ETS1 TS 102 622 V12.0.0 (2013-10)

Smart Cards;
UICC - Contactless Front-end (CLF) Interface;
Host Controller Interface (HCI)
(Release 12)
9.1 Overview ................................................................. 32
9.2 Void........................................................................ 34
9.3 Gates........................................................................ 34
9.3.1 Void ..................................................................... 34
9.3.2 Identity management gate................................. 34
9.3.3 Card RF gates .................................................... 34
9.3.3.1 Overview.......................................................... 35
9.3.3.2 Commands ......................................................... 35
9.3.3.3 Events............................................................... 35
9.3.3.3.1 EVT_SEND_DATA...................................................... 35
9.3.3.4 Registry ............................................................. 35
9.3.3.4.1 RF technology type A ........................................ 35
9.3.3.4.2 RF technology type B ........................................ 37
9.3.3.4.3 RF technology type B'................................. 38
9.3.3.4.4 RF technology Type F (ISO/IEC 18092 212 kbps/424 kbps card emulation only) ........ 38
9.3.3.4 Card application gates................................. 39
9.3.3.4.1 Overview.......................................................... 39
9.3.3.4.2 Commands ......................................................... 39
9.3.3.4.3 Events............................................................... 39
9.3.3.4.3.1 EVT_FIELD_ON.................................................. 40
9.3.3.4.3.2 EVT_CARD_DEACTIVATED....................... 40
9.3.3.4.3.3 EVT_CARD_ACTIVATED.............................. 40
9.3.3.4.3.4 EVT_FIELD_OFF............................................. 40
9.3.3.4.3.5 EVT_SEND_DATA...................................................... 40
9.3.3.4.4 Registry ............................................................. 40
9.4 Procedures ................................................................ 41
9.4.1 Use of contactless card application ......................... 41
9.4.2 Non ISO/IEC 14443-4 type A................................. 42
9.4.3 Type B' RF technology ........................................ 42
9.4.4 Type F RF technology ........................................ 42
9.4.5 Update RF technology settings............................ 43
9.4.6 Identity check....................................................... 43
10 Contactless reader ................................................................. 44
10.1 Overview .................................................................. 44
10.2 Reader RF gates ............................................................. 45
10.2.1 Overview ............................................................. 45
10.2.2 Command ............................................................ 45
10.2.2.1 WR_XCHG_DATA.................................................. 45
10.2.3 Registries .............................................................. 46
10.2.3.1 Type A reader RF gate ....................................... 46
10.2.3.2 Type B reader RF gate ....................................... 48
10.2.4 Events ................................................................. 48
10.2.4.1 EVT_READER_REQUESTED....................... 49
10.2.4.2 EVT_END_OPERATION ............................... 49
10.2.5 Responses ......................................................... 49
10.3 Reader application gates................................. 49
10.3.1 Overview ............................................................. 49
10.3.2 Command ............................................................ 49
10.3.3 Registry .............................................................. 49
10.3.4 Events ................................................................. 49
10.3.4.1 EVT_TARGET_DISCOVERED....................... 50
10.3.4.2 EVT_READER_STATUS ........................................................................................................ 50
10.4 Procedures ....................................................................................................................................... 51
10.4.1 Use of contactless reader application ......................................................................................... 51
10.4.2 Contactless reader not available ................................................................................................. 52
10.4.3 Error management ...................................................................................................................... 53

11 Connectivity ....................................................................................................................................... 53
11.1 Overview ......................................................................................................................................... 53
11.2 Connectivity gate ............................................................................................................................. 54
11.2.1 Commands .................................................................................................................................. 54
11.2.1.1 PRO_HOST_REQUEST ........................................................................................................... 54
11.2.2 Events ....................................................................................................................................... 54
11.2.2.1 EVT_CONNECTIVITY ........................................................................................................... 54
11.2.2.2 Void ....................................................................................................................................... 55
11.2.2.3 EVT_OPERATION_ENDED ..................................................................................................... 55
11.2.2.4 EVT_TRANSACTION ............................................................................................................... 55
11.2.3 Registry ...................................................................................................................................... 55

11.3 Connectivity application gate ........................................................................................................... 55
11.3.1 Commands .................................................................................................................................. 55
11.3.2 Events ....................................................................................................................................... 55
11.3.2.1 EVT_STANDBY ....................................................................................................................... 55
11.3.3 Registry ...................................................................................................................................... 55

11.4 Procedures ....................................................................................................................................... 56
11.4.1 Use of connectivity gate ............................................................................................................... 56

Annex A (normative): State diagram for contactless reader mode ......................................................... 57
Annex B (informative): Change history .................................................................................................. 58
History ..................................................................................................................................................... 60
Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for ETSI members and non-members, and can be found in ETSI SR 000 314: "Intellectual Property Rights (IPRs): Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards", which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://ipr.etsi.org).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This Technical Specification (TS) has been produced by ETSI Technical Committee Smart Card Platform (SCP).

The contents of the present document are subject to continuing work within TC SCP and may change following formal TC SCP approval. If TC SCP modifies the contents of the present document, it will then be republished by ETSI with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

x  the first digit:
0  early working draft;
1  presented to TC SCP for information;
2  presented to TC SCP for approval;
3  or greater indicates TC SCP approved document under change control.

Y  the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.

z  the third digit is incremented when editorial only changes have been incorporated in the document.
1 Scope

The present document specifies a logical interface that enables contactless applications hosted on the UICC. The present document covers the configuration where the one host is embedded in the UICC which is connected to the host controller embedded in the CLF.

The interface is specified in two parts:

- the HCI core as described in the first part of the present document (clauses 4 to 8), specifies an application independent logical interface; and
- the contactless platform as described in the second part of the present document (clauses 9 to 11) specifies an implementation of the HCI core for contactless applications using the UICC and the CLF.

Lower layer protocols that support the HCI like the SWP specified in TS 102 613 [2] are out of the scope of the present document.

2 References

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the reference document (including any amendments) applies.

- In the case of a reference to a TC SCP document, a non specific reference implicitly refers to the latest version of that document in the same Release as the present document.

Referenced documents which are not found to be publicly available in the expected location might be found at http://docbox.etsi.org/Reference.

NOTE: While any hyperlinks included in this clause were valid at the time of publication ETSI cannot guarantee their long term validity.

2.1 Normative references

The following referenced documents are necessary for the application of the present document.


