

- 0x00 – this trigger SHALL be sent by the NFCC when the NFCC is capable of determining the Application Identifier of the accessed application. In the case of [ISO/IEC_7816-4] SELECT commands (applies to NFC Forum Tag Type 4), the Notification SHALL be sent for each such SELECT command received and the value of the Supporting Data SHALL contain the AID selected. The NFCEE ID field SHALL identify the DH-NFCEE or the NFCEE on which the corresponding application is hosted.
- 0x01 – this trigger SHALL be sent when a routing decision is made based on the RF Protocol. The Supporting Data SHALL contain the corresponding RF Protocol. The ID field SHALL identify the DH-NFCEE or the NFCEE the traffic is routed to.
- 0x02 – this trigger SHALL be sent when a routing decision is made based on the RF Technology. The Supporting Data SHALL contain the corresponding RF Technology. The ID field SHALL identify the DH-NFCEE or the NFCEE the traffic is routed to.
- 0x10 – this trigger SHALL be sent if and when the NFCEE provides the information to the NFCC. The manner in which the application on the NFCEE communicates with the NFCC to provide the information is outside the scope of this specification as is the content of the Supporting Data. The ID field SHALL identify the NFCEE providing the information.

For all cases above, a notification SHALL NOT be sent when the NFCEE ID is set to the value of 0x00 (DH-NFCEE ID).

The DH can configure whether the NFCC is allowed to send NFCEE Action notifications by setting the following configuration parameter:

Table 70: RF_NFCEE_ACTION configuration parameter

ID	Length	Value	Description
RF_NFCEE_ACTION	1 Octet	0x00	The NFCC SHALL NOT send RF NFCEE Action notifications to the DH.
		0x01 (default)	The NFCC SHALL send RF NFCEE Actions to the DH upon the triggers described in this section.
		0x02-0xFF	RFU

8 RF Interfaces

8.1 NFCEE Direct RF Interface

The NFCEE Direct RF Interface is a pseudo interface that is used when the NFCC in state **RFST_DISCOVERY** can determine that the RF communication has to be routed to an NFCEE. One example for such a case is if an NFCEE is directly coupled to the RF (e.g. when the NFC Wired Interface (see [ISO/IEC_28361] is used).

The NFCEE Direct RF Interface does not enable NCI Data Message exchange between the DH and the NFCC (and as a consequence no communication between the DH and the Remote NFC Endpoint). Therefore this RF Interface does not define a Data Mapping or Discovery Configuration. The NFCEE Direct RF Interface can not be mapped to an RF Protocol.

The following sections apply for Poll Mode and Listen Mode.

8.1.1 Discovery and Interface Activation

When the DH has enabled an NFCEE is in state **RFST_DISOVERY** and determines that the RF communication has to be routed to an NFCEE, it sends an RF_INTF_ACTIVATED_NTF to the DH to indicate that this Interface has been activated.

8.1.2 Interface Deactivation

Using this interface deactivation cases with Deactivation Types 'Sleep Mode' or 'Sleep_AF Mode' SHALL NOT be allowed.

8.2 Frame RF Interface

Both the Poll-side and Listen-side Frame RF Interfaces provide access to the Payload of the RF frames exchanged between the NFC Forum device and a Remote NFC Endpoint. The RF frame formats are technology dependent. Any higher layer protocols (for example ISO-DEP or NFC-DEP) must be handled on the DH. In addition, when using Frame RF Interface, the activation of the RF Interface does not always coincide with the completion of Device Activation. For ISO-DEP Protocol, the RATS or ATTRIB commands/responses are not handled by the NFCC but must be handled by the DH, and are sent over NCI as the Payload of Data Messages. The same is true for ATR_REQ/RES and PSL_REQ/RES when activating NFC-DEP Protocol.

All Protocols defined in Table 98 can be mapped to this Interface (see Section 6.2).

8.2.1 Data Mapping between the DH and RF

The DH and the NFCC SHALL only use the Static RF Connection for data communication with a Remote NFC Endpoint.

The DH MAY send a Data Message to the NFCC according to Section 8.2.1.1. The NFCC SHALL populate an RF frame of the currently used Technology with this data and send the RF frame to the Remote NFC Endpoint.

When the NFCC receives an RF frame from the Remote NFC Endpoint, then the NFCC SHALL extract the Payload from the RF frame and send it as Payload of a Data Message to the DH according to Section 8.2.1.2.

NCI Segmentation and Reassembly MAY be applied to Data Messages in either direction.

The format of the Data within the Data message used for the Frame RF Interface (NFC-A / NFC-B / NFC-F) differs depending on the transmission direction of the message.

8.2.1.1 Data from the DH to RF

For NFC-A and NFC-B the Data Message SHALL correspond to the Payload of the Data and Payload Format defined in [DIGITAL] Section 4.4 for NFC-A and 5.4 for NFC-B.

For NFC-F the Data Message SHALL correspond to the SoD and Payload of the Data and Payload Format defined Section 6.4 of [DIGITAL].

After receiving a Data Message, the NFCC SHALL append the appropriate EoD and send the result in an RF Frame of the currently used technology to the Remote NFC Endpoint.

NOTE In addition to the formats in Sections 4.4, 5.4, and 6.4, [DIGITAL] defines Data and Payload Formats for ISO-DEP, NFC-DEP and the Tag Platforms. Mapping those formats to the definitions above result in the following:

In case of ISO-DEP, the Data Message corresponds to the SoD and Payload of an ISO-DEP Block defined in Section 13.1. In case of NFC-DEP, the Data Message corresponds to the SoD and Payload of the Data and Payload Format defined in Section 14.4. For Type 1 and Type 2 Tag Platform, the Data Message includes only the Payload field of the applicable Data and Payload Format sections (8.4 and 9.4). For Type 3 Tag Platform, the Data Message includes the SoD and Payload field of the Data and Payload Format defined in Section 10.4.

Figure 11, Figure 12, and Figure 13 illustrate the mapping between the Data Message Format and the RF frame for each Frame RF Interface when sending the RF frame to the Remote NFC Endpoint.

NOTE These figures show the case where NCI Segmentation and Reassembly feature is not used.

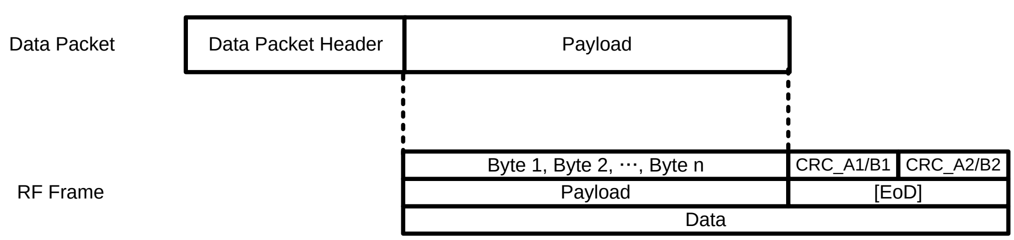

Figure 11: Format for Frame RF Interface (NFC-A) for Transmission

For NFC-A, Data Message SHALL NOT contain the parity bits used by Standard Frame. When sending the RF Frames, the NFCC SHALL insert the parity bits according to [DIGITAL].

For NFC-B, Data Message SHALL NOT contain the start and stop bits used by the frame format. When sending the RF Frames, the NFCC SHALL insert the start and stop bits according to [DIGITAL].

For Type 1 Tag Platform and when in Poll Mode, the first octet of the Data Message SHALL consist of 0b as the most significant bit followed by the 7-bit Command Code. The RF Frame format is defined in Section 8.4 of [DIGITAL], where EoD consists of CRC_B1 and CRC_B2.

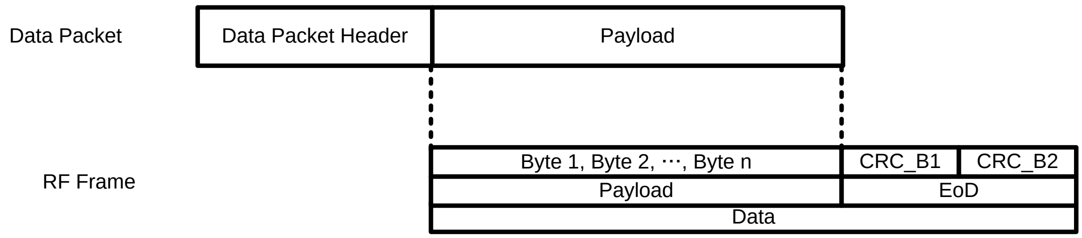

Figure 12: Format for Frame RF Interface (NFC- B) for Transmission

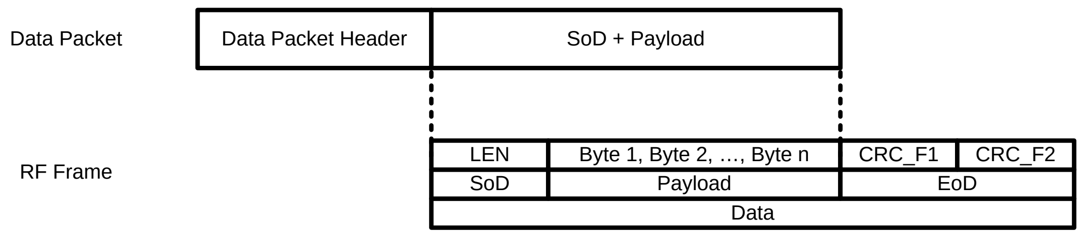

Figure 13: Format for Frame RF Interface (NFC-F) for Transmission

8.2.1.2 Data from RF to the DH

For NFC-A and NFC-B the Data Message SHALL correspond to the Payload of the Data and Payload Format defined in [DIGITAL] Section 4.4 for NFC-A and 5.4 for NFC-B followed by a Status field of 1 octet.

For NFC-F the Data Message SHALL correspond to the SoD and Payload of the Data and Payload Format defined Section 6.4 of [DIGITAL] followed by a Status field of 1 octet.

After receiving an RF frame, the NFCC SHALL check and remove the EoD and send the result in a Data Message to the DH.

NOTE The information in the Note of 8.2.1.1 applies also for the Data Messages send to the DH, except that the Status field is added to each Data Message.

In case of an error the Data Message MAY consist of only a part of the Payload of the received RF frame but SHALL include the trailing Status field. A Data Message consisting of only the Status field is a valid message.

If the RF frame was received correctly, the NFCC SHALL set the Status field of Data Message to a value of STATUS_OK. If the NFCC detected an error when receiving the RF frame, the NFCC SHALL set the Status field of the Data Message to a value of STATUS_RF_FRAME_CORRUPTED (see Table 94).

Figure 14, Figure 15, and Figure 16 illustrate the mapping of the RF frame received from the Remote NFC Endpoint to the Data Message format to be sent to the DH for each Technology.

NOTE These figures show the case where NCI Segmentation and Reassembly feature is not used.

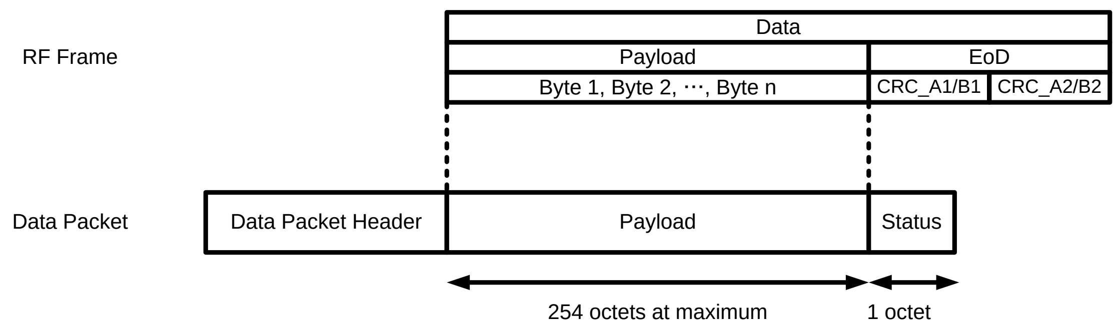

Figure 14**: Format for Frame RF Interface (NFC-A) for Reception**

For NFC-A, Data Message SHALL NOT contain the parity bits present in the Standard Frame and Bit Oriented Frame. When receiving RF Frames, the NFCC SHALL check and remove these parity bits according to [DIGITAL].

For NFC-B, Data Message SHALL NOT contain the start and stop bits used by NFC-B frame format. When receiving the RF Frames, the NFCC SHALL remove the start and stop bits according to [DIGITAL].

For Type 1 Tag Platform, the RF Frame format is defined in Section 8.4 of [DIGITAL], where EoD consists of CRC_B1 and CRC_B2.

For Type 2 Tag Platform, when the NFCC in Poll Mode receives 4-bit ACK or NACK RF Frames from the Remote NFC Endpoint, the NFCC SHALL populate these 4 bits to the lower order bits of the first and only payload octet while each higher order bit is set to 0b.

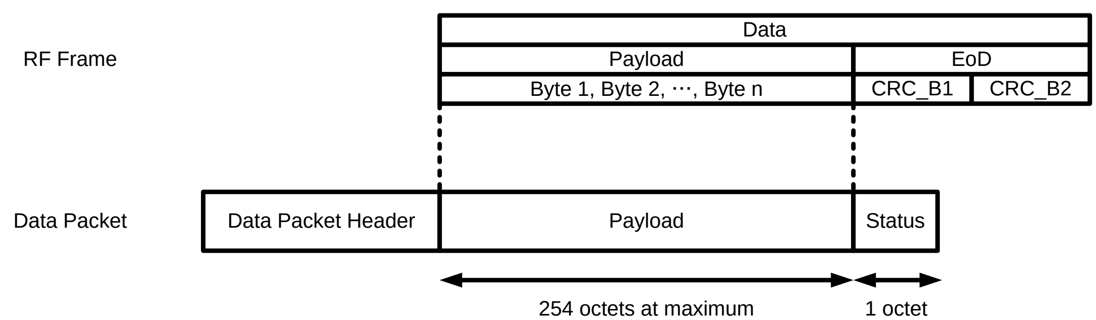

Figure 15: Format for Frame RF Interface (NFC-B) for Reception

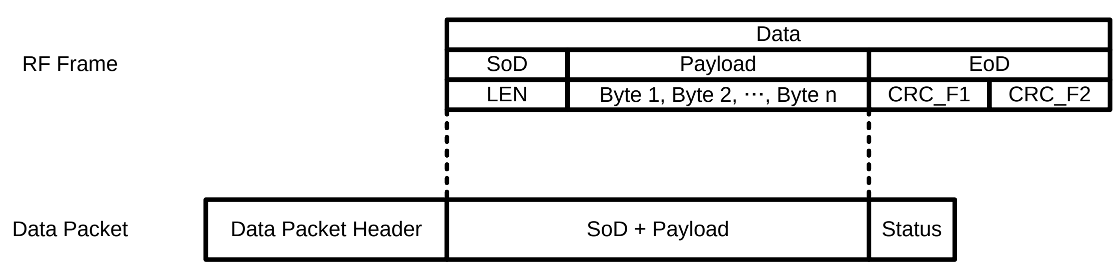

Figure 16: Format for Frame RF Interface (NFC-F) for Reception

8.2.2 Frame RF Interface specific Control Messages

8.2.2.1 RF Communication Parameter Update

These Control Messages are used to update RF Communication parameters once the Frame RF Interface has been activated.

The RF_PARAMETER_UPDATE_CMD SHALL NOT be sent in states other than **RFST_POLL_ACTIVE** and **RFST_LISTEN_ACTIVE**.

Table 71: Control Messages for RF Parameter Update

RF_PARAMETER_UPDATE_CMD				
Payload Field(s)	**Length**	**Value/Description**		
Number of Parameters	1 Octet	**The number of RF Communication Parameter fields to follow (n).**		
RF Communication Parameter [1..n]	x+2 Octets	ID	1 Octet	The identifier of the RF Communication Parameter as defined in Table 72.
		Length	1 Octet	The length of Value (x).
		Value	x Octets	Value of the RF Communication Parameter.

RF_PARAMETER_UPDATE_RSP		
Payload Field(s)	**Length**	**Value/Description**
Status	1 Octet	See Table 94.
Number of Parameters	1 Octet	The number of RF Communication Parameter ID fields to follow (n). Value SHALL be 0x00 and no Parameter IDs listed unless Status = STATUS_INVALID_PARAM.
RF Communication Parameter ID [0..n]	1 Octet	The identifier of the invalid RF Communication Parameter. See Table 72 for a list of IDs.

Table 72: TLV Coding for RF Communication Parameter ID

Type	Length	Value
0x00	1 Octet	RF Technology and Mode, coded as defined in Table 96.
0x01	1 Octet	Transmit Bit Rate, coded as defined in Table 97.
0x02	1 Octet	Receive Bit Rate, coded as defined in Table 97.
0x03	1 Octet	NFC-B Data Exchange Configuration, coded as defined in Table 73.
0x04-0x7F		RFU
0x80-0xFF		Proprietary

If any of the RF Communication parameters listed in Table 72 to be used for Data Exchange differ from those used during Device Activation, the DH SHALL send the new values to the NFCC using an RF_PARAMETER_UPDATE_CMD.

Not all RF Communication parameter settings are permissible in all modes of operation; the DH is responsible for ensuring values sent to the NFCC are correct. There is no obligation for the NFCC to check whether a given parameter value is permitted.

The RF Technology and Mode parameter specifies the RF Technology and Mode that SHALL be used by the NFCC when transmitting and receiving. Refer to [DIGITAL] for permitted values of RF Technology and Mode for a given RF Interface activation.

The Transmit Bit Rate parameter specifies the bit rate that SHALL be used by the NFCC when transmitting. For a polling device this is the bit rate from poll to listen, and for a listening device this is the bit rate from listen to poll. Refer to [DIGITAL] for permitted values of bit rate for a given RF Interface activation.

The Receive Bit Rate parameter specifies the bit rate that SHALL be used by the NFCC when receiving. For a polling device this is the bit rate from listen to poll, and for a listening device this is the bit rate from poll to listen. Refer to [DIGITAL] for permitted values of bit rate for a given RF Interface activation.

The NFC-B Data Exchange Configuration parameter specifies a number of NFC-B related values. Not all values are relevant to a given operating mode. Values that are relevant to the current RF Communication State SHALL be used by the NFCC as defined in [DIGITAL] during subsequent Data Exchange. They consist of Minimum TR0, Minimum TR1, Minimum TR2, Suppression of SoS, and Suppression of EoS. The format of the octet is defined below.

Table 73: NFC-B Data Exchange Configuration Parameter

	Bit Mask								Description
	b7	**b6**	**b5**	**b4**	**b3**	**b2**	**b1**	**b0**	
Octet 0	X	X							Minimum TR0 as defined in [DIGITAL]
			X	X					Minimum TR1 as defined in [DIGITAL]
					X				Suppression of EoS as defined in [DIGITAL]
						X			Suppression of SoS as defined in [DIGITAL]
							X	X	Minimum TR2 as defined in [DIGITAL]

On receipt of the RF_PARAMETER_UPDATE_CMD Command,

- if the NFCC is in state **RFST_POLL_ACTIVE**, it SHALL update the RF Communication parameters contained within the Command, before responding with an RF_PARAMETER_UPDATE_RSP.

  NOTE For NFC-DEP RF Protocol, the RF Communication parameters need to be aligned with the parameter values contained in the PSL_REQ, and the update Command is issued after the PSL_RES is received from the Remote NFC Endpoint, but before the first DEP_REQ is sent.

- if the NFCC is in state **RFST_LISTEN_ACTIVE**, it SHALL respond with an RF_PARAMETER_UPDATE_RSP and then SHALL wait until it has sent the next RF Frame before updating the RF Communication parameters contained within the Command.

  NOTE For NFC-DEP RF Protocol, the RF Communication parameters need to be aligned with the parameter values contained in the PSL_REQ, and the update Command is issued after the PSL_REQ is received from the Remote NFC Endpoint, but before the PSL_RES is sent.

The Status field in the RF_PARAMETER_UPDATE_RSP indicates whether the setting of these RF Communication parameters was successful or not. A Status of STATUS_OK SHALL indicate that all RF Communication parameters have been set to these new values within the NFCC.

If the DH tries to set a parameter which is not applicable for the NFCC, the NFCC SHALL respond with an RF_PARAMETER_UPDATE_RSP with a Status field of STATUS_INVALID_PARAM and including one or more invalid RF Communication parameter ID(s). All other RF Communication parameters SHALL have been set to the new values within the NFCC.

8.2.2.2 Request Type 3 Tag Polling

In the **RFST_POLL_ACTIVE** state, the SENSF command, which is also part of the T3T command set, cannot be handled by the DH over the Frame RF Interface, therefore specific Commands are introduced for this case.

The following Control Messages are used to request the NFCC to send a Type 3 Tag Polling Command.

Table 74: Control Messages to Request the NFCC to send a Type 3 Tag Polling Command

RF_T3T_POLLING_CMD		
Payload Field(s)	**Length**	**Value/Description**
SENSF_REQ_PARAMS	4 Octets	Byte 2-5 of SENSF as defined in Section 6.6.1 of [DIGITAL].

RF_T3T_POLLING_RSP		
Payload Field(s)	**Length**	**Value/Description**
Status	1 Octet	See Table 94.

<table>
<tr><th colspan="5">RF_T3T_POLLING_NTF</th></tr>
<tr><th>Payload Field(s)</th><th>Length</th><th colspan="3">Value/Description</th></tr>
<tr><td>Status</td><td>1 Octet</td><td colspan="3">See Table 94.</td></tr>
<tr><td>Number of Responses</td><td>1 Octet</td><td colspan="3">The number of Response fields to follow (n).</td></tr>
<tr><td rowspan="2">Response [1..n]</td><td rowspan="2">m+1 Octets</td><td>Length</td><td>1 Octet</td><td>Length of following SENSF_RES field (m).
Allowed values SHALL be 0x10 or 0x12. Other values are RFU.</td></tr>
<tr><td>SENSF_RES</td><td>m Octets</td><td>Byte 2-17/19 of SENSF_RES Response as defined in [DIGITAL].</td></tr>
</table>

The DH is allowed to use this Command only

- if in **RFST_POLL_ACTIVE** state and
- if the activated RF Interface is the Poll-side Frame RF Interface for PROTOCOL_T3T

After receiving a RF_T3T_POLLING_CMD and if above conditions are met, the NFCC SHALL answer with a RF_T3T_POLLING_RSP with a Status value of STATUS_OK and send a SENSF_REQ Command with Bytes 2-5 set as specified in the Command parameter to the RF field.

When the NFCC receives SENSF_RES response(s) from the Remote NFC Endpoint (s), the NFCC SHALL populate the corresponding parameters of a RF_T3T_POLLING_NTF and send it with Status set to STATUS_OK to the DH. If no response is received, the NFCC SHALL send a RF_T3T_POLLING_NTF with Status set to STATUS_FAILED and containing no other Payload Fields.

If above conditions are not met, the NFCC SHALL send a RF_T3T_POLLING_RSP with a Status value of STATUS_SEMANTIC_ERROR. In this case the NFCC SHALL NOT send a RF_T3T_POLLING_NTF.

8.2.3 Poll-side Frame RF Interface Management

This Interface can be used if the NFC Forum Device is operating in Poll Mode (Reader/Writer Mode or Peer Mode – NFC-DEP Initiator).

8.2.3.1 Discovery Configuration

The Discovery configuration parameters defined in the following tables are relevant depending on the NFC technology used in Discovery (RF Technology and Mode in RF_DISCOVER_CMD). They MAY be changed from defaults by using the CORE_SET_CONFIG_CMD before moving to state **RFST_DISCOVERY**.

- Table 23: Discovery Configuration Parameters for Poll A
- Table 24: Discovery Configuration Parameters for Poll B
- Table 26: Discovery Configuration Parameters for Poll F
- Table 41: Common Parameters for Discovery Configuration

8.2.3.2 Discovery and Interface Activation

To enable Poll Mode, the DH sends the RF_DISCOVER_CMD to the NFCC containing at least one configuration for a Poll Mode RF Technology and Mode.

In the case where multiple Remote NFC Endpoints are detected and the DH has selected the NFC Endpoint to be used or in the case only a single Remote NFC Endpoint is detected, then the NFCC sends an RF_INTF_ACTIVATED_NTF to the DH to indicate that this Interface has been activated.

The protocol activation is fully under the control of the DH when the Frame RF Interface is used, therefore for all RF Technologies the RF_INTF_ACTIVATED_NTF SHALL NOT contain any Activation Parameters.

8.2.3.3 Interface Deactivation

The deactivation cases for the Poll-side Frame RF Interface are as described in Section 5.2 for **RFST_POLL_ACTIVE**.

8.2.3.4 Failures during Data Exchange

Even if there are any failures during data exchange, the NFCC SHALL NOT send any CORE_INTERFACE_ERROR_NTF to the DH. The DH itself identifies TRANSMISSION ERROR, PROTOCOL ERROR, or TIMEOUT ERROR in communication with the Remote NFC Endpoint.

The DH MAY deactivate the RF Interface when it identifies such errors.

8.2.4 Listen-side Frame RF Interface Management

This Interface can be used if the NFC Forum Device is operating in Listen Mode (Card Emulation Mode or Peer Mode – NFC-DEP Target).

8.2.4.1 Discovery Configuration

The Discovery configuration parameters defined in the following tables are relevant depending on the NFC technology used in Discovery (RF Technology and Mode in RF_DISCOVER_CMD). They MAY be changed from defaults by using the CORE_SET_CONFIG_CMD before moving to state **RFST_DISCOVERY** state.

- Table 30: Discovery Configuration Parameters for Listen A
- Table 32: Discovery Configuration Parameters for Listen B
- Table 35: Discovery Configuration Parameters for Listen F
- Table 38: Discovery Configuration Parameters for Listen ISO-DEP
  Using this Interface, the LI_FWI parameter is only applicable for NFC-B.

In the case of NFC Forum Type 4 Tag Emulation by the DH, the DH sends the CORE_SET_CONFIG_CMD to configure the Listen A Parameters and/or the Listen B Parameters as defined in Table 30 and Table 32.

In the case of NFC Forum Type 3 Tag Emulation by the DH, the DH sends the CORE_SET_CONFIG_CMD to configure the Listen F Parameters as defined in Table 35.

The RF_SET_LISTEN_MODE_ROUTING_CMD MAY be used to route the received data from the Remote NFC Endpoint to specific NFCEEs which will in this case include routing some or all received data to the DH (for example if a NFC Forum Type 4 Tag is being emulated by an NFCEE on the DH.)

8.2.4.2 Discovery and Interface Activation

To enable Listen Mode, the DH sends the RF_DISCOVER_CMD to the NFCC containing at least one configuration for a Listen Mode RF Technology and Mode.

When the NFCC has, based on the communication with the Remote NFC Endpoint (see [ACTIVITY]), determined whether to activate the Frame RF Interface, the NFCC sends the RF_INTF_ACTIVATED_NTF to the DH to indicate that this Interface has been activated to be used for communication with the specified Remote NFC Endpoint.

To provide the RF Protocol information in the RF_INTF_ACTIVATED_NTF, the NFCC needs to recognize the corresponding protocol activation command (for example ATTRIB, RATS or ATR_REQ) before activating the Frame RF Interface and forwarding the command to the DH. If the NFCC has not determined which protocol is being activated, the RF Protocol value SHALL be PROTOCOL_UNDETERMINED.

There are five different states in the Listen state machine, defined in [ACTIVITY], from which the NFCC may activate the Frame RF Interface. These five “pre-activation” states are listed in the Table 75. Once in one of these pre-activation states, some commands received from the remote NFC Endpoint are managed by the NFCC, others are forwarded to and then managed by the DH. Table 75 details which commands the NFCC handles when using the Frame RF Interface in Listen Mode. Any other commands are forwarded to the DH.

Table 75: Pre-activation states and the split of commands between NFCC & DH:

Technology	Pre-activation state	Commands handled by the NFCC
NFC-A	ACTIVE_A & ACTIVE_A*	SENS_REQ, ALL_REQ, SLP_REQ, SDD_REQ, SEL_REQ, Command with Transmission Error (as defined in [DIGITAL], e.g. CRC errors, Parity errors)
NFC-B	READY_B_DECL	SENSB_REQ, ALLB_REQ, SLPB_REQ, Command with Transmission Error (as defined in [DIGITAL], e.g. CRC errors)
NFC-F	IDLE & READY_F	SENSF_REQ Command with Transmission Error (as defined in [DIGITAL], e.g. CRC errors)

When the NFCC receives a command while it is in one of these pre-activation states:

- based on the information in Table 75, if the command has to be managed by the DH and if configured accordingly, the NFCC SHALL activate the Frame RF Interface.
- In any other case, the NFCC SHALL not activate the Frame RF Interface.

The protocol activation is fully under the control of the DH when the Frame RF Interface is used, therefore for all RF Technologies the RF_INTF_ACTIVATED_NTF SHALL NOT contain any Activation Parameters.

NOTE Multiple SENSF_RES responses may be sent from one or more of the NFCEEs connected to the NFCC. In this case, the NFCC has to assign a time slot to each SENSF_RES response prior to sending them to the Remote NFC Endpoint. The NFCC sends the RF_INTF_ACTIVATED_NTF corresponding to the RF Protocol indicated by the RF Frame received after sending SENSF_RES responses to the Remote NFC Endpoint.