[2] ETSI TS 102 613: "Smart Cards; UICC - Contactless Front-end (CLF) Interface; Part 1: Physical and data link layer characteristics".

[3] ETSI TS 102 223: "Smart Cards; Card Application Toolkit (CAT)".


[9] ETSI TS 101 220: "Smart Cards; ETSI numbering system for telecommunication application providers".
2.2 Informative references

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.


3 Definitions, symbols and abbreviations

3.1 Definitions

For the purposes of the present document, the following terms and definitions apply:

gate: entry point towards a service that is operated inside a host

host controller: host that is also responsible for managing a host network

host network: network of two or more hosts

host: logical entity that operates one or more service(s)

pipe: logical communication channel between two gates from different hosts

registry: data related to a gate that are stored as a collection of parameter - value pairs

service: collection of related atomic functions in a host for management purposes or for performing contactless actions

3.2 Symbols

For the purpose of the present document the following symbols apply:

\[ G_{ID} \quad \text{gate identifier} \]
\[ H_{ID} \quad \text{host identifier} \]
\[ P_{ID} \quad \text{pipe identifier} \]
\[ \text{PIPE}_x \quad \text{a pipe with a } P_{ID} \text{ value equal to } x \]

3.3 Abbreviations

For the purposes of the present document, the following abbreviations apply:

AFI Application Family Identifier
AID Application Identifier
APDU Application Protocol Data Unit
BER-TLV Basic Encoding Rules - Tag, Length, Value
C-APDU Command APDU
CB Chaining Bit
CLF ContactLess Front-end
CLT ContactLess Tunnelling
CRC Cyclic Redundancy Code
HCI Host Controller Interface
HCP Host Controller Protocol
JIS Japanese Industrial Standard
NFC Near Field Communication
PA PreAmble
PCD Proximity Coupling Device
PICC Proximity Card
R-APDU Response APDU
3.4 Void

The content of this clause has been moved to clause 3A.

3A Coding conventions

For the purposes of the present document, the following coding conventions apply:

- All lengths are presented in bytes, unless otherwise stated. Each byte is represented by bits b8 to b1, where b8 is the most significant bit and b1 is the least significant bit. In each representation, the leftmost bit is the most significant bit.
- Hexadecimal values are specified between single quotes, e.g. ’1F’. In a sequence of bytes, byte 1 is the leftmost byte.
- All bytes specified as RFU shall be set to '00' and all bits specified as RFU shall be set to 0.

4 HCI architecture

4.1 Overview

A valid host network has a star topology where one or more hosts physically connect to a host controller. The HCI defines the interface between hosts. More specifically the HCI has three levels:

- a collection of gates that exchange commands, responses and events; and
- an HCP messaging mechanism; and
- an HCP routing mechanism that may optionally segment messages when required.

Figure 1 illustrates the HCP stack in a possible host network.
NOTE: For clarity only two gates are shown. In particular the host controller also has gates that connect via HCP to other hosts.

For proper operation, the HCP requires underlying data link layers to have the following properties:

- The data link layer (e.g. SWP) shall be error free and the order of the received/sent data shall be respected.
- The data link layer provides its own data flow control.
- The data link layer shall deliver packets of the upper layer up to a maximum size specific to the data link layer.
- The data link layer shall report the size of each received packet to its upper layer.

### 4.2 Hosts

The identity of a host is coded in a byte. Table 1 lists the reserved values for the host identifier, $H_{ID}$.

<table>
<thead>
<tr>
<th>Host</th>
<th>$H_{ID}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>host controller</td>
<td>'00'</td>
</tr>
<tr>
<td>terminal host</td>
<td>'01'</td>
</tr>
<tr>
<td>UICC host</td>
<td>'02'</td>
</tr>
<tr>
<td>RFU '03' to 'BF'</td>
<td></td>
</tr>
<tr>
<td>Proprietary 'C0' to 'FF'</td>
<td></td>
</tr>
</tbody>
</table>

In the present document, the generic term "host" is used to refer to any logical host (e.g. terminal host, UICC host) excluding the host controller.

### 4.3 Gates

A gate provides an entry point to a service that is operated inside a host. The HCP enables gates from different hosts to exchange messages. There are two types of gates:

- Management gates that are needed for the management of the host network.
- Generic gates that are not related to the management of the host networks. Only the generic aspects of these gates are defined in the HCI core.

The type of a gate is identified by a gate identifier. Gate identifiers are listed in table 2 and are either unique within the scope of a host ('10' to 'FF'), or their values refer to the same gate type for every host ('00' to '0F').

<table>
<thead>
<tr>
<th>Gate</th>
<th>$G_{ID}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>reserved for proprietary use</td>
<td>'00' to '03'</td>
</tr>
<tr>
<td>loop back gate</td>
<td>'04'</td>
</tr>
<tr>
<td>identity management gate</td>
<td>'05'</td>
</tr>
<tr>
<td>RFU</td>
<td>'06' to '0F'</td>
</tr>
<tr>
<td>Host specific</td>
<td>'10' to 'EF'</td>
</tr>
<tr>
<td>reserved for proprietary use</td>
<td>'F0' to 'FF'</td>
</tr>
</tbody>
</table>

Host specific gates for hosts defined in the present document (including RFU ones) are reserved for use in the present document.

The following rules apply to hosts and gates:

- all hosts and the host controller shall have one administration gate (see clause 7.1.1); and
• all hosts may have one link management gate (see clause 7.1.2) and the host controller shall have one link management gate; and
• all hosts and the host controller shall have one identity management gate (see clause 7.1.3); and
• all hosts and the host controller shall have one loop back gate (see clause 7.1.4); and
• all hosts and the host controller may have one or more generic gates (see clause 7.2).

4.4 Pipes

A pipe is a logical communication channel between two gates. There are two types of pipes:
• static pipes that are always available, i.e. they do not need to be created and cannot be deleted; and
• dynamic pipes that can be created and deleted.

The state of a pipe is either open or closed. The state shall remain persistent if the hosts are powered down and up again. It shall also remain persistent if a host is temporarily removed from the host network and was not replaced by a different device in the meantime. The state of a dynamic pipe after creation and the initial state of a static pipe shall be closed.