8.2.4.3 Interface Deactivation

Additional rules apply for the following deactivation cases described in Section 5.2 for **RFST_LISTEN_ACTIVE**:

- RF_DEACTIVATE_CMD/RSP/NTF(Sleep Mode)

  Before sending the RF_DEACTIVATE_NTF, the NFCC SHALL move to the following state of the Listen Mode state machine defined in [ACTIVITY]

  - SLEEP_A when NFC-A is the technology currently in use
  - SLEEP_B when NFC-B is the technology currently in use
  - IDLE when NFC-F is the technology currently in use
- RF_DEACTIVATE_CMD/RSP/NTF(Sleep_AF Mode)

Before sending the RF_DEACTIVATE_NTF, the NFCC SHALL move to the following state of the Listen Mode state machine defined in [ACTIVITY]

- SLEEP_AF when NFC-A or NFC-F is the technology currently in use

If the currently used technology is NFC-B, the NFCC SHALL respond with a RF_DEACTIVATE_RSP with Status set to STATUS_REJECTED, not send a RF_DEACTIVATE_NTF and keep the Frame RF Interface activated.

- RF_DEACTIVATE_NTF(Sleep Mode, Endpoint Request)

After receiving a SLP_REQ or sending a SLPB_RES the NFCC SHALL send the DEACTIVATE_NTF.

When using this RF Interface, the following deactivation cases described in Section 5.2 for **RFST_LISTEN_ACTIVE** SHALL NOT be allowed:

- RF_DEACTIVATE_NTF(Discovery, Endpoint_Request)
- RF_DEACTIVATE_NTF(Sleep_AF Mode, Endpoint Request)

8.2.4.4 Failures during Data Exchange

Even if there are any failures during data exchange, the NFCC SHALL NOT send any CORE_INTERFACE_ERROR_NTF to the DH.

The DH itself identifies TRANSMISSION ERROR and PROTOCOL ERROR in communication with the Remote NFC Endpoint from the Data in the Data Packet.

The DH MAY deactivate the RF Interface when it identifies such errors.

8.3 ISO-DEP RF Interface

Both the Poll-side and Listen-side ISO-DEP RF Interface provides access to the Payload of the ISO-DEP I-Blocks exchanged between the NFC Forum Device and a Remote NFC Endpoint. Using this interface, the DH need not have any knowledge about ISO-DEP block format, however any higher layer protocol (e.g. based on 7816 APDU exchange) must be handled on the DH.

The following Protocols can be mapped to this Interface (see Section 6.2): PROTOCOL_ISO_DEP.

8.3.1 Data Mapping between the DH and RF

The DH and the NFCC SHALL only use the Static RF Connection for data communication with a Remote NFC Endpoint.

NCI Segmentation and Reassembly MAY be applied to Data Messages in either direction.

8.3.1.1 Data from the DH to RF

When receiving a Data Message from the DH, the NFCC SHALL send the data contained in the Data Message as a Payload of an I-Block or as multiple Payloads of a chained series of I-Blocks to the Remote NFC Endpoint using the activated technology.

The last octet of the Data Packet with the Packet Boundary Flag set to 0b SHALL be the last byte of the Payload of an I-Block not indicating chaining sent by the NFCC to the Remote NFC Endpoint.

Figure 17 illustrates the mapping between the Data Packet and the RF Frame when sending the frame to the Remote NFC Endpoint.

NOTE This figure shows the case where NCI Segmentation and Reassembly feature is not used. However, ISO-DEP chaining is used where one Data Message is sent with multiple I-Blocks to the Remote NFC Endpoint.

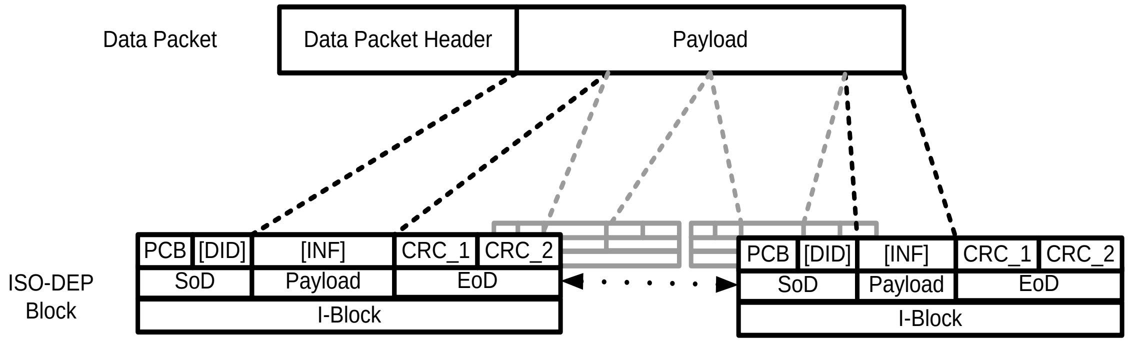

Figure 17: Format for ISO-DEP RF Interface for Transmission

8.3.1.2 Data from RF to the DH

The NFCC SHALL transfer the Payload of received I-Blocks to the DH.

The Payload of an ISO-DEP I-Block not indicating chaining or the combined Payload of a chained series of ISO-DEP I-Blocks SHALL be sent as one Data Message.

The last byte of the Payload of an I-Block not indicating chaining received by the NFCC from the Remote NFC Endpoint SHALL be the last octet of the Data Packet with Packet Boundary Flag set to 0b.

Figure 18 illustrates the mapping between the NCI Data Message format and the ISO-DEP I-blocks for the ISO-DEP RF Interface.

NOTE This figure shows the case where NCI Segmentation and Reassembly feature is not used. However, ISO-DEP chaining is used where one Data Message is received by multiple I-Blocks from the Remote NFC Endpoint.

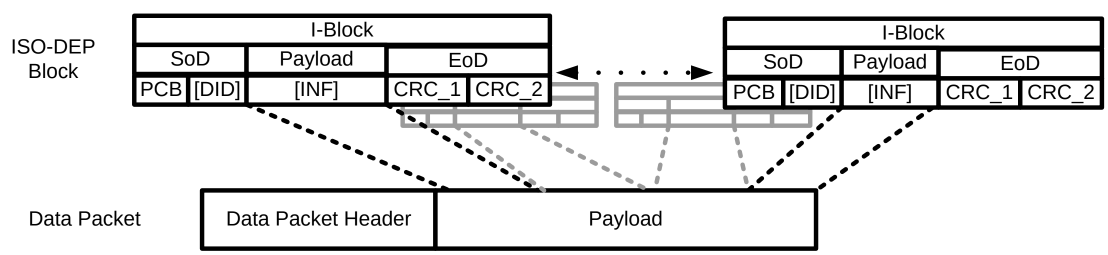

Figure 18: Format for ISO-DEP RF Interface for Reception

8.3.2 Poll-side ISO-DEP RF Interface Management

The Poll-side ISO-DEP RF Interface MAY be used for Reader/Writer Mode when the Remote NFC Endpoint supports ISO-DEP based on NFC-A or NFC-B.

8.3.2.1 Discovery Configuration

The Discovery configuration parameters defined in the following tables are relevant depending on the NFC technology used in Discovery (NFC-A or NFC-B as set by RF Technology and Mode in RF_DISCOVER_CMD). They MAY be changed from defaults by using the CORE_SET_CONFIG_CMD before moving to state **RFST_DISCOVERY**.

- Table 23: Discovery Configuration Parameters for Poll A
- Table 24: Discovery Configuration Parameters for Poll B
- Table 27: Discovery Configuration Parameters for ISO-DEP

8.3.2.2 Discovery and Interface Activation

To enable Poll Mode for ISO-DEP, the DH sends the RF_DISCOVER_CMD to the NFCC containing configurations with RF Technology and Mode values of NFC_A_PASSIVE_POLL_MODE and/or NFC_B_PASSIVE_POLL_MODE.

When the NFCC is ready to exchange data (that is, after receiving a response to the protocol activation command from the Remote NFC Endpoint), it sends the RF_INTF_ACTIVATED_NTF to the DH to indicate that this Interface has been activated to be used with the specified Remote NFC Endpoint.

Detailed ISO-DEP RF Interface activation handling in the NFCC:

For NFC-A:

Following the anticollision sequence, if the Remote NFC Endpoint supports ISO-DEP Protocol, the NFCC sends the RATS Command to the Remote NFC Endpoint and after receiving the RATS response, the NFCC SHALL send the RF_INTF_ACTIVATED_NTF to the DH to indicate a Remote NFC Endpoint based on ISO-DEP has been activated.

For NFC-A the RF_INTF_ACTIVATED_NTF SHALL include the Activation Parameters defined in Table 76.

Table 76: Activation Parameters for NFC-A/ISO-DEP Poll Mode

Parameter	Length	Description
RATS Response Length	1 Octet	Length of RATS Response Parameter (n)
RATS Response	n Octets	All Bytes of the RATS Response as defined in [DIGITAL] starting from and including Byte 2.

For NFC-B:

Following the anticollision sequence, if the Remote NFC Endpoint supports ISO-DEP Protocol, the NFCC sends the ATTRIB command to the Remote NFC Endpoint and following receipt of the ATTRIB response, the NFCC SHALL send the RF_INTF_ACTIVATED_NTF to the DH to indicate a Remote NFC Endpoint based on ISO-DEP has been activated.

For NFC-B the RF_INTF_ACTIVATED_NTF SHALL include the Activation Parameters defined in Table 77.

Table 77: Activation Parameters for NFC-B/ISO-DEP Poll Mode

Parameter	Length	Description
ATTRIB Response Length	1 Octet	Length of ATTRIB Response Parameter (n)
ATTRIB Response	n Octets	ATTRIB Response as defined in [DIGITAL]

8.3.2.3 Interface Deactivation

Additional rules apply for the following deactivation cases described in Section 5.2 for **RFST_POLL_ACTIVE**:

- RF_DEACTIVATE_CMD/RSP/NTF(Idle Mode),
  RF_DEACTIVATE_CMD/RSP/NTF(Discovery)
  RF_DEACTIVATE_CMD/RSP/NTF(Sleep Mode)

  Before sending the RF_DEACTIVATE_NTF to the DH, the NFCC SHALL de-activate the Remote NFC Endpoint according to the De-activation rules for ISO-DEP specified in [DIGITAL].

When using this RF Interface, the following deactivation cases described in Section 5.2 for **RFST_LISTEN_ACTIVE** SHALL NOT be allowed:

- RF_DEACTIVATE_CMD/RSP/NTF(Sleep_AF Mode)

8.3.2.4 Failures during Data Exchange

If an ISO-DEP Timeout Error PROTOCOL EXCEPTION occurs, the NFCC SHALL send the CORE_INTERFACE_ERROR_NTF with the Status of RF_TIMEOUT_ERROR.

If an ISO-DEP Transmission Error PROTOCOL EXCEPTION occurs, the NFCC SHALL send the CORE_INTERFACE_ERROR_NTF with the Status of RF_TRANSMISSION_ERROR.

If an ISO-DEP Protocol Error PROTOCOL EXCEPTION occurs, the NFCC SHALL send the CORE_INTERFACE_ERROR_NTF with the Status of RF_PROTOCOL_ERROR.

8.3.3 Listen-side ISO-DEP RF Interface Management

This Listen-side ISO-DEP RF Interface MAY be used for Card Emulation Mode when the Remote NFC Endpoint uses ISO-DEP based on NFC-A or NFC-B.

In Listen Mode, if the NFCC receives an I-Block from the Remote NFC Endpoint with an empty Payload as part of the presence check mechanism defined in [ISO/IEC_14443], the NFCC SHALL respond with an I-Block and not forward the received I-Block to the DH.

8.3.3.1 Discovery Configuration

The Discovery configuration parameters defined in the following tables are relevant depending on the NFC technology used in Discovery (NFC-A or NFC-B as set by RF Technology and Mode in RF_DISCOVER_CMD). They MAY be changed from defaults by using the CORE_SET_CONFIG_CMD before moving to state **RFST_DISCOVERY**.

- Table 30: Discovery Configuration Parameters for Listen A
- Table 32: Discovery Configuration Parameters for Listen B

  Except parameter LB_SFGI, which SHALL be configured by the NFCC when using this Interface.

- Table 38: Discovery Configuration Parameters for Listen ISO-DEP

To use this Interface, at least one of the Parameters LA_SEL_INFO and LB_SENSB_INFO SHALL be configured to indicate ISO-DEP support.

In the case of Type 4 Tag Emulation being hosted on the DH, the DH sends the CORE_SET_CONFIG_CMD to configure the Listen A Parameters and/or the Listen B Parameters defined in Table 30 and Table 32, and to set the Selected Interface to ISO-DEP.

The DH MAY send the RF_SET_LISTEN_MODE_ROUTING_CMD to set the routing information on the NFCC which will route the received data to the specific NFCEEs and in this case to the DH.

8.3.3.2 Discovery and Interface Activation

To enable Listen Mode for ISO-DEP, the DH sends the RF_DISCOVER_CMD to the NFCC containing configurations with RF Technology and Mode values of NFC_A_PASSIVE_LISTEN_MODE and/or NFC_B_PASSIVE_LISTEN_MODE.

When the NFCC is ready to exchange data with the DH (that is, after receiving protocol activation command from the Remote NFC Endpoint), it sends the RF_INTF_ACTIVATED_NTF to the DH to indicate that this Interface has been activated to be used with the specified Remote NFC Endpoint.

Detailed ISO-DEP RF Interface activation handling in the NFCC:

For NFC-A:

Following the anticollision sequence, the NFCC sends the SEL_RES – indicating according to [DIGITAL] that the NFC Forum Device may support the Type 4A Tag Platform – to the Remote NFC Endpoint. The NFCC then receives the RATS command and sends the RATS response to the Remote NFC Endpoint. The NFCC SHALL then send the RF_INTF_ACTIVATED_NTF to the DH to indicate that a Remote NFC Endpoint based on ISO-DEP has been detected.

For NFC-A the RF_INTF_ACTIVATED_NTF SHALL include the Activation Parameters defined in Table 78.

Table 78: Activation Parameters for NFC-A/ISO-DEP Listen Mode

Parameter	Length	Description
RATS Command Param	1 Octet	Byte 2 (‘PARAM’) of the RATS Command as defined in [DIGITAL]

For NFC-B:

Following the anticollision sequence, the NFCC sends the SENSB_RES – indicating the NFC Forum Device may support the Type 4B Tag Platform (bit 1 = 1b in the Protocol Type) – to the Remote NFC Endpoint. The NFCC then receives the ATTRIB Command and sends the ATTRIB Response to the Remote NFC Endpoint. The NFCC SHALL then send the RF_INTF_ACTIVATED_NTF to the DH to indicate that a Remote NFC Endpoint based on ISO-DEP has been detected.

For NFC-B, the RF_INTF_ACTIVATED_NTF SHALL include the Activation Parameters defined in Table 79.

Table 79: Activation Parameters for NFC-B/ISO-DEP Listen Mode

Parameter	Length	Description
ATTRIB Command Length	1 Octet	Length of ATTRIB Command Parameter (n)
ATTRIB Command	n Octets	All Bytes of the ATTRIB Command as defined in [DIGITAL] starting from and including Byte 2.

8.3.3.3 Interface Deactivation

Additional rules apply for the following deactivation cases described in Section 5.2 for **RFST_LISTEN_ACTIVE**:

- RF_DEACTIVATE_NTF(Sleep Mode, Endpoint Request)

  After sending an S(DESELECT) response block to the Remote NFC Endpoint, the NFCC SHALL send the RF_DEACTIVATE_NTF.

When using this RF Interface, the following deactivation cases described in Section 5.2 for **RFST_LISTEN_ACTIVE** SHALL NOT be allowed:

- RF_DEACTIVATE_NTF(Discovery, Endpoint_Request)
- RF_DEACTIVATE_NTF(Sleep_AF Mode, Endpoint Request)
- RF_DEACTIVATE_CMD/RSP/NTF(Sleep_AF Mode)
- RF_DEACTIVATE_CMD/RSP/NTF(Sleep Mode)

8.3.3.4 Failures during Data Exchange

If an ISO-DEP Protocol Error PROTOCOL EXCEPTION as defined in [DIGITAL] occurs, the NFCC SHALL send the CORE_INTERFACE_ERROR_NTF with the Status of RF_PROTOCOL_ERROR'.

When the DH receives the CORE_INTERFACE_ERROR_NTF, the DH MAY initiate ISO-DEP RF Interface deactivation or perform some other error handling procedure.

8.4 NFC-DEP RF Interface

Both the Poll-side and Listen-side NFC-DEP RF Interfaces provide access to the Transport Data Bytes of the NFC-DEP frames exchanged between the NFC Forum Device and a Remote NFC Endpoint. Using this interface, the DH does not need to have any knowledge related to the NFC-DEP frame format, however any higher layer protocol (e.g. LLCP exchange) must be handled on the DH.

The following Protocols can be mapped to this Interface (see Section 6.2): PROTOCOL_NFC_DEP.

8.4.1 Data Mapping between the DH and RF

The DH and the NFCC SHALL only use the Static RF Connection for data communication with a Remote NFC Endpoint.

NCI Segmentation and Reassembly MAY be applied to Data Messages in either direction.

8.4.1.1 Data from the DH to RF

When receiving a Data Message from the DH, the NFCC SHALL send the payload data contained in the Data Message as an information PDU transferred within the Transport Data Bytes of one or a chained series of DEP_REQ commands (if the NFCC is an NFC-DEP Initiator) or DEP_RES responses (if the NFCC is an NFC-DEP Target) to the Remote NFC Endpoint using the activated technology.

The last octet of a Data Packet with Packet Boundary Flag set to 0b SHALL be the last byte of the Transport Data Bytes of the DEP_REQ or DEP_RES frame not indicating chaining, sent by the NFCC to the Remote NFC Endpoint.Figure 19 illustrates the mapping between the Data Packet and the NFC-DEP – DEP_REQ frame when sending a frame to the Remote NFC Endpoint.

NOTE This figure shows the case where NCI Segmentation and Reassembly feature is not used. However, NFC-DEP chaining is used where one Data Message is sent with multiple DEP_REQ frames to the Remote NFC Endpoint.

NOTE In case of a DEP_RES frame the first Byte would be 0xD5 and the second Byte 0x07.

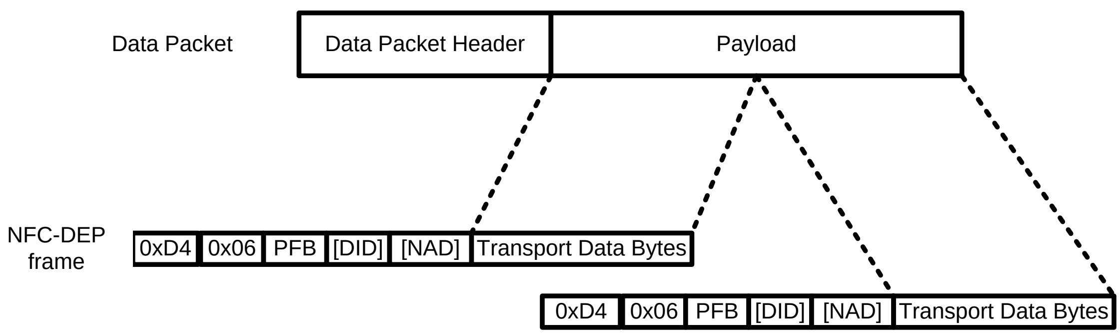

Figure 19: Format for NFC-DEP RF Interface for Transmission

8.4.1.2 Data from RF to the DH

The NFCC SHALL transfer the Information PDU received within the Transport Data Bytes of one or a chained series of DEP_RES responses (if the NFCC is an NFC-DEP Initiator) or DEP_REQ commands (if the NFCC is an NFC-DEP Target) to the DH.

The last byte of the Transport Data Bytes of a DEP_RES or DEP_REQ frame not indicating chaining, received by the NFCC from the Remote NFC Endpoint SHALL be the last octet of the Data Packet with the Packet Boundary Flag set to 0b.

Figure 20 illustrates the mapping between the Data Message format and the NFC-DEP – DEP_RES frame.

NOTE This figure shows the case where NCI Segmentation and Reassembly feature is not used. However, NFC-DEP chaining is used where one Data Message is received by multiple DEP_RES frames from the Remote NFC Endpoint.

NOTE In case of a DEP_REQ frame the first Byte would be 0xD4 and the second Byte 0x06.

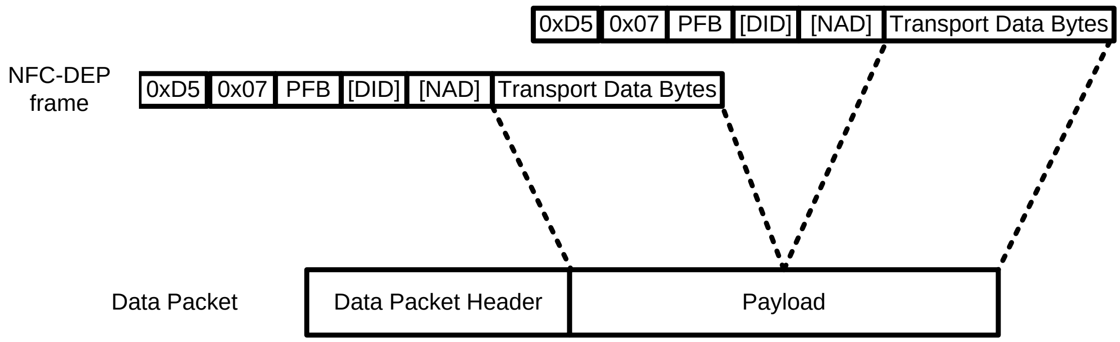

Figure 20: Format for NFC-DEP RF Interface for Reception

8.4.2 NFC-DEP RF Interface Configuration

The behavior of the NFC-DEP protocol MAY be configured using CORE_SET_CONFIG_CMD indicating the following parameter:

Table 80: Specific Parameters for NFC-DEP RF Interface

ID	Length	Description
NFCDEP_OP	1 Octet	See Table 81 Default : 0x0F

Table 81: NFC-DEP Operation Parameter

	Bit Mask								Description
	b7	**b6**	**b5**	**b4**	**b3**	**b2**	**b1**	**b0**	
Octet 0	0	0	0	0					RFU
					X				If set to 1b, all PDUs indicating chaining (MI bit set) SHALL use the maximum number of Transport Data Bytes.
						X			If set to b1, Information PDU with no Transport Data Bytes SHALL NOT be sent
							X		If set to 1b, the NFC-DEP Initiator SHALL use the ATTENTION command only as part of the error recovery procedure as described in [DIGITAL].
								X	If set to 1b, the NFC-DEP Target SHALL NOT send RTOX requests.

This parameter SHALL only be configured when the NFCC is in **RFST_IDLE** state.

Some settings are only relevant for NFC-DEP Target or NFC-DEP Initiator respectively. The settings not matching the current role of the device SHALL be ignored.

NOTE These settings allow configuring the NFC-DEP protocol to comply with the requirements stated in the NFC Forum Logical Link Control Protocol [LLCP] for the NFC-DEP protocol binding.

8.4.3 Poll-side NFC-DEP RF Interface Management

The Poll-side NFC-DEP RF Interface MAY be used when the device operates as an NFC-DEP Initiator when in Peer Mode and the Remote NFC Endpoint supports NFC-DEP based on NFC-A or NFC-F.

8.4.3.1 Discovery Configuration

The Discovery configuration parameters defined in the following tables are relevant depending on the NFC technology used in Discovery (NFC-A or NFC-F as set by RF Technology and Mode in RF_DISCOVER_CMD). They MAY be changed from defaults by using the CORE_SET_CONFIG_CMD before moving to state **RFST_DISCOVERY**.

- Table 23: Discovery Configuration Parameters for Poll A
- Table 26: Discovery Configuration Parameters for Poll F
- Table 28: Discovery Configuration Parameters for Poll NFC-DEP

8.4.3.2 Discovery and Interface Activation

To enable Poll Mode for NFC-DEP, the DH sends the RF_DISCOVER_CMD to the NFCC containing configurations with RF Technology and Mode values of NFC_A_PASSIVE_POLL_MODE and/or NFC_F_PASSIVE_POLL_MODE.

When the NFCC is ready to exchange data (that is, after successful protocol activation), it SHALL send the RF_INTF_ACTIVATED_NTF to the DH to indicate that the NFC-DEP protocol has been activated as follows:

Following the anticollision sequence, if the Remote NFC Endpoint supports NFC-DEP Protocol, the NFCC sends an ATR_REQ with PN_ATR_REQ_GEN_BYTES configured during Discovery Configuration to the Remote NFC Endpoint. When an ATR_RES is received from the Remote NFC Endpoint, the NFCC SHALL forward the ATR_RES to the DH within the Activation Parameters of the RF_INTF_ACTIVATED_NTF. Based on the current bit rate and the configured PN_NFC_DEP_SPEED value, the NFCC determines whether to send a PSL_REQ before sending the RF_INTF_ACTIVATED_NTF.

If PSL_REQ is sent, the values in the three Data Exchange parameters of the RF_INTF_ACTIVATED_NTF SHALL reflect the technology and bit rates set with the PSL_REQ otherwise they SHALL indicate the technology and bit rates used during the protocol activation.

The RF_INTF_ACTIVATED_NTF SHALL include the Activation Parameters defined in Table 82.

Table 82: Activation Parameters for NFC-DEP Poll Mode

Parameter	Length	Description
ATR_RES Response Length	1 Octet	Length of ATR_RES Command Parameter (n)
ATR_RES Response	n Octets	All Bytes of ATR_RES Response starting from and including Byte 3 as defined in [DIGITAL]

8.4.3.3 Interface Deactivation

Additional rules apply for the following deactivation cases described in Section 5.2 for **RFST_POLL_ACTIVE**:

- RF_DEACTIVATE_CMD/RSP/NTF(Idle Mode), RF_DEACTIVATE_CMD/RSP/NTF(Discovery)

  Before sending the RF_DEACTIVATE_NTF to the DH, the NFCC SHALL release the Remote NFC Endpoint by sending an NFC-DEP Release Request (RLS_REQ) to the Remote NFC Endpoint as specified in [DIGITAL].

- RF_DEACTIVATE_CMD/RSP/NTF(Sleep_AF Mode)

  Before sending the RF_DEACTIVATE_NTF to the DH, the NFCC SHALL deselect the Remote NFC Endpoint by sending an NFC-DEP Deselect Request (DSL_REQ) to the Remote NFC Endpoint as specified in [DIGITAL].

When using this RF Interface, the following deactivation cases described in Section 5.2 for **RFST_LISTEN_ACTIVE** SHALL NOT be allowed:

- RF_DEACTIVATE_CMD/RSP/NTF(Sleep Mode)

8.4.3.4 Failures during Data Exchange

If an NFC-DEP Timeout Error PROTOCOL EXCEPTION as defined in [DIGITAL] occurs, the NFCC SHALL send the CORE_INTERFACE_ERROR_NTF with the Status of RF_TIMEOUT_ERROR.

If an NFC-DEP Transmission Error PROTOCOL EXCEPTION as defined in [DIGITAL] occurs, the NFCC SHALL send the CORE_INTERFACE_ERROR_NTF with the Status of RF_TRANSMISSION_ERROR.

If an NFC-DEP Protocol Error PROTOCOL EXCEPTION as defined in [DIGITAL] occurs, the NFCC SHALL send the CORE_INTERFACE_ERROR_NTF with the Status of RF_PROTOCOL_ERROR.

8.4.4 Listen-side NFC-DEP RF Interface Management

The Listen-side NFC-DEP RF Interface MAY be used when the device operates as a NFC-DEP Target in Peer Mode and the Remote NFC Endpoint supports NFC-DEP based on NFC-A or NFC-F.

8.4.4.1 Discovery Configuration

The Discovery configuration parameters defined in the following tables are relevant depending on the NFC technology used in Discovery (NFC-A or NFC-F as set by RF Technology and Mode in RF_DISCOVER_CMD). They MAY be changed from defaults by using the CORE_SET_CONFIG_CMD before moving to state **RFST_DISCOVERY**.

- Table 30: Discovery Configuration Parameters for Listen A
  The parameter LA_HIST_BY is not applicable when using this Interface.
- Table 39: Discovery Configuration Parameters for Listen NFC-DEP

When using this Interface, the DH SHALL configure the parameters LA_SEL_INFO (see Table 30) and LF_PROTOCOL_TYPE (see Table 35) to indicate NFC-DEP support.

When using this Interface the parameter LF_CON_BITR_F SHALL be configured by the NFCC to the value 0x06. The NFCC SHALL reject any other configuration for this parameter as long as this Interface is mapped to the NFC-DEP protocol.

8.4.4.2 Discovery and Interface Activation

To enable Listen Mode for NFC-DEP, the DH sends the RF_DISCOVER_CMD to the NFCC containing configurations with RF Technology and Mode values of NFC_A_PASSIVE_LISTEN_MODE and/or NFC_F_PASSIVE_LISTEN_MODE.