The pipe identifier, P_ID, is 7 bits long. The value of P_ID is used in the header of HCP packets as routing information (see clause 5.1). For static pipes the pipe identifiers are predefined with values as defined in table 3. For dynamic pipes, pipe identifiers are dynamically allocated by the host controller.

<table>
<thead>
<tr>
<th>P_ID</th>
<th>Pipe ending at:</th>
<th>Pipe type</th>
</tr>
</thead>
<tbody>
<tr>
<td>'00'</td>
<td>link management gate</td>
<td>static</td>
</tr>
<tr>
<td>'01'</td>
<td>administration gate</td>
<td>static</td>
</tr>
<tr>
<td>'02' to '6F'</td>
<td>other gates</td>
<td>dynamic</td>
</tr>
<tr>
<td>'70' to '7F'</td>
<td>RFU</td>
<td></td>
</tr>
</tbody>
</table>

The following rules apply to gates and pipes:
• a static pipe always connects a gate of a host to a gate of the host controller;
• a dynamic pipe connects two gates from different hosts; and
• static and dynamic pipes connect to different types of gates; see table 3 for the mapping; and
• dynamic pipe identifiers shall be unique in the host network.
Figure 2 illustrates hosts and gates in a valid host network.

NOTE: For clarity the identity management and loop back gates are not shown.

4.5 Registries

With every gate a registry template may be associated that defines parameters that are related to the gate. Parameters are identified by parameter identifiers consisting of one byte. Parameter identifiers are unique within the scope of the gate.

For all gates defined in the present document, parameter identifiers in the range of '00' to 'EF' are reserved for use in the present document. Parameter identifiers in the range of 'F0' to 'FF' may be used for proprietary purposes.

A new instance of the registry is created for every pipe that connects to the gate. Upon pipe creation all registry parameters with access rights Read-write (RW) or Write-only (WO) shall be set to their default values. Read-only (RO) parameters shall be set by the entity managing the registry to an appropriate value which may differ from the default values.

A host is responsible for managing its associated registries. The persistence of a registry and the default values for parameters shall be indicated with each registry description.

When a pipe is deleted its registry instance is also deleted.
5      HCP

5.1      HCP packets

Using the data link layer hosts exchange packets with the host controller. The format of a packet is defined in figure 3.

![Figure 3](image)

The interpretation of the fields in the packet header shall be the following:

- CB is the chaining bit and its value is equal to 1 except when message fragmentation is used (see clause 5.3); and
- P_ID specifies the pipe identifier.

The host controller uses the value of P_ID to forward a packet to the destination host. The destination host forwards the packet to the destination gate. Using these mechanisms any two gates that are connected by a pipe can exchange messages.

The host controller shall verify that the pipe identifier is used by a host involved in the creation of the pipe.

The size of a message is application specific. The message structure is described in clause 5.2.

5.2      HCP message structure

A message carries one instruction and optional data as defined in figure 4.

![Figure 4](image)

The meaning of the fields in the message header is the following:

- TYPE identifies the type of instruction; and
- INSTRUCTION identifies the instruction.
The following types of instructions are defined:

- commands (type value 0);
- events (type value 1); and
- responses to commands (type value 2).

Type value 3 is RFU. The instruction value further qualifies the command, event or response. All three types may carry data.

For instructions the following rules apply:

- An event is defined within the scope of the gate that accepts the events.
- A command is defined within the scope of the gate that accepts the command.
- A response is defined within the scope of its associated command.

A gate shall only accept a command or an event on a pipe when the state of that pipe is open unless otherwise stated. A gate shall not send a command or event on a pipe when it is waiting for a response to a previous command on that pipe unless otherwise stated. A gate shall interpret incoming events and commands even while it is waiting for a response to a previously sent command.

5.3 Message fragmentation

Message fragmentation shall be used when the size of the message is larger than supported by the underlying data link layer. Messages shall be fragmented according to the following rules:

- All message fragments shall have a packet header.
- The value of the chaining bit in the packet header is equal to 0, except for the packet with the last fragmented message where the value shall be 1.
- Only the first message fragment shall contain a message header.

Figure 5 illustrates how one message is split into two fragments.

The source gate is responsible for fragmenting messages. The destination gate is responsible for rebuilding the message from the fragmented messages.

If a reset of the underlying data link layer occurs, fragments of a partially received message shall be discarded and a partially sent message shall be re-sent from the beginning.
6 Instructions

The structure of the data field for the instructions is defined in the tables in this clause as the sequence of fields in the tables.

6.1 Commands

6.1.1 Overview

Table 4 lists all the commands that will be described in clauses 6.1.2 and 6.1.3. The interpretation of all the commands shall be the same for all gates specified in clause 7. There are two groups of commands:

- generic commands are applicable to all gates; and
- administration commands needed for the management of the host network.

Table 4

<table>
<thead>
<tr>
<th>Gate</th>
<th>Command</th>
<th>Description</th>
<th>Instruction</th>
</tr>
</thead>
<tbody>
<tr>
<td>All gates</td>
<td>RFU</td>
<td>all commands</td>
<td>'00'</td>
</tr>
<tr>
<td></td>
<td>ANY_SET_PARAMETER</td>
<td>the command to set a parameter in a registry</td>
<td>'01'</td>
</tr>
<tr>
<td></td>
<td>ANY_GET_PARAMETER</td>
<td>the command to get a parameter from a registry</td>
<td>'02'</td>
</tr>
<tr>
<td></td>
<td>ANY_OPEN_PIPE</td>
<td>the command to open a pipe</td>
<td>'03'</td>
</tr>
<tr>
<td></td>
<td>ANY_CLOSE_PIPE</td>
<td>the command to close a pipe</td>
<td>'04'</td>
</tr>
<tr>
<td></td>
<td>RFU</td>
<td></td>
<td>'05' to '0F'</td>
</tr>
<tr>
<td>Administration gate only</td>
<td>ADM_CREATE_PIPE</td>
<td>the administration command to create a dynamic pipe</td>
<td>'10'</td>
</tr>
<tr>
<td></td>
<td>ADM_DELETE_PIPE</td>
<td>the administration command to delete a dynamic pipe</td>
<td>'11'</td>
</tr>
<tr>
<td></td>
<td>ADM_NOTIFY_PIPE_CREATED</td>
<td>the administration command to notify the creation of a dynamic pipe</td>
<td>'12'</td>
</tr>
<tr>
<td></td>
<td>ADM_NOTIFY_PIPE_DELETED</td>
<td>the administration command to notify the deletion of a dynamic pipe</td>
<td>'13'</td>
</tr>
<tr>
<td></td>
<td>ADM_CLEAR_ALL_PIPE</td>
<td>the command to close/delete all pipes connected to a requesting host</td>
<td>'14'</td>
</tr>
<tr>
<td></td>
<td>ADM_NOTIFY_ALL_PIPE_CLEARED</td>
<td>the command to notify the closing/deletion of all pipes connected to a requesting host</td>
<td>'15'</td>
</tr>
<tr>
<td>Other non-proprietary gates</td>
<td>RFU</td>
<td></td>
<td>'16' to '3F'</td>
</tr>
<tr>
<td></td>
<td>Reserved</td>
<td></td>
<td>'10' to '3F'</td>
</tr>
</tbody>
</table>