When the NFCC is ready to exchange data with the DH (that is, after successful protocol activation), it SHALL send the RF_INTF_ACTIVATED_NTF to the DH to indicate that the NFC-DEP protocol has been activated as follows:

Following the anticollision sequence and when an ATR_REQ has been received from the Remote NFC Endpoint, the NFCC SHALL send an ATR_RES to the Remote NFC Endpoint with LN_ATR_RES_GEN_BYTES configured during Discovery Configuration, but then wait until it receives either a PSL_REQ or a first command. It SHALL then forward the ATR_REQ and the updated data rate and technology information to the DH within the Activation Parameters of the RF_INTF_ACTIVATED_NTF.

For NFC-A and NFC-F the RF_INTF_ACTIVATED_NTF SHALL include the Activation Parameters defined in Table 83.

Table 83: Activation Parameters for NFC-DEP Listen Mode

Parameter	Length	Description
ATR_REQ Command Length	1 Octet	Length of ATR_REQ Command Parameter (n)
ATR_REQ Command	n Octets	All Bytes of the ATR_REQ Command as defined in [DIGITAL] starting from and including Byte 3.

8.4.4.3 Interface Deactivation

Additional rules apply for the following deactivation cases described in Section 5.2 for **RFST_LISTEN_ACTIVE**:

- RF_DEACTIVATE_NTF(Sleep_AF Mode, Endpoint Request)

  After sending a Deselect response (DSL_RES) to the Remote NFC Endpoint, the NFCC SHALL send the DEACTIVATE_NTF.

- RF_DEACTIVATE_NTF(Discovery, Endpoint_Request)

  After sending a Release response (RLS_RES) to the Remote NFC Endpoint, the NFCC SHALL send the DEACTIVATE_NTF.

When using this RF Interface, the following deactivation cases described in Section 5.2 for **RFST_LISTEN_ACTIVE** SHALL NOT be allowed:

- RF_DEACTIVATE_NTF(Sleep Mode, Endpoint Request)
- RF_DEACTIVATE_CMD/RSP/NTF(Sleep Mode)
- RF_DEACTIVATE_CMD/RSP/NTF(Sleep_AF Mode)

8.4.4.4 Failures during Data Exchange

If an NFC-DEP Protocol Error PROTOCOL EXCEPTION [DIGITAL] occurs, the NFCC SHALL send the CORE_INTERFACE_ERROR_NTF with the Status of RF_PROTOCOL_ERROR.

When the DH receives the CORE_INTERFACE_ERROR_NTF, the DH MAY initiate NFC-DEP RF Interface deactivation or perform some other error handling procedure.

9 NFCEE Discovery and Mode Set

9.1 NFCEE ID

The NFCC dynamically assigns IDs for NFCEEs, called NFCEE IDs. The DH gets to know the ID values by performing NFCEE Discovery. NFCEE IDs are valid until an NFCEE is removed from the NFCC.

An ID with a value of zero is referred to in this specification as a DH-NFCEE ID, and SHALL represent the DH-NFCEE.

Table 84: NFCEE IDs

Value	Description
0	DH NFCEE ID, a static ID representing the DH-NFCEE
1 – 254	Dynamically assigned by the NFCC
255	RFU

9.2 NFCEE Discovery

These Control Messages are used to discover whether one or more NFCEEs are connected to the NFCC.

Table 85: Control Messages for NFCEE Discovery

NFCEE_DISCOVER_CMD			
Payload Field(s)	**Length**	**Value/Description**	
Discovery Action	1 Octet	0x00	Disable discovery of NFCEE
		0x01	Enable discovery of NFCEE
		0x02 – 0xFF	RFU

NFCEE_DISCOVER_RSP		
Payload Field(s)	**Length**	**Value/Description**
Status	1 Octet	See Table 94
Number of NFCEEs	1 Octet	0 – 255 Indicates the number of NFCEEs connected to the NFCEE

<table>
<tr><th colspan="5">NFCEE_DISCOVER_NTF</th></tr>
<tr><th>Payload Field(s)</th><th>Length</th><th colspan="3">Value/Description</th></tr>
<tr><td>NFCEE ID</td><td>1 Octet</td><td colspan="3">An NFCEE ID as defined in Table 84. The value of 0x00 (DH-NFCEE ID) SHALL NOT be used.</td></tr>
<tr><td rowspan="4">NFCEE Status</td><td rowspan="4">1 Octet</td><td>0x00</td><td colspan="2">NFCEE connected and enabled</td></tr>
<tr><td>0x01</td><td colspan="2">NFCEE connected and disabled</td></tr>
<tr><td>0x02</td><td colspan="2">NFCEE removed</td></tr>
<tr><td>0x03 – 0xFF</td><td colspan="2">RFU</td></tr>
<tr><td>Number of Protocol Information Entries</td><td>1 Octet</td><td colspan="3">The number of Supported NFCEE Protocol fields to follow (n).</td></tr>
<tr><td>Supported NFCEE Protocols [0..n]</td><td>n Octet</td><td colspan="3">See Table 100.</td></tr>
<tr><td>Number of NFCEE Information TLVs</td><td>1 Octet</td><td colspan="3">Number of NFCEE Information TLV fields to follow (m).</td></tr>
<tr><td rowspan="3">NFCEE Information TLV[0..m]</td><td rowspan="3">x+2 Octets</td><td>Type</td><td>1 Octet</td><td>The Type of the NFCEE Information TLV.
See Table 86.</td></tr>
<tr><td>Length</td><td>1 Octet</td><td>The length of Value (x).</td></tr>
<tr><td>Value</td><td>x Octets</td><td>The Value of the NFCEE Information TLV</td></tr>
</table>

Table 86: TLV Coding for NFCEE Discovery

Type	Length	Value
0x00	n	Hardware / Registration Identification The Hardware/Registration Identification can be used by the DH to validate that any communication between the DH and an NFCEE only occurs with a valid NFCEE. It can be used by the NFC upper layer protocols or a DH application to enable security for communicating with NFCEEs. The value field coding is out of scope of this document and can be defined by each NFCEE manufacturer.
0x01	n	ATR Bytes ATR information contains the transmission parameters such as T = 0 and T = 1, which are supported by the NFCEE. It also carries all the necessary information that MAY be required by the DH such as Data transmission rate, Mask Version number, NFCEE Serial number, NFCEE hardware parameters etc., The format of ATR is defined in [ISO/IEC_7816-3] specification.
0x02	9 – 169	T3T Command Set Interface Supplementary Information The T3T Command Set Interface Supplementary Information MAY be present when one of the Supported NFCEE Protocol parameters contains the value Type 3 Tag Command Set. It SHALL NOT be present otherwise. If the T3T Command Set Interface Supplementary Information is present, it SHALL contain the PMm and Number of Entries field as outlined in Table 87. If the value for Number of Entries is larger than 0, it SHALL be followed by the corresponding number of Entries. Each Entry SHALL consist of the System Code and Idm fields as defined in Table 87.
0x03-0x9F		RFU
0xA0-0xFF		For proprietary use

Table 87: Value Field for T3T Command Set Interface Supplementary Information

Payload Field(s)	Length	Value/Description
PMm	8 Octets	PMm as defined in [T3TOP]
Number of Entries	1 Octets	The number of Entries to follow (n). An entry consists of System Code and Idm parameters. Allowed values SHALL be from 0x00 to 0x10. Other values are RFU.
System Code[n]	2 Octets	System Code as defined in [T3TOP]
Idm[n]	8 Octets	Idm as defined in [T3TOP]

To discover whether one or more NFCEEs are connected to the NFCC, the NFCEE_DISCOVER_CMD is sent by the DH to the NFCC with the Discovery Action of 0x01.

On receipt of the NFCEE_DISCOVER_CMD, the NFCC SHALL respond to the DH with the NFCEE_DISCOVER_RSP with a Status of STATUS_OK and a Number of NFCEEs indicating how many NFCEEs are connected to the NFCC (a value of 0x00 indicates no connected NFCEEs, 0x01 indicates 1 connected NFCEE, etc.).

If there is at least one NFCEE connected to the NFCC, for each connected NFCEE, the NFCC SHALL send a NFCEE_DISCOVER_NTF to the DH indicating the following:

- The unique NFCEE ID assigned by the NFCC to the NFCEE.
- The current NFCEE Status.
- Each NFCEE Protocol supported by the NFCEE
- Zero or more NFCEE Information Records to provide additional information on the NFCEE

NOTE NFCEE Protocols and NFCEE Interfaces have one-to-one mapping, and values for these are defined in the common

NOTE Table 100. If an NFCEE supports certain NFCEE Protocol(s) reported with NFCEE_DISCOVER_NTF, then one of supported protocols may be used for communication between the DH and the NFCEE. A creation of that communication channel is called NFCEE Interface activation.

The assigned NFCEE ID remains valid until the NFCEE is removed or the NFCC is reset with a Configuration Status of 0x01.

Once NFCEE Discovery Process has been enabled, the NFCC SHALL notify the DH of any detectable removals and/or reconnections of NFCEEs through the NFCEE_DISCOVER_NTF.

If a NFCEE_DISCOVER_NTF is sent to notify that a new NFCEE has been connected to the NFCC, the initial state of this NFCEE SHALL be connected and disabled (NFCEE Status value set to 0x01).

If the NFCEE Status is set to 0x02, (NFCEE removed) the Number of Protocol Information Entries SHALL be 0.

If the NFCEE Discovery Process fails, then NFCEE_DISCOVER_RSP SHALL be sent with Status of STATUS_FAILED (see Table 94). In the failure case the Number of NFCEEs SHALL be 0.

If the NFCC detects that an NFCEE is removed, then the Logical Connection to this NFCEE SHALL be closed implicitly and the corresponding NFCEE Interface (Protocol) SHALL be deactivated immediately. In case NFCEE Discovery is enabled a NFCEE_DISCOVER_NTF is sent.

To disable the sending of NFCEE_DISCOVER_NTF, the DH SHALL send the NFCEE_DISCOVER_CMD with Discovery Action is set to 0x00. The NFCC SHALL respond by sending a NFCEE_DISCOVER_RSP with a Status set to STATUS_OK and the Number of NFCEEs parameter SHALL be ignored by DH.

9.2.1 HCI Network Specific Handling

The HCI Network as specified [ETSI_102622] is treated logically by the NCI Specification as one “virtual” NFCEE.

If the NFCC supports the HCI Network, it SHALL return one, and only one, NFCEE_DISCOVER_NTF with a Protocol type of “HCI Access”, even if the HCI Network contains multiple NFCEEs.

The NFCEE ID returned by the NFCC in the NFCEE_DISCOVER_NTF is used by the DH to address the HCI Network.

The following rules apply to the NFCEE_DISCOVER_NTF for an HCI Network:

- If NFCEE Status field has a value of "0x00 – NFCEE connected and enabled", the HCI Network is allowed to perform RF communication (under the conditions given by the RF State Machine).
- If the NFCEE Status field has a value of "0x01 – NFCEE connected and disabled", RF communication is not allowed. This is typically due to a previous NFCEE_MODE_SET_CMD sent to deactivate the HCI Network.

No Hardware/Registration Identification SHALL be provided within NFCEE_DISCOVERY_NTF. The Length of Hardware/Registration Identification SHALL be set to 0. The Hardware/Registration Identification information of the different hosts within the HCI Network can be received by HCP packets according [ETSI_102622].

- An NFCEE_DISCOVER_NTF that contains a Protocol type of "HCI Access" SHALL NOT contain any other additional Protocol.

9.3 NFCEE Enabling and Disabling

These Control Messages are used to enable or disable an NFCEE.

Table 88: Control Messages to Enable and Disable a Connected NFCEE

NFCEE_MODE_SET_CMD			
Payload Field(s)	**Length**	**Value/Description**	
NFCEE ID	1 Octet	NFCEE ID as defined in Table 84. The value of 0x00 (DH-NFCEE ID) SHALL NOT be used.	
NFCEE Mode	1 Octet	0x00	Disable the connected NFCEE
		0x01	Enable the connected NFCEE
		0x02 – 0xFF	RFU

NFCEE_MODE_SET_RSP		
Payload Field(s)	**Length**	**Value/Description**
Status	1 Octet	See Table 94

To enable or disable an NFCEE, the DH sends the NFCEE_MODE_SET_CMD to the NFCC. The NFCEE ID identifies the NFCEE on which the action should occur and the NFCEE Mode identifies whether that NFCEE should be enabled or disabled. On completion of the necessary processing, the NFCC indicates whether the action was successful or unsuccessful by sending the NFCEE_MODE_SET_RSP to the DH. In a failure case the Status SHALL be set to STATUS_FAILED (see Table 94).

The NFCC SHALL NOT route communication to or from a disabled NFCEE. This also includes any communication from a Remote NFC Endpoint or from another NFCEE routed via the NFCC.

The NFCC SHALL only enable or disable a NFCEE when triggered by a NFCEE_MODE_SET_CMD.

If the NFCEE is disabled by the DH with NFCEE_MODE_SET_CMD, then Logical Connection to this NFCEE SHALL be closed implicitly (without sending of a CORE_CONN_CLOSE_CMD) and corresponding NFCEE Interface SHALL be deactivated immediately.

9.3.1 HCI Network Specific Handling

To enable or disable the HCI Network, the DH SHALL send NFCEE_MODE_SET_CMD with the NFCEE ID set to that of the HCI Network.

If the NFCEE Mode field of the NFCEE_MODE_SET_CMD is set to “Enable the connected NFCEE”, the HCI Network is allowed to perform RF communication. Therefore, if they were not already activated, the communication interfaces to the hosts of the HCI Network SHALL be activated. If the HCI Access NFCEE Interface is already enabled while receiving the NFCEE_MODE_SET_CMD, the NFCC SHALL activate all communication interfaces to the hosts in the HCI Network (some of these interfaces may have been deactivated before e.g. for power consumption reasons).

If the NFCEE Mode field of the NFCEE_MODE_SET_CMD is set to “Disable the connected NFCEE”, the HCI Network is not allowed to perform any further RF communication. Therefore it MAY be no longer necessary to keep the communication interfaces to the hosts within the HCI Network activated.

Before disabling the HCI Network, if the HCI Access NFCEE Interface was activated, the DH SHALL deactivate it, as described in Section 10.2.1.

The enabled/disabled state of the HCI Network SHALL be persistent in the NFCC (including while the DH is switched off) until a new NFCEE_MODE_SET_CMD is sent by the DH. This includes across a reset and initialization of the NFCC.

10 NFCEE Interfaces

This section describes the supported NFCEE Interfaces. Unless defined otherwise, all NFCEE Interfaces are optional.

The DH learns which NFCEE Interfaces are supported by an NFCEE during NFCEE Discovery Process, see Section 9.1. The “Supported NFCEE Protocol parameter” field(s) in NFCEE_DISCOVER_NTF identifies the supported NFCEE Protocol(s).

The DH SHALL only initiate NFCEE Interface activation for an NFCEE Protocol that was reported during the NFCEE Discovery Process.

NFCEE Interface activation and deactivation is performed automatically when creating or closing a Logical Connection to an NFCEE, see Section 4.4. There are no specific Control Messages for NFCEE Interface activation or deactivation.

The combination of NFCEE ID and NFCEE Protocol (as reported in the NFCEE_DISCOVER_NTF) used during connection creation uniquely identifies a specific NFCEE Interface to be activated.

In case of an error during NFCEE Interface activation the NFCC SHALL set the Status in the CORE_CONN_CREATE_RSP to NFCEE_INTERFACE_ACTIVATION_FAILED.

There MAY be multiple simultaneous active NFCEE Interfaces, but there SHALL be only one active NFCEE Interface for each NFCEE. This means that for each NFCEE, only one Logical Connection is allowed between the DH and one NFCEE.

An NFCEE Interface SHALL be deactivated when the corresponding Logical Connection is closed. The DH MAY initiate connection closure by using the Conn ID used for the NFCEE Interface, see details in Section 4.4.3.

In case of unrecoverable transmission errors of a message between the NFCC and the NFCEE, the NFCC SHALL send a CORE_INTERFACE_ERROR_NTF with the Status set to NFCEE_TRANSMISSION_ERROR.

10.1 APDU NFCEE Interface

10.1.1 Data Exchange

This communication will be the sending and receiving of APDU command-response pairs – using short messages lengths – as described in Section 12 of [ISO/IEC_7816-3]. That is, $\mathbf{L}_c$ and $\mathbf{L}_e$ are coded on one byte.

The DH MAY send a Data Message to the NFCC according to Section 10.1.1.1. The NFCC will extract and send the Command APDU data contained in the payload of the Data Message(s) to the NFCEE.

When the NFCC receives Response APDU data from the NFCEE, the NFCC SHALL populate the payload of a Data Message with the Response APDU and send the Data Message to the DH according to Section 10.1.1.2.

NCI Segmentation and Reassembly MAY be applied to Data Messages in either direction.

The NFCC is responsible for managing the variances necessary for the communication and receipt of Command and Response APDUs between itself and the NFCEE and this SHALL be transparent to the DH and NCI.

10.1.1.1 Format for Data Messages sent from DH to NFCC

When receiving a Data Message from the DH, the NFCC SHALL send the Data Message, i.e. Command APDU to the NFCEE. If a single Command APDU is split across multiple Data Packets, the NFCC SHALL receive all relevant Data Packets and combine the Command APDU data from all Data Packets prior to sending the Command APDU to the NFCEE.

Figure 21 illustrates the mapping between the Data Packet(s) and the Command APDU to be sent to the NFCEE.

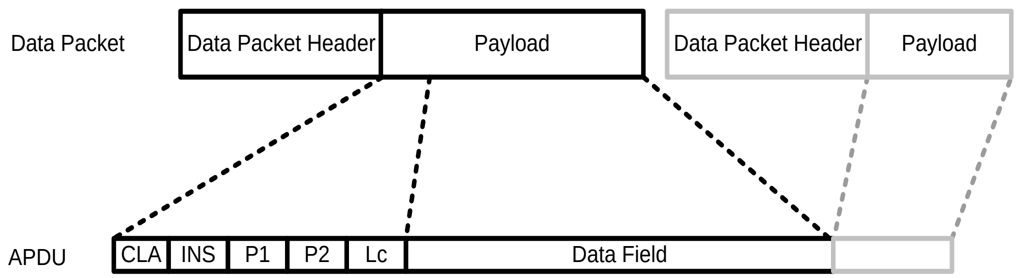

Figure 21: Mapping of Command APDU

The following is the structure of the Command APDU in the Data Message and is dependent on the case of the command.

For Case 1: CLA | INS| P1 | P2

For Case 2: CLA | INS| P1 | P2 | Le

For Case 3: CLA | INS | P1 | P2 | Lc | Data Field

For Case 4: CLA | INS | P1 | P2 | Lc | Data Field | Le

It is the responsibility of the NFCC to correctly map these to the protocol used between the NFCC and the NFCEE. In the case that T=0 is being used as this protocol, it is the responsibility of the NFCC to manage the 0x6Cxx and 0x61xx status words and these status words are not sent from the NFCC to the DH. In these cases the NFCC SHALL transfer the full response.

10.1.1.2 Format for Data Messages sent from NFCC to DH

The NFCC SHALL retrieve the complete APDU Response from the NFCEE and handle it as a Data Message and send it to the DH in one or more Data Packets. If the retrieved APDU Response(s) from an NFCEE does not fit within a single Data Packet, the NFCEE SHALL split the APDU Response across multiple Data Packets.

Figure 22 illustrates the mapping between Response APDU and the Data Packets to be sent to the DH.

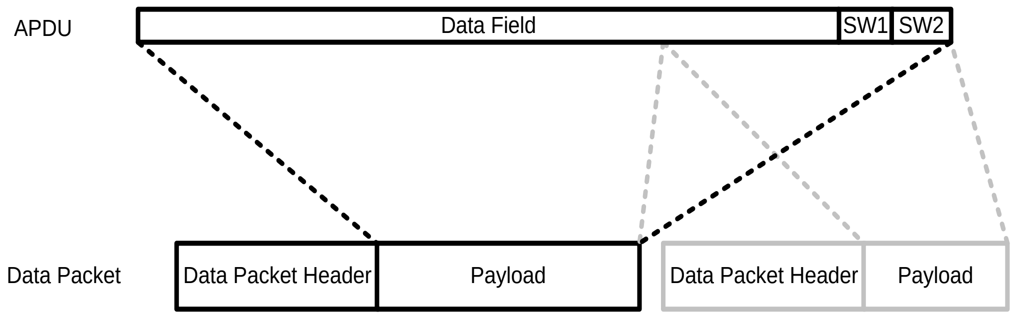

Figure 22: Mapping of Response APDU

10.1.2 Failures during Data Exchange

If the NFCEE fails to respond to an APDU command within the appropriate time, the NFCC SHALL send a CORE_INTERFACE_ERROR_NTF with the Status set to NFCEE_TIMEOUT_ERROR.

If the data received from the DH is not formatted correctly (that is, is less than 4 octets in length or for example, the value provided for Lc in the Command Header does not match the length of the data field), the NFCC SHALL send a CORE_INTERFACE_ERROR_NTF with the Status set to NFCEE_PROTOCOL_ERROR.

10.2 HCI Access NFCEE Interface

The HCI Access Interface is used by the DH to communicate with NFCEEs connected to the NFCC and supporting HCI according to [ETSI_102622].

The NFCC SHALL support the HCI Access Interface if one or more NFCEEs are connected to NFCC using HCI.

NOTE [ETSI_102622] uses different terms to those used in this specification:
- "Terminal" used in [ETSI_102622] corresponds to DH in this specification
- "CLF" used in [ETSI_102622] corresponds to NFCC in this specification
- "Host controller" according [ETSI_102622] is implemented by the NFCC
- UICC is a specific implementation of an NFCEE having also a link with DH via the HCI Access Interface

10.2.1 NFCEE Interface Activation and Deactivation

If the DH has received a NFCEE_DISCOVER_NTF indicating 'HCI Access' (see

Table 100) in the Supported NFCEE Protocol parameter, the DH MAY activate this NFCEE Interface by opening a Logical Connection using the NFCEE ID specified in the NFCEE_DISCOVER_NTF and using 'HCI Access' as the protocol.

When the HCI Access NFCEE Interface is activated and if the HCI Network is already enabled, then the Host Controller SHALL send EVT_HOT_PLUG as described within [ETSI_102622] to the hosts within the HCI Network to make them aware that the Terminal Host is available.

When the HCI Access NFCEE Interface is deactivated by closing the Logical Connection to the HCI Access NFCEE Interface, the hosts within the HCI Network SHALL be made aware by EVT_HOT_PLUG that the Terminal Host has disappeared.

10.2.2 Data Exchange

The payload of the Data Packets sent on the Logical Connection SHALL be valid HCP packets, as defined within [ETSI_102622]. Each Data Packet SHALL contain a single HCP packet. NCI Segmentation and Reassembly SHALL NOT be applied to Data Messages in either direction. The HCI fragmentation mechanism is used if required.

10.3 Type 3 Tag Command Set NFCEE Interface

The Type 3 Tag Command Interface is used by the DH to communicate with NFCEEs connected to the NFCC by exchanging Type 3 Tag Commands and Responses as defined in [T3TOP].

10.3.1 Data Exchange

The payload of the Data Messages sent on the Logical Connection SHALL be Type 3 Tag Commands and Responses as defined in [T3TOP] preceded by a 1 octet length field. Data Messages MAY be segmented into multiple Data Packets when sent over the NCI.

The length field value SHALL equal the length of the Type 3 Tag Command or Response plus one.

Type 3 Tag Commands SHALL only be sent in the direction of DH to NFCC. Type 3 Tag Responses SHALL only be sent in the direction of NFCC to DH.

Figure 23 illustrate the mapping between the Data Packet Format and the Type 3 Tag Command Set Format when no segmentation is applied.

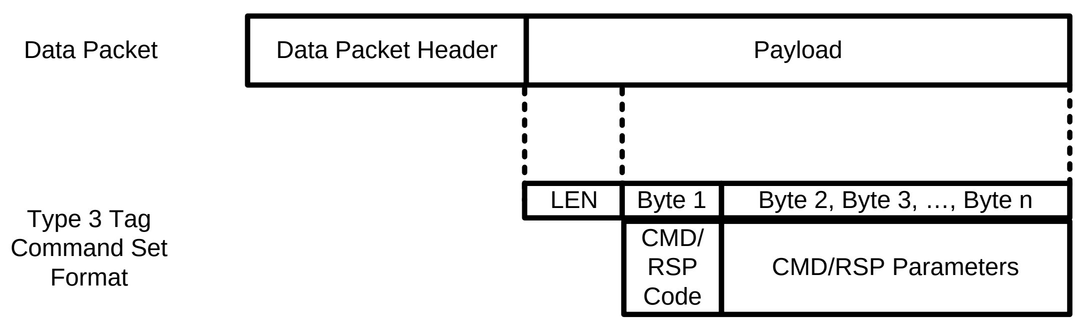

Figure 23: Data Message Format for Type 3 Tag Command Set Interface

10.3.1.1 Polling Command and Response

If a SENSF_REQ is sent by the DH to the NFCEE, it SHALL be coded as defined in the Section 6.5 of [DIGITAL] and SHALL have a TSN value of 0x00.

The DH SHOULD choose a larger value for ΔTPoll as defined in [DIGITAL] to accommodate a potential longer transmission time of the SENS_RES from the NFCEE to the DH compared to the RF communication.

10.4 Transparent NFCEE Interface

The Transparent Interface is used by the DH to communicate with NFCEEs connected to the NFCC by exchanging data which is not understood by the NFCC, but just passed without modification.

10.4.1 Data Exchange

The DH MAY send Data Messages on such a connection to the NFCC for transmission to the NFCEE. The NFCC SHALL extract and forward the payload of the Data Messages – in the appropriate message format – directly to the NFCEE without any modification.

When the NFCC receives a message from the NFCEE, the NFCC SHALL extract the data from the message, populate it to the payload of a Data Message and send the Data Message to the DH on the relevant connection. The mechanism used to transmit such data between the NFCC and the NFCEE is implementation specific. The NFCC is responsible for managing the variances necessary for the communication between itself and the NFCEE and this SHALL be transparent to the DH and NCI.

NCI Segmentation and Reassembly MAY be applied to Data Messages in either direction.

11 Transport Mappings

The NCI Core design is intended to be independent of any specific underlying transport layer or its speed.

The following are requirements for any underlying Transport Mapping:

- Transport Mappings SHALL provide means to transport Data and Control Packets in both directions between the DH and NFCC.
- Transport Mappings SHALL provide a reliable data transfer.
- Transport Mappings MAY include flow control mechanisms. However it is recommended to rely on the flow control built into the NCI protocol.
- Transport Mappings providing framing SHALL NOT forward Packets with a size smaller than 3 bytes to the NCI Core.

The following sub-sections describe Transport Mappings for NCI.

It is not mandated to use any of the Transport Mapping defined on the following sub-sections. The device implementation MAY use a proprietary Transport Mapping that fulfills the above requirements (even for transport layers where this specification contains a mapping).

11.1 UART Transport

NCI Frames SHALL be transmitted over the UART between an DH and an NFCC with no additional framing. Since there is no additional framing, the UART transport cannot introduce any errors; otherwise the NCI messages stream may be unrecoverable. So the UART connection SHALL be of a sufficiently high quality as to provide a reliable data transfer at the configured baud rate.

The NCI UART Transport SHOULD use the following settings for RS232:

- 8 data bits
- 1 stop bit
- No parity
- Automatic (i.e. h/w based) RTS/CTS Flow control

The baud rate is manufacturer-specific.

Flow control with RTS/CTS is used to prevent temporary UART buffer overrun, and is not intended to be used other than temporarily. NCI has its own credit-based logical flow control mechanism which is better suited to the different types of NCI data that flow across the transport.

If CTS is 1, then the DH/NFCC is allowed to send. If CTS is 0, then the DH/NFCC is not allowed to send.

The RS232 signals SHOULD be connected in a null-modem fashion; i.e. the local TXD SHOULD be connected to the remote RXD and the local RTS SHOULD be connected to the remote CTS and vice versa.

11.2 I2C Transport

NCI Frames SHALL be transmitted over the I^2C [I2C] between a DH and an NFCC with no additional framing. Since there is no additional framing, the I^2C transport cannot introduce any errors; otherwise the NCI messages stream may be unrecoverable. So the I^2C connection SHALL be of a sufficiently high quality as to provide a reliable data transfer at the configured transfer rate.

The DH SHOULD operate as a bus master and MAY also be able operate as a bus slave if addressed by another bus master. The NFCC MAY operate as either bus master or slave. In the latter case 'out-of-band' means SHOULD be provided in order for the NFCC to request the DH to initiate a data transfer as a bus master. However, if 'out-of-band' method is not available, the DH SHALL poll frequently the NFCC. Polling frequency is implementation specific.

The I2C transport SHOULD support standard (up to 100kbps) and Fast-Speed mode (up to 400kbps) and MAY support Fast-Mode Plus (up to 1 Mbit/s) and High Speed mode (up to 3.4Mbit/s).

Additionally, 10-bit addressing mode and clock stretching MAY be supported on the I^2C transport.

11.3 Half Duplex SPI Transport

11.3.1 Physical

NCI control and data messages are transmitted over SPI. The DH SHALL be the master of the communication. The communication SHALL use standard SPI signal lines i.e. SPI_CSN, SPI_CLK, SPI_MOSI and SPI_MISO. In addition to attain flow control there SHALL be an additional SPI_INT signal line which SHALL be driven by the slave. The clock rate negotiation is manufacturer specific and is out of scope of this document. The data sent through the transport is assumed to be in Little Endian format.