For every command the following information is provided:

- a description of the command; and
- a list of parameters for the command; and
- the response on successful completion of the command (with optional parameters).

The mapping of possible response codes is given in clause 6.3.
6.1.2 Generic commands

6.1.2.1 ANY_SET_PARAMETER

The command ANY_SET_PARAMETER is used to write a parameter value into a gate registry.

The command parameters are as follows:

<table>
<thead>
<tr>
<th>Description</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>index of the parameter in the registry</td>
<td>1</td>
</tr>
<tr>
<td>value of the parameter; the length and content of the parameter depend on the registry</td>
<td>N</td>
</tr>
</tbody>
</table>

When successful the host shall respond with ANY_OK without any parameters.

6.1.2.2 ANY_GET_PARAMETER

The command ANY_GET_PARAMETER is used to read a parameter value from a gate registry.

The command parameters are as follows:

<table>
<thead>
<tr>
<th>Description</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>index of the parameter in the registry</td>
<td>1</td>
</tr>
</tbody>
</table>

When successful the host shall respond with ANY_OK with parameter as follows:

<table>
<thead>
<tr>
<th>Description</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>value of the parameter; the length and content of the parameter depends on the registry</td>
<td>N</td>
</tr>
</tbody>
</table>

6.1.2.3 ANY_OPEN_PIPE

The command ANY_OPEN_PIPE allows a host to open a pipe.

NOTE: This command is sent over a closed pipe.

The command has no parameters.

If the destination host is the host controller, and if the pipe is successfully opened the host controller shall send an ANY_OK response without any parameter.

For hosts other than the host controller, when the pipe is successfully opened, the destination host shall send an ANY_OK response with a parameter as follows:

<table>
<thead>
<tr>
<th>Description</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of pipes already open on this gate before the execution of the command</td>
<td>1</td>
</tr>
</tbody>
</table>
6.1.2.4 ANY_CLOSE_PIPE

The command ANY_CLOSE_PIPE allows a host to close a pipe. The command has no parameters. When the pipe is successfully closed the destination host shall send an ANY_OK response without parameters.

6.1.3 Administration commands

6.1.3.1 ADM_CREATE_PIPE

With the command ADM_CREATE_PIPE a host can request the host controller to create a new dynamic pipe between two gates. The host requesting the pipe shall be the source host. When successful a pipe is created between the source host and a destination host. The host controller shall use the WHITELIST defined by the destination host in order to verify that the source host is authorized to create a pipe.

The service that can be used on this pipe is determined only by the destination gate. Any gate identifier can be used as source gate; however, only one pipe is allowed to be created for each combination of source host/source gate and destination host/destination gate.

The command parameters are as follows:

<table>
<thead>
<tr>
<th>Description</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>source GID</td>
<td>1</td>
</tr>
<tr>
<td>destination HID</td>
<td>1</td>
</tr>
<tr>
<td>destination GID</td>
<td>1</td>
</tr>
</tbody>
</table>

When the pipe was successfully created, the host controller shall send the response ANY_OK with parameters as follows:

<table>
<thead>
<tr>
<th>Description</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>source HID</td>
<td>1</td>
</tr>
<tr>
<td>source GID</td>
<td>1</td>
</tr>
<tr>
<td>destination HID</td>
<td>1</td>
</tr>
<tr>
<td>destination GID</td>
<td>1</td>
</tr>
<tr>
<td>PID of pipe</td>
<td>1</td>
</tr>
</tbody>
</table>

6.1.3.2 ADM_NOTIFY_PIPE_CREATED

The command ADM_NOTIFY_PIPE_CREATED is sent by the host controller to a destination host to notify of the creation of a dynamic pipe. The source host is the host that requested the creation of the pipe from the host controller.

The command parameters are as follows:

<table>
<thead>
<tr>
<th>Description</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>source HID</td>
<td>1</td>
</tr>
<tr>
<td>source GID</td>
<td>1</td>
</tr>
<tr>
<td>destination HID</td>
<td>1</td>
</tr>
<tr>
<td>destination GID</td>
<td>1</td>
</tr>
<tr>
<td>PID of pipe</td>
<td>1</td>
</tr>
</tbody>
</table>
If the host accepts the pipe it shall send the response ANY_OK without parameters.

### 6.1.3.3 ADM_DELETE_PIPE

With the command ADM_DELETE_PIPE a host may request the host controller to delete a dynamic pipe. The host that requested the deletion of the pipe can only be the source host or destination host.

The command parameters are as follows:

<table>
<thead>
<tr>
<th>Description</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>P_ID of pipe</td>
<td>1</td>
</tr>
</tbody>
</table>

When the pipe is successfully deleted, the host controller shall send the response ANY_OK without parameters.

### 6.1.3.4 ADM_NOTIFY_PIPE_DELETED

The command ADM_NOTIFY_PIPE_DELETED is sent by the host controller to a host to notify the deletion of a dynamic pipe.

The command parameters are as follows:

<table>
<thead>
<tr>
<th>Description</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>P_ID of pipe</td>
<td>1</td>
</tr>
</tbody>
</table>

When the pipe has successfully been deleted, the host shall send the response ANY_OK without parameters.

### 6.1.3.5 ADM_CLEAR_ALL_PIPE

With the command ADM_CLEAR_ALL_PIPE a host requests the host controller:

- to delete all the dynamic pipes connected to the requesting host; and
- to close all static pipes connected to the requesting host; and
- to set all registry values related to static pipes to their default values.