11.3.1.1 SPI Modes

SPI can be operated in one of the four modes described below. The mode selection procedure is manufacturer specific and is out of scope of this document.

Table 89: SPI modes

SPI mode	CPOL	CPHA
0	0	0
1	0	1
2	1	0
3	1	1

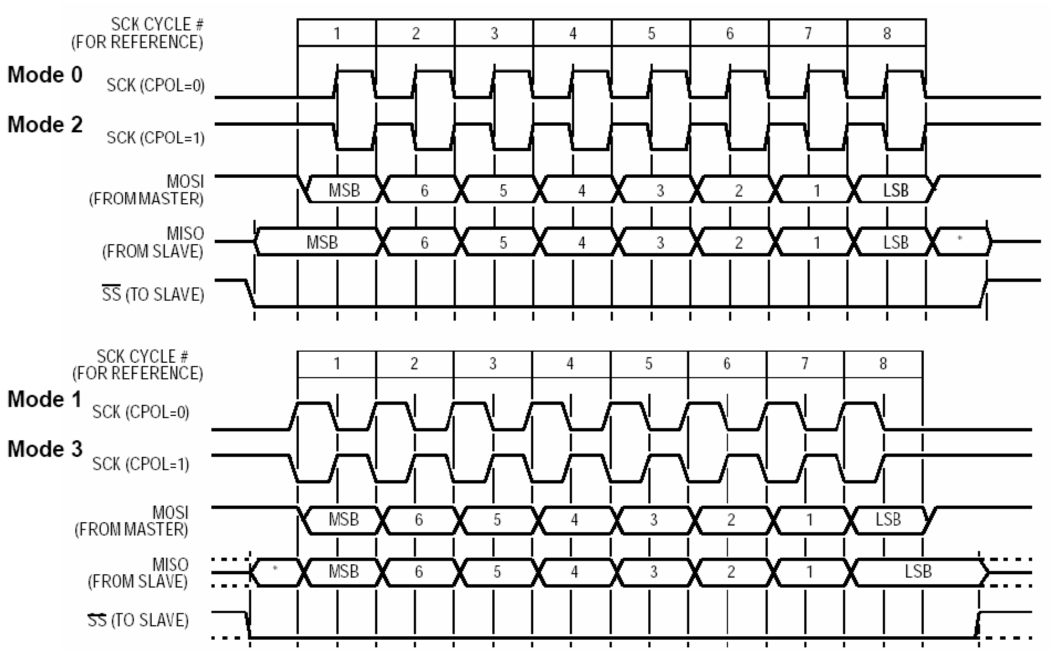

Figure 24: SPI Operation

11.3.2 Data Transfer

This proposal reduces the full-duplex nature of the Generic SPI transport to a half-duplex interface. Packet framing and flow-control mechanisms are gained in exchange for the loss of bandwidth. These are more important as the intended clock rate far exceeds the NCI bandwidth requirement.

The transport mechanism has the capability to send a maximum sized NCI packet in a single SPI frame.

The following Data Transfer sections define two modes of operation: acknowledged and unacknowledged. Which mode is used is determined by the Second Octet. In the acknowledged mode a CRC is following the Payload and the receiver sends an positive acknowledge for correctly received payloads or a negative acknowledge for erroneous payloads.

11.3.2.1 Data Transfer from DH to NFCC

The master SHALL use the “DirectWrite” command to send data to a slave. The master can initiate a data transfer in the following fashion:

- Master asserts SPI_CSN
- Slave asserts SPI_INT
- Master sends 2-byte DirectWrite header, followed by 2-byte SPI payload length parameter
- Master sends SPI payload
- Slave deasserts SPI_INT
- Master deasserts SPI_CSN

SPI_CSN may be toggled (deasserted and asserted) in every 8-bit transfer.

If the Second Octet is equal to 0x00, the DirectWrite header is sent as shown in the table below.

Table 90: SPI Header Coding (DH to NFCC) without CRC

	First Octet	Second Octet	Third Octet	Fourth Octet
Master	0x01	0x00	MSB (Payload Length	LSB (Payload Length)
Slave	0x00	0x00	0x00	0x00

The Payload length is the length of the whole NCI packet inclusive of the NCI header. Since the NCI packet can be as long as 258 octets we need a 2 octet payload length field.

Figure 25 illustrates the above procedure to transfer data from the DH to the NFCC.

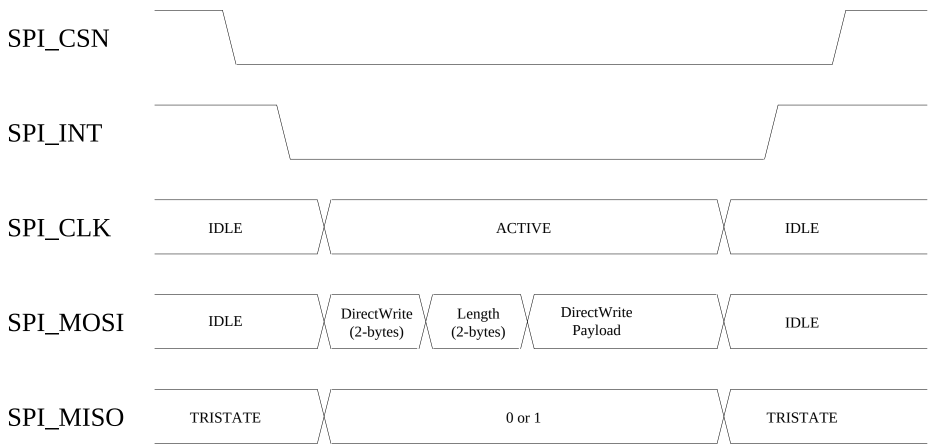

Figure 25: SPI Data Transfer from the DH to the NFCC without CRC

If the second octet is equal to 0x01, the DirectWrite header is sent as shown in the table below and 2-octet CRC SHALL follow the Payload:

CRC SHALL be a CRC-16-CCITT using the polynomial x16 + x12 + x5 + 1, calculated for all octets of Header and Payload. The initial value SHALL be FFFFh. The first CRC octet transmitted SHALL be the MSB.

The master SHALL send ACK if the CRC is correct when receiving data from the slave.

The master SHALL send NAK if the CRC is not correct when receiving data from the slave.

Table 91: SPI Header Coding (DH to NFCC) with CRC

	First Octet	Second Octet	Third Octet	Fourth Octet
Master	0x01	0x01	b8: NAK if set to 1b b7: ACK if set to 1b b6-b1: MSB (Payload Length)	LSB (Payload Length)
Slave	0x00	0x00	0x00	0x00

Figure 26 illustrates the procedure to transfer data from the DH to the NFCC when the Second Octet is equal to 0x01.

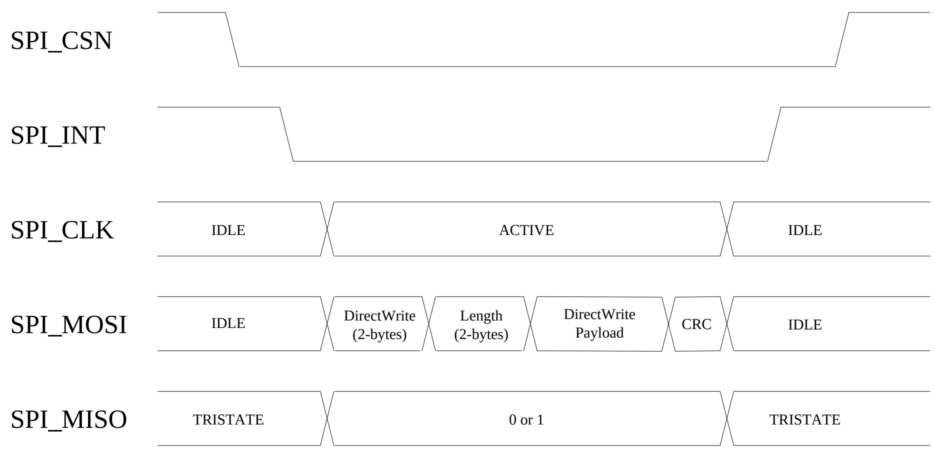

Figure 26: SPI Data Transfer from the DH to the NFCC with CRC

If the slave does not assert SPI_INT in a timely fashion, the master is allowed to deassert SPI_CSN and use the SPI bus to talk to a different peripheral. This can happen if the slave is busy processing its FIFOs.

11.3.2.2 Data Transfer from NFCC to DH

The “DirectRead” command is used to transfer data from slave to the master. Data can be transferred from the slave to the master in the following fashion:

- Slave asserts SPI_INT
- Master asserts SPI_CSN
- Master send 2-octet SPI header
- Slave sends 2-octet SPI payload length
- Slave sends SPI payload
- Master deasserts SPI_CSN

Slave may deasserts SPI_INT at any time after step 2.

Slave SHALL deasserts SPI_INT before step 6.

SPI_CSN may be toggled (deasserted and asserted) in every 8-bit transfer.

If the Second Octet is equal to 0x00, the DirectRead header is sent as shown in the table below.

Table 92: SPI Header Coding (NFCC to DH) without CRC

	First Octet	Second Octet	Third Octet	Fourth Octet
Master	0x02	0x00	0x00	0x00
Slave	0x00	0x00	MSB (Payload Length)	LSB (Payload Length)

The payload length is defined in the same manner as above. Figure 27 bellow illustrates the procedure to transfer data from NFCC to the DH.

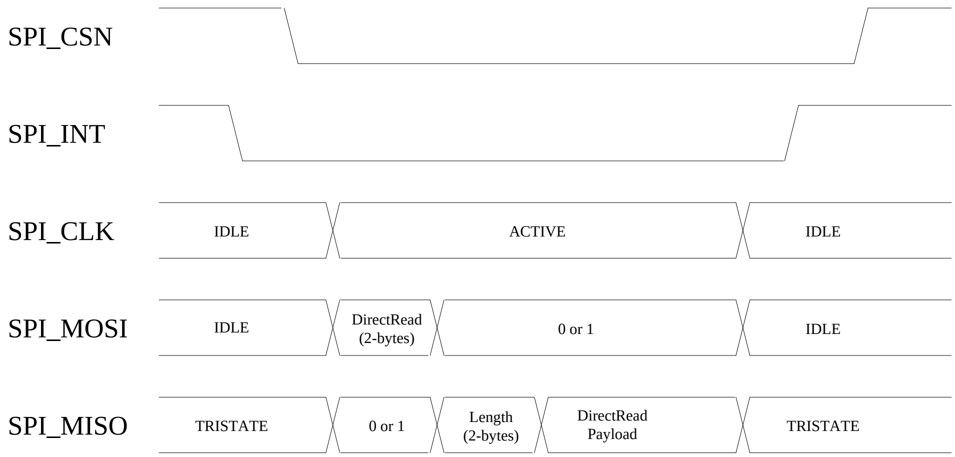

Figure 27: SPI Data Transfer from the NFCC to the DH without CRC

If the second octet is equal to 0x01, the DirectRead header is sent as shown in the table below and 2-octet CRC SHALL follow the Payload:

The CRC SHALL be the same as defined in Section 11.3.2.1.

The slave SHALL send ACK if the CRC is correct when receiving data from the master.

The slave SHALL send NAK if the CRC is not correct when receiving data from the master.

Table 93: SPI Header Coding (NFCC to DH) with CRC

	First Octet	Second Octet	Third Octet	Fourth Octet
Master	0x02	0x01	0x00	0x00
Slave	0x00	0x00	b8: NAK if set to 1b b7: ACK if set to 1b b6-b1: MSB (Payload Length	LSB (Payload Length)

Figure 28 illustrates the procedure to transfer data from NFCC to the DH when Second Octet is equal to 0x01.

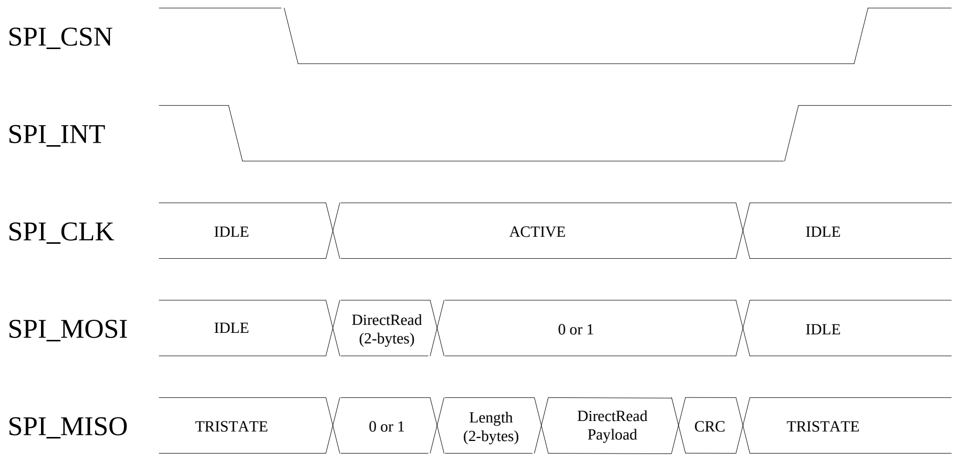

Figure 28: SPI Data Transfer from the NFCC to the DH with CRC

If the master does not assert SPI_CSN in a timely fashion, the slave is allowed to deassert SPI_INT. This can happen if the master is busy processing its FIFOs.

11.3.2.3 Race Condition

Since this is a half-duplex interface there is a possibility that both master (DH) and slave (NFCC) would want to send data at the same time. In such a race condition the master would deassert the SPI_CSN at the same time as the slave would deassert the SPI_INT. In such a situation the 2-octet header sent by the master determines the transaction type. A consequence of this is that the slave SHALL only deassert SPI_INT when it is capable of receiving a maximum-sized Direct-Write command. The two figures below illustrate such a scenario.

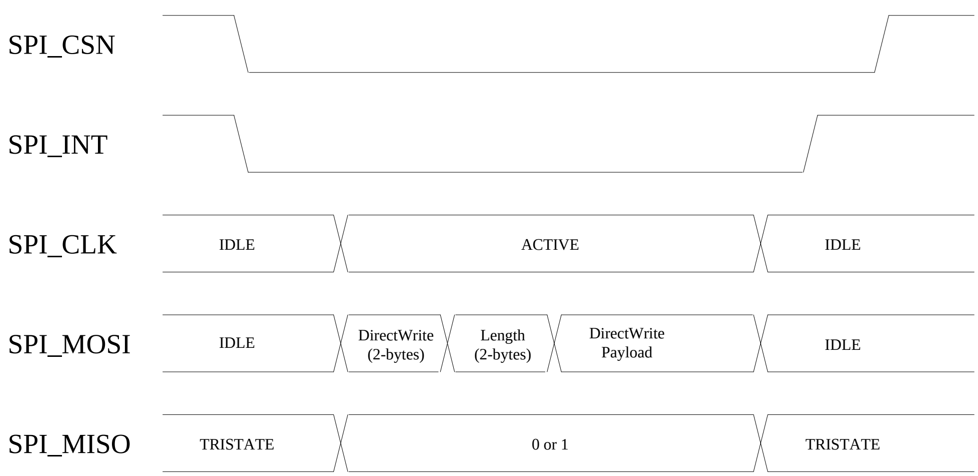

Figure 29: SPI Race Condition 1

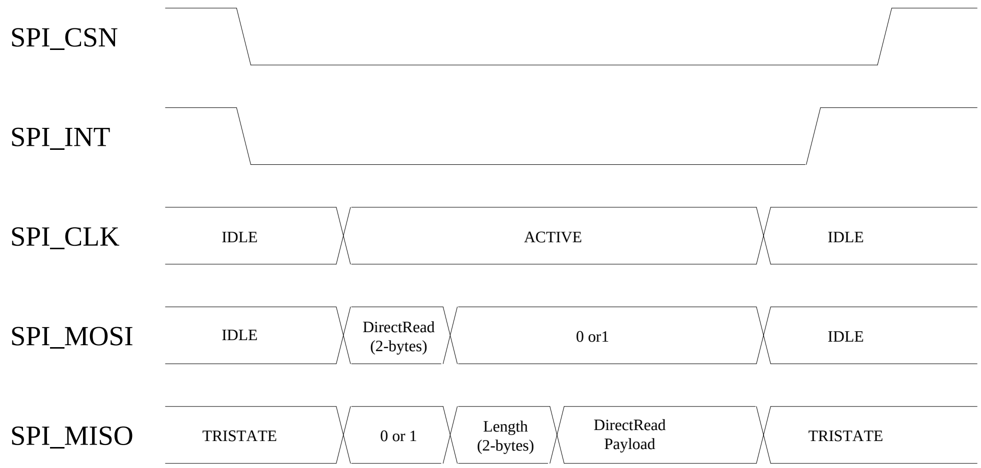

Figure 30: SPI Race Condition 2

12 Testing

The Commands, Responses and Notifications in this section provide mechanisms to facilitate testing.

12.1 Local Loopback Mode

The local loopback mode is used to verify that the NFCC can receive and loopback data.

For testing purposes the DH can enable the loopback mode by creating a Logical Connection with Destination Type value 0x01.

After the successful creation of the Logical Connection the DH MAY send Data Messages over that connection, and the NFCC SHALL loopback the same Data Messages to the DH. Flow control and segmentation & reassembly mechanisms apply also for the loopback connection.

NOTE Data Messages can be segmented differently when returning them to the DH.

A. Exhibit A

Exhibit A is left blank intentionally.

B. Common Tables

Table 94: Status Codes

Status Code	Description
Generic Status Codes	
0x00	STATUS_OK
0x01	STATUS_REJECTED
0x02	STATUS_RF_FRAME_CORRUPTED
0x03	STATUS_FAILED
0x04	STATUS_NOT_INITIALIZED
0x05	STATUS_SYNTAX_ERROR
0x06	STATUS_SEMANTIC_ERROR
0x07 – 0x08	RFU
0x09	STATUS_INVALID_PARAM
0x0A	STATUS_MESSAGE_SIZE_EXCEEDED
0x0B-0x9F	RFU
RF Discovery Specific Status Codes	
0xA0	DISCOVERY_ALREADY_STARTED
0xA1	DISCOVERY_TARGET_ACTIVATION_FAILED
0xA2	DISCOVERY_TEAR_DOWN
0xA3-0xAF	RFU
RF Interface Specific Status Codes	
0xB0	RF_TRANSMISSION_ERROR
0xB1	RF_PROTOCOL_ERROR
0xB2	RF_TIMEOUT_ERROR
0xB3-0xBF	RFU
NFCEE Interface Specific Status Codes	
0xC0	NFCEE_INTERFACE_ACTIVATION_FAILED
0xC1	NFCEE_TRANSMISSION_ERROR
0xC2	NFCEE_PROTOCOL_ERROR
0xC3	NFCEE_TIMEOUT_ERROR
0xC4-0xDF	RFU
Proprietary Status Codes	
0xE0-0xFF	For proprietary use

Table 95: RF Technologies

RF Technology value	Definition
0x00	NFC_RF_TECHNOLOGY_A
0x01	NFC_RF_TECHNOLOGY_B
0x02	NFC_RF_TECHNOLOGY_F
0x03	NFC_RF_TECHNOLOGY_15693
0x04 – 0x7F	RFU
0x80-0xFE	For proprietary use
0xFF	RFU

NOTE The RF Technology with value 0x03: NFC_RF_TECHNOLOGY_15693 is not covered by [DIGITAL].

Table 96: RF Technology and Mode

Value	Description
0x00	NFC_A_PASSIVE_POLL_MODE
0x01	NFC_B_PASSIVE_POLL_MODE
0x02	NFC_F_PASSIVE_POLL_MODE
0x03	NFC_A_ACTIVE_POLL_MODE (RFU)
0x04	RFU
0x05	NFC_F_ACTIVE_POLL_MODE (RFU)
0x06	NFC_15693_PASSIVE_POLL_MODE (RFU)
0x07 – 0x6F	RFU
0x70 – 0x7F	Reserved for Proprietary Technologies in Poll Mode
0x80	NFC_A_PASSIVE_LISTEN_MODE
0x81	NFC_B_PASSIVE_LISTEN_MODE
0x82	NFC_F_PASSIVE_LISTEN_MODE
0x83	NFC_A_ACTIVE_LISTEN_MODE (RFU)
0x84	RFU
0x85	NFC_F_ACTIVE_LISTEN_MODE (RFU)
0x86	NFC_15693_PASSIVE_LISTEN_MODE (RFU)
0x87 – 0xEF	RFU
0xF0 – 0xFF	Reserved for Proprietary Technologies in Listen Mode

NOTE Active communication mode and ISO/IEC 15693 is currently not supported by the NFC Forum.

Table 97: Bit Rates

Bit Rate value	Definition
0x00	NFC_BIT_RATE_106: 106 Kbit/s
0x01	NFC_BIT_RATE_212: 212 Kbit/s
0x02	NFC_BIT_RATE_424: 424 Kbit/s
0x03	NFC_BIT_RATE_848: 848 Kbit/s
0x04	NFC_BIT_RATE_1695: 1695 Kbit/s
0x05	NFC_BIT_RATE_3390: 3390 Kbit/s
0x06	NFC_BIT_RATE_6780: 6780 Kbit/s
0x07 – 0x7F	RFU
0x80-0xFE	For proprietary use
0xFF	RFU

The allowed values for each technology SHALL be as defined in [DIGITAL] and [ACTIVITY].

Table 98: RF Protocols

RF Protocol value	Definition
0x00	PROTOCOL_UNDETERMINED
0x01	PROTOCOL_T1T
0x02	PROTOCOL_T2T
0x03	PROTOCOL_T3T
0x04	PROTOCOL_ISO_DEP
0x05	PROTOCOL_NFC_DEP
0x06 – 0x7F	RFU
0x80-0xFE	For proprietary use
0xFF	RFU

NOTE Type 4 Tag Platform is based on the ISO-DEP Protocol.

Table 99: RF Interfaces

RF Interface value	Definition
0x00	NFCEE Direct RF Interface
0x01	Frame RF Interface
0x02	ISO-DEP RF Interface
0x03	NFC-DEP RF Interface
0x04 – 0x7F	RFU
0x80-0xFE	For proprietary use
0xFF	RFU

Table 100: NFCEE Protocols / Interfaces

NFCEE Interface / Protocol value	Definition
0x00	APDU
0x01	HCI Access
0x02	Type 3 Tag Command Set
0x03	Transparent
0x04-0x7F	RFU
0x80-0xFE	For proprietary use
0xFF	RFU

Table 101: Configuration Parameter Tags

Parameter Name	Tag
Common Discovery Parameters	
TOTAL_DURATION	0x00
CON_DEVICES_LIMIT	0x01
RFU	0x02-0x07
Poll Mode – NFC-A Discovery Parameters	
PA_BAIL_OUT	0x08
RFU	0x09-0x0F
Poll Mode – NFC-B Discovery Parameters	
PB_AFI	0x10
PB_BAIL_OUT	0x11
PB_ATTRIB_PARAM1	0x12
PB_SENSB_REQ_PARAM	0x13
RFU	0x14-0x17
Poll Mode – NFC-F Discovery Parameters	
PF_BIT_RATE	0x18
PF_RC_CODE	0x19
RFU	0x1A-0x1F
Poll Mode – ISO-DEP Discovery Parameters	
PB_H_INFO	0x20
PI_BIT_RATE	0x21
PA_ADV_FEAT	0x22
RFU	0x23-0x27
Poll Mode – NFC-DEP Discovery Parameters	
PN_NFC_DEP_SPEED	0x28
PN_ATR_REQ_GEN_BYTES	0x29
PN_ATR_REQ_CONFIG	0x2A
RFU	0x2B-0x2F
Listen Mode – NFC-A Discovery Parameters	
LA_BIT_FRAME_SDD	0x30
LA_PLATFORM_CONFIG	0x31
LA_SEL_INFO	0x32

Parameter Name	Tag
LA_NFCID1	0x33
RFU	0x34-0x37
Listen Mode – NFC-B Discovery Parameters	
LB_SENSB_INFO	0x38
LB_NFCID0	0x39
LB_APPLICATION_DATA	0x3A
LB_SFGI	0x3B
LB_ADC_FO	0x3C
RFU	0x3D – 0x3F
Listen Mode – NFC-F Discovery Parameters	
LF_T3T_IDENTIFIERS_1	0x40
LF_T3T_IDENTIFIERS_2	0x41
LF_T3T_IDENTIFIERS_3	0x42
LF_T3T_IDENTIFIERS_4	0x43
LF_T3T_IDENTIFIERS_5	0x44
LF_T3T_IDENTIFIERS_6	0x45
LF_T3T_IDENTIFIERS_7	0x46
LF_T3T_IDENTIFIERS_8	0x47
LF_T3T_IDENTIFIERS_9	0x48
LF_T3T_IDENTIFIERS_10	0x49
LF_T3T_IDENTIFIERS_11	0x4A
LF_T3T_IDENTIFIERS_12	0x4B
LF_T3T_IDENTIFIERS_13	0x4C
LF_T3T_IDENTIFIERS_14	0x4D
LF_T3T_IDENTIFIERS_15	0x4E
LF_T3T_IDENTIFIERS_16	0x4F
LF_PROTOCOL_TYPE	0x50
LF_T3T_PMM	0x51
LF_T3T_MAX	0x52
LF_T3T_FLAGS	0x53
LF_CON_BITR_F	0x54
LF_ADV_FEAT	0x55

Parameter Name	Tag
RFU	0x56-0x57
Listen Mode – ISO-DEP Discovery Parameters	
LI_FWI	0x58
LA_HIST_BY	0x59
LB_H_INFO_RESP	0x5A
LI_BIT_RATE	0x5B
RFU	0x5C-0x5F
Listen Mode – NFC-DEP Discovery Parameters	
LN_WT	0x60
LN_ATR_RES_GEN_BYTES	0x61
LN_ATR_RES_CONFIG	0x62
RFU	0x63-0x7F
Other Parameters	
RF_FIELD_INFO	0x80
RF_NFCEE_ACTION	0x81
NFCDEP_OP	0x82
RFU	0x83-0x9F
Reserved for Proprietary Use	
Reserved	0xA0-0xFE
Reserved for Extension	
RFU	0xFF

Table 102: GID and OID Definitions

GID	OID	Message Name
NCI Core 0000b	0000b	CORE_RESET_CMD CORE_RESET_RSP CORE_RESET_NTF
	0001b	CORE_INIT_CMD CORE_INIT_RSP
	0010b	CORE_SET_CONFIG_CMD CORE_SET_CONFIG_RSP
	0011b	CORE_GET_CONFIG_CMD CORE_GET_CONFIG_RSP
	0100b	CORE_CONN_CREATE_CMD CORE_CONN_CREATE_RSP
	0101b	CORE_CONN_CLOSE_CMD CORE_CONN_CLOSE_RSP
	0110b	CORE_CONN_CREDITS_NTF
	0111b	CORE_GENERIC_ERROR_NTF
	1000b	CORE_INTERFACE_ERROR_NTF
	1001b- 1111b	RFU
RF Management 0001b	0000b	RF_DISCOVER_MAP_CMD RF_DISCOVER_MAP_RSP
	0001b	RF_SET_LISTEN_MODE_ROUTING_CMD RF_SET_LISTEN_MODE_ROUTING_RSP
	0010b	RF_GET_LISTEN_MODE_ROUTING_CMD RF_GET_LISTEN_MODE_ROUTING_RSP RF_GET_LISTEN_MODE_ROUTING_NTF
	0011b	RF_DISCOVER_CMD RF_DISCOVER_RSP RF_DISCOVER_NTF
	0100b	RF_DISCOVER_SELECT_CMD RF_DISCOVER_SELECT_RSP
	0101b	RF_INTF_ACTIVATED_NTF
	0110b	RF_DEACTIVATE_CMD RF_DEACTIVATE_RSP RF_DEACTIVATE_NTF
	0111b	RF_FIELD_INFO_NTF

GID	OID	Message Name
	1000b	RF_T3T_POLLING_CMD RF_T3T_POLLING_RSP RF_T3T_POLLING_NTF
	1001b	RF_NFCEE_ACTION_NTF
	1010b	RF_NFCEE_DISCOVERY_REQ_NTF
	1011b	RF_PARAMETER_UPDATE_CMD RF_PARAMETER_UPDATE_RSP
	1100b-1111b	RFU
NFCEE Management 0010b	0000b	NFCEE_DISCOVER_CMD NFCEE_DISCOVER_RSP NFCEE_DISCOVER_NTF
	0001b	NFCEE_MODE_SET_CMD NFCEE_MODE_SET_RSP
	0010b-1111b	RFU
Proprietary 1111		

NOTE An implementation using proprietary Notifications should take into account that any DH not supporting those proprietary extensions will silently discard the Notifications.

C. Revision History

The following table outlines the revision history of NFC Controller Interface (NCI) Specification.

Table 103: Revision History

Document Name	**Revision and Release Date**	**Status**	**Change Notice**	**Supersedes**
NFC Controller Interface (NCI) Specification	Version 1, November 2012	Final		None