For the data link layer specified in TS 102 613 [2], the command parameters are as follows:

<table>
<thead>
<tr>
<th>Description</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>Identity reference data</td>
<td>2</td>
</tr>
</tbody>
</table>

The identity reference data shall be used to initialize the reference data used by the host controller to check the UICC host identity. The identity reference data shall contain random elements.

When successful the host controller shall respond with an ANY_OK without parameters.

### 6.1.3.6 ADM_NOTIFY_ALL_PIPE_CLEARED

If the requesting host is not the host controller, the command ADM_NOTIFY_ALL_PIPE_CLEARED is sent by the host controller, following an ADM_CLEAR_ALL_PIPE command from the requesting host, to every host with at least one pipe to the requesting host to notify:

- that it deleted all dynamic pipes between the host and the requesting host.
If the requesting host is the host controller, the command ADM_NOTIFY_ALL_PIPE_CLEARED is sent by the host controller to a host to notify:

- that it deleted all dynamic pipes between the host controller and the host; and
- that it closed all static pipes between the host and the host controller.

The command parameters are as follows:

<table>
<thead>
<tr>
<th>Description</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>HID of requesting host</td>
<td>1</td>
</tr>
</tbody>
</table>

The host shall respond with an ANY_OK without parameters.

### 6.2 Responses

For the commands specified in clause 6.1, table 15 lists the possible response codes and table 16 their mapping to each command. Unless otherwise stated these responses carry no additional parameters. A response shall be sent to all commands received even to those unknown to the receiving gate. Responses received out of order (i.e. if no command was sent previously) shall be discarded.

<table>
<thead>
<tr>
<th>Value</th>
<th>Response</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>'00'</td>
<td>ANY_OK</td>
<td>command completed successfully (with optional parameters)</td>
</tr>
<tr>
<td>'01'</td>
<td>ANY_E_NOT_CONNECTED</td>
<td>the destination host is not connected</td>
</tr>
<tr>
<td>'02'</td>
<td>ANY_E_CMD_PAR_UNKNOWN</td>
<td>the format of the command parameters is wrong</td>
</tr>
<tr>
<td>'03'</td>
<td>ANY_E_NOK</td>
<td>command was rejected and/or not completed</td>
</tr>
<tr>
<td>'04'</td>
<td>ADM_E_NO_PIPES_AVAILABLE</td>
<td>no more dynamic pipes can be created</td>
</tr>
<tr>
<td>'05'</td>
<td>ANY_E_REG_PAR_UNKNOWN</td>
<td>the registry parameter identifier is either unknown to the registry or an optional registry parameter is not implemented</td>
</tr>
<tr>
<td>'06'</td>
<td>ANY_E_PIPE_NOT_OPENED</td>
<td>the pipe is not open</td>
</tr>
<tr>
<td>'07'</td>
<td>ANY_E_CMD_NOT_SUPPORTED</td>
<td>the command is not supported by the gate</td>
</tr>
<tr>
<td>'08'</td>
<td>ANY_E_INHIBITED</td>
<td>command is inhibited due to failure of lower layer identity check</td>
</tr>
<tr>
<td>'09'</td>
<td>ANY_E_TIMEOUT</td>
<td>an application level time-out occurred</td>
</tr>
<tr>
<td>'0A'</td>
<td>ANY_E_REG_ACCESS_DENIED</td>
<td>permission denied to write/read a value to/from a registry</td>
</tr>
<tr>
<td>'0B'</td>
<td>ANY_E_PIPE_ACCESS_DENIED</td>
<td>Permission denied to create a pipe due to a WHITELIST violation</td>
</tr>
<tr>
<td>'0C' to '3F'</td>
<td>RFU</td>
<td></td>
</tr>
</tbody>
</table>
Table 16

<table>
<thead>
<tr>
<th>RESPONSE CODE</th>
<th>ANY_OK</th>
<th>ANY_E_NOT_CONNECTED</th>
<th>ANY_E_CMD_PAR_UNKNOW</th>
<th>ANY_E_NOK</th>
<th>ADM_E_NO_PIPES_AVA</th>
<th>ANY_E_REG_PAR_UNKNOW</th>
<th>ANY_E_PIPE_NOT_OPENED</th>
<th>ANY_E_CMND_SUPPORT</th>
<th>ANY_E_PIPE_ACCESS_DENI</th>
<th>ANY_E_REG_ACCESS_DENI</th>
<th>ANY_E_INHIBITED</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANY_SET_PARAMETER</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>ANY_GET_PARAMETER</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>ANY_OPEN_PIPE</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>ANY_CLOSE_PIPE</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>ADM_CREATE_PIPE</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>ADM_NOTIFY_PIPE_CREATED</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>ADM_DELETE_PIPE</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>ADM_NOTIFY_PIPE_DELETED</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>ADM_CLEAR_ALL_PIPE</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>ADM_NOTIFY_ALL_PIPE_CLEARED</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>Command with RFU instruction value</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
</tbody>
</table>

• = This response code may result from this command.

6.3 Events

The interpretation of the events listed in table 17 shall be the same for all gates specified in clause 7. Unknown events received shall be discarded.

Table 17

<table>
<thead>
<tr>
<th>Value</th>
<th>Event</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>'01'</td>
<td>EVT_HCI_END_OF_OPERATION</td>
<td>This event shall be sent by a host to the host controller when the host requires no more activity on the HCI interface and its underlying layer at the time the event is sent. This event has no parameters.</td>
</tr>
<tr>
<td>'02'</td>
<td>EVT_POST_DATA</td>
<td>This event shall be used to send and receive data with a parameter that contains a block of data of variable size</td>
</tr>
<tr>
<td>'03'</td>
<td>EVT_HOT_PLUG</td>
<td>This event shall be sent by the host controller to any other connected host to notify the connection or disconnection of a host to the host controller. This event has no parameters.</td>
</tr>
</tbody>
</table>

All other event values to non-proprietary gates that are not assigned in the present document are RFU.

7 GATES

The next clauses define all gates. For each gate the following information is provided:

- a brief description of the gate's responsibility; and
- a list of parameter - value pairs for the gate's registry.

The mapping of gates to the supported events and commands are given in tables 18 and 19, respectively.
Table 18

<table>
<thead>
<tr>
<th>Gate</th>
<th>Link management</th>
<th>Host controller Link management</th>
<th>Host controller administration</th>
<th>Host administration</th>
<th>Identity management</th>
<th>Loop back</th>
<th>Generic</th>
</tr>
</thead>
<tbody>
<tr>
<td>EVT HCI END_OF_OPERATION</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>▲</td>
</tr>
<tr>
<td>EVT_POST_DATA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• ▲</td>
</tr>
<tr>
<td>EVT_HOT_PLUG</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>•</td>
</tr>
</tbody>
</table>

* = This command/event is supported by this Gate.
▲ = The host specific implementation of the generic gate defines if this command/event is supported.

Table 19

<table>
<thead>
<tr>
<th>GATE</th>
<th>Link management</th>
<th>Host controller administration</th>
<th>Host administration</th>
<th>Identity management</th>
<th>Loop back</th>
<th>Generic</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMAND</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ANY_SET_PARAMETER</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>▲</td>
</tr>
<tr>
<td>ANY_GET_PARAMETER</td>
<td>•</td>
<td></td>
<td>•</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ANY_OPEN_PIPE</td>
<td>•</td>
<td></td>
<td>•</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ANY_CLOSE_PIPE</td>
<td>•</td>
<td></td>
<td>•</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADM_CREATE_PIPE</td>
<td>•</td>
<td></td>
<td>•</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADM_NOTIFY_PIPE_CREATED</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADM_DELETE_PIPE</td>
<td>•</td>
<td></td>
<td>•</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADM_NOTIFY_PIPE_DELETED</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADM_CLEAR_ALL_PIPE</td>
<td>•</td>
<td></td>
<td>•</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADM_NOTIFY_ALL_PIPE_CLEARED</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

* = This command/event is supported by this Gate.
▲ = The host specific implementation of the generic gate defines if this command/event is supported.

NOTE: Host specific implementations of the generic gates define commands for transmission of data and additional events.
7.1 Management gates

7.1.1 Administration gates

7.1.1.1 Host controller administration gate

The administration gate in the host controller provides access to services that manage the network of pipes in the HCI network. In addition, this gate provides access to services that allow the discovery of hosts at the first startup and when the configuration of the host network has changed (see clause 8). The registry shall be persistent.

Table 20 lists the entries in the gate registry.

<table>
<thead>
<tr>
<th>Identifier</th>
<th>Parameter</th>
<th>Access Right</th>
<th>Comment</th>
<th>Length</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>'01'</td>
<td>SESSION_IDENTITY</td>
<td>RW</td>
<td>Session identifier that is used to detect if the connected host configuration changed.</td>
<td>8</td>
<td>'FFFFFFFFFFFFFFFFFFFFFFF'</td>
</tr>
<tr>
<td>'02'</td>
<td>MAX_PIPE</td>
<td>RO</td>
<td>Maximum number of created dynamic pipes supported by the host controller per host. The minimum value is '10' and the maximum value is '7D'.</td>
<td>1</td>
<td>'10'</td>
</tr>
<tr>
<td>'03'</td>
<td>WHITELIST</td>
<td>RW</td>
<td>List of hosts that may communicate with the host connected to this administration gate.</td>
<td>N</td>
<td>empty</td>
</tr>
<tr>
<td>'04'</td>
<td>HOST_LIST</td>
<td>RO</td>
<td>The list of the hosts that are accessible from this host controller including the host controller itself. The array contains a list of host identifiers.</td>
<td>N</td>
<td>'00'</td>
</tr>
</tbody>
</table>

The session identity shall be modified by the host whenever a modification of the configuration is performed by the host. The default value of the session identity shall never be written by a host. The session identity shall use random values.

Every host writes its WHITELIST into the host controller administration gate in order to inform the host controller which hosts are allowed to communicate with it. The host controller shall reject create pipe requests if the source host is not listed in the WHITELIST of the destination host.

The WHITELIST shall not contain the host controller identifier and host identifier accessing the WHITELIST.

The WHITELIST is an array containing a list of host identifiers as defined in table 1.

7.1.1.2 Host administration gate

The administration gate at a host provides access to services involved in the management of the pipes towards that host.

The administration gate at a host has no registry entries defined.
7.1.2  Link management gate

7.1.2.1  Host controller link management gate

The host controller link management gate provides information about the underlying layer. The registry may not be persistent.

Table 21 lists the entries in the registry.

<table>
<thead>
<tr>
<th>Identifier</th>
<th>Parameter</th>
<th>Access</th>
<th>Description</th>
<th>Length</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>'01'</td>
<td>REC_ERROR</td>
<td>RW</td>
<td>Number of invalid or lost frames previously sent by the host due to communication errors at the data link layer. This parameter can only be set to 0 in order to restart an error rate measure. When 'FFFF' is reached, the counter stops.</td>
<td></td>
<td>'0000'</td>
</tr>
</tbody>
</table>

7.1.2.2  Host link management gate

The host link management gate provides access to information related to the link layer. The registry may not be persistent.

Table 22 lists the entries in the registry.

<table>
<thead>
<tr>
<th>Identifier</th>
<th>Parameter</th>
<th>Access</th>
<th>Description</th>
<th>Length</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>'01'</td>
<td>REC_ERROR</td>
<td>RW</td>
<td>Number of invalid or lost frames previously sent by the host controller due to communication errors at the data link layer. This parameter can only be set to 0 in order to restart an error rate measure. When 'FFFF' is reached, the counter stops.</td>
<td></td>
<td>'0000'</td>
</tr>
</tbody>
</table>
7.1.3 Identity management gate

The identity management gate provides software and hardware information about the host. The registry shall be persistent.

This gate shall be provided by all hosts and the host controller. As destination gate, the identity management gate shall accept at least one pipe from each host in its WHITELIST.

Table 23 lists the entries in the registry.

<table>
<thead>
<tr>
<th>Type</th>
<th>Identifier</th>
<th>Parameter</th>
<th>Access</th>
<th>Right</th>
<th>Description</th>
<th>Length</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>OPTIONAL</td>
<td>'01'</td>
<td>VERSION_SW</td>
<td>RO</td>
<td></td>
<td>Version of the software defined by the vendor.</td>
<td>3</td>
<td>'000000'</td>
</tr>
<tr>
<td></td>
<td>'03'</td>
<td>VERSION_HARD</td>
<td>RO</td>
<td></td>
<td>Version of the hardware defined by vendor</td>
<td>3</td>
<td>'000000'</td>
</tr>
<tr>
<td></td>
<td>'04'</td>
<td>VENDOR_NAME</td>
<td>RO</td>
<td></td>
<td>Vendor name UTF8 coding. The maximum value for N shall be 20.</td>
<td>N0</td>
<td>N0=0</td>
</tr>
<tr>
<td></td>
<td>'05'</td>
<td>MODEL_ID</td>
<td>RO</td>
<td></td>
<td>Model identifier assigned by the vendor.</td>
<td>1</td>
<td>'00'</td>
</tr>
<tr>
<td></td>
<td>'02'</td>
<td>HCI_VERSION</td>
<td>RO</td>
<td></td>
<td>Version of HCI supported by the host.</td>
<td>1</td>
<td>'01'</td>
</tr>
<tr>
<td></td>
<td>'08'</td>
<td>MAX_CURRENT</td>
<td>R0</td>
<td></td>
<td>The maximum current provided/required by the host controller/host during operation (mA)</td>
<td>1</td>
<td>10</td>
</tr>
<tr>
<td>MANDATORY</td>
<td>'06'</td>
<td>GATES_LIST</td>
<td>RO</td>
<td></td>
<td>The list of all gates that accept dynamic pipes as an array of gate identifiers.</td>
<td>N1</td>
<td>N1=0</td>
</tr>
</tbody>
</table>

A host according to the present document shall set the HCI_VERSION parameter if provided to '01'. The HCI_VERSION parameter will be incremented for each release of this HCI specification that is not fully backward compatible with the previous release. Every host shall manage backward compatibility with previous HCI versions and use only commands and parameters defined in the present document having the lower HCI version number exposed by the 2 hosts involved in a transaction. A host connected to a host with higher HCI version number shall operate according to its own version.

The MAX_CURRENT parameter of the host controller represent the maximum current that it is able to provide to a host during operation as defined for the different contactless mode of operation. The MAX_CURRENT parameter of a host represent the current required by this host to allow optimum performance during contactless mode operation. A host can use this maximum current only during the period defined in this document.

7.1.4 Loop back gate

The loop back gate provides access to services for testing the HCI network. As destination gate, the loop back gate shall accept at least one pipe from each host in its WHITELIST.

The loop back gate has no registry entries defined.

7.2 Generic gates

Generic gates are not related to the management of the host network, but provide functions in the context of their host. For the contactless platform the functions and registries are defined in clauses 9 and 10.
8 HCI procedures

8.1 Pipe management

8.1.1 Pipe creation

Figure 6 illustrates how host A requests the creation of a dynamic pipe, between one of its gates and a gate in host B. All communications use PIPE1.

![Diagram](image)

The sequence flow is as follows:

1) Host A requests the host controller to create a pipe, PIPE<sub>x</sub>. The host controller shall verify that Host B administration gate WHITELIST contains Host A identifier. If Host A is not part of the WHITELIST of Host B, the host controller shall send ANY_E_PIPE_ACCESS_DENIED response to Host A and stop any further processing of this command.

2) The host controller assigns an unused pipe identifier.

3) The host controller notifies host B that host A requested the creation of PIPE<sub>x</sub>.

4) Host B responds to confirm that it accepts PIPE<sub>x</sub>.

5) The host controller responds that PIPE<sub>x</sub> has been created.

6) If host B does not accept the creation of the pipe, it shall respond with an appropriate response code.

When the host controller wants to create a pipe then the pipe identifier is assigned and only steps 2 and 3 are needed.

When a pipe is created towards the host controller then only steps 1 and 4 are needed.
Figure 7 illustrates how the WHITELIST affects a request from a host to create a pipe. In the following, the WHITELIST associated with Host B does not initially contain the HID for Host A and so the request is rejected. Following updating of the WHITELIST by Host B a further request is accepted.

![Figure 7](image)

The sequence flow is as follows:

1) Host A requests the host controller to create a pipe, PIPE_{x}. The WHITELIST in the host controller administration gate connected to the Host B administration gate does not contain the host identifier of the host A. The Host A is not authorized by Host B.

2) The host controller shall send ANY_E_PIPE_ACCESS_DENIED response to Host A and stop any further processing of this command.

3) The Host B updates its WHITELIST register in the host controller administration gate by adding the Host A identifier.

4) The host controller returns ANY_OK.

5) Host A requests the host controller to create a pipe. The host controller assigns an available pipe identifier.

6) The host controller notifies host B that host A requested the creation of PIPE_{x}.

7) Host B responds to confirm that it accepts PIPE_{x}.

8) The host controller responds that PIPE_{x} has been created.

If host B does not accept the creation of the pipe, it shall respond with an appropriate response code.
8.1.2 Pipe deletion

Figure 8 illustrates how host A requests the deletion of a dynamic pipe, $\text{PIPE}_x$, between one of its gates and a gate in host B. All communications use $\text{PIPE}_1$.

The sequence flow is as follows:

1) Host A requests the host controller to delete $\text{PIPE}_x$.

2) The host controller notifies host B.

3) Host B confirms that it deleted $\text{PIPE}_x$.

4) The host controller confirms to host A that the $\text{PIPE}_x$ is deleted.

Pipe deletion cannot be refused by host B.

A pipe created by host A can also be deleted by host B.

When $\text{PIPE}_x$ connects to a gate at the host controller then:

- the connecting host may request the deletion and only steps 1 and 4 are needed; or
- the host controller may request the deletion and only steps 2 and 3 are needed.
8.1.3 Clear all Pipes

Figure 9 illustrates how host A requests the host controller to delete all of its dynamic pipes and close all of its static pipes. All communications use PIPE1.

The sequence flow is as follows:

1) Host A requests the host controller to clear all pipes that connect to host A.
2) The host controller signals host B that all pipes are cleared towards host A.
3) Host B confirms.
4) The host controller signals host C that all pipes are cleared towards host A.
5) Host C confirms.
6) The host controller confirms to host A that pipes are cleared and host A closes its static pipes.

8.2 Registry access

Figure 10 shows how host A can read/write parameters in the registry of host B.
The sequence flow is as follows:

1) Host A requests a registry parameter from a gate of host B.
2) Host B replies with ANY_OK that includes the parameter value in its data.
3) Host A updates a registry parameter in a gate of host B.
4) Host B confirms that the parameter has been updated.

8.3 Host and Gate discovery

Figure 11 illustrates how a host may discover the hosts in a host network and the gates that a host supports.

The sequence flow is as follows:

1) Host A uses the administration gate, over PIPE₁, to request the list of connected hosts.
2) The host controller returns the list.
3) Host A selects a host from the list, i.e. host B, and requests the host controller, over PIPE₁, to create a pipe towards the identity gate of host B.
4) The host controller confirms the creation of the pipe, PIPEₓ. Details of the pipe creation are shown in clause 8.1.1.
5) Host A opens the PIPEₓ.
6) Host B confirms that PIPEₓ is open.
7) Host A requests the list of gates of host B.
8) The list of gates is returned.
9) Host A may request the VENDOR_NAME parameter from the registry.
10) The VENDOR_NAME parameter is returned.

Steps 9 and 10 are optional.

Optionally host A may close and delete PIPE\(_x\).

### 8.4 Session initialization

As pipe states are persistent, moving the host (e.g. UICC host) into a different terminal may result in inconsistent configurations. Using the session initialization procedure allows a host to detect that the host network has changed, either because the host controller itself changed or because the configuration has been changed. The host shall perform this procedure only at startup and only if no contactless transaction is pending. In case of an underlying data link layer according to TS 102 613 [2] this procedure shall only be performed after the initial interface activation in full-power mode and only if no contactless transaction is pending.

Figure 12 illustrates how a host A detects that the host controller has changed and the subsequent recovery mechanism.

![Figure 12](image)

The flow is as follows:

1) Host A queries over PIPE\(_i\) the SESSION.IDENTITY parameter in the administration gate of the host controller (an ANY.OPEN_PIPE command may precede this command).
2) If the returned value equals the previous value stored in the host then the procedure stops.
3) Otherwise host A needs to reinitialize and it requests the host controller to clear all pipes (see clause 8.1.3).
4) The host controller confirms and host A can perform any initializations.
5) Host A generates a new session identity and stores its value and stores it in the host controller registry.
6) The host controller confirms.

If the host controller stores a configuration (registry values and pipe states) of host A that is used before HCI initialization and need to be protected, an appropriate identity check mechanism shall be implemented by the lower layers. This identity check mechanism is out of the scope of the present document. In case the lower layer identity check fails, the host controller shall:

- execute only the following commands: ANY_OPEN_PIPE, ADM_CLEAR_ALLPIPE, ANY_GET_PARAMETER, and only if these are sent on PIPE1. All the other commands shall return ANY_E_INHIBITED and all events sent to the host controller shall be ignored; and
- return the default value of the SESSION.IDENTITY (see table 20). However the value of the SESSION.IDENTITY in the registry remains unchanged.

This inhibited state shall be terminated after processing a valid ADM_CLEAR_ALL_PIPE command.

Additional behaviour of the host controller in case of card emulation is defined in clause 9.4.6.

NOTE: It is up to host controller implementation to maintain the inhibited state after a subsequent successful identity check at the lower layer or not.

8.5 Loop back testing

Figure 13 shows how host A can verify the pipe connectivity to host B.

---

**Figure 13**

The sequence flow is as follows:

1) Host A requests over PIPE1 the creation of a pipe between any gate and the loop back gate in host B.

2) The host controller confirms the creation of the pipe, PIPEX. Details of the pipe creation are shown in clause 8.1.1.

3) Host A opens PIPEX.
4) Host B accepts the opening.

5) Host A sends the event EVT_POST_DATA with some data as parameter. The loopback gate shall support messages with a size of up to 250 bytes and may also support longer messages.

6) Host B sends back the event EVT_POST_DATA with the same data as received in the previous step. Upon reception host A may compare the received data against the original data.

Optionally host A may close and delete PIPEₐ.

9 Contactless card emulation

9.1 Overview

This clause specifies how a contactless card supporting multiple RF technologies is emulated on top of HCI core described in the previous clauses.

The RF technologies that are supported are:

- type A as defined in ISO/IEC 14443 parts 2 [5], 3 [6] and 4 [7]; or
- type B as defined in ISO/IEC 14443 parts 2 [5], 3 [6] and 4 [7]; or
- type B’ compliant with ISO/IEC 14443-2 [5] and with a standard framing as defined in ISO/IEC 14443-3 [6];
- type F as defined in ISO/IEC 18092 [4] as 212 kbps and 424 kbps passive mode.

NOTE 1: Type F technology, framing and protocol is also described in Japanese Industrial Standard JIS X 6319-4 [i.1]. The JIS terminology is different from ISO/IEC 18092 [4] but the technology is the same.

The contactless platform for card emulation mode is realized as a host network with two hosts; the host controller and the host (e.g. the UICC host). The CLF shall run the host controller of the host network. Additional hosts using the services of the CLF are out of the scope of the present document.

The host operates contactless card applications and the CLF handles the RF communication layers to the external contactless reader.

The host controller has one card RF gate for each RF technology it supports. For each card RF gate it wants to use, the host has one card application gate.

For the contactless platform for card emulation mode the pipes to card RF gates shall be created, opened, closed and deleted by the host. The RF technology of a card RF gate is active when there is an open pipe connected to it. The host shall not create more than one pipe to each RF gate.

For ISO/IEC 14443-4 [7] compliant contactless card applications the processing of the transmission protocol (i.e. up to the layer specified in ISO/IEC 14443-4 [7]) is done in the CLF. The card application gate and card RF gate exchange APDUs over their pipe.

For type B’ contactless card applications, the host controller and the host exchange raw frames. The raw frame structure is as defined in ISO/IEC 14443-3 [6]; but the data are transferred without the CRC_B.

In case of an underlying data link layer according to TS 102 613 [2] and for type A contactless card applications that indicate "not compliant with ISO/IEC 14443-4 [7]" in the SAK according to ISO/IEC 14443-3 [6], the CLF and the UICC exchange data using the CLT as defined in TS 102 613 [2].

For Type F contactless card applications, the host controller and the host exchange ISO/IEC 18092 [4] 212 kbps/424 kbps frames. The frame structure is as defined in ISO/IEC 18092 [4] but the data are transferred without PA and SYNC fields and E2.
The host is allowed to consume a current up to the maximum defined by the host controller in its identity management gate registry between the appearance and the disappearance of the RF unless restricted by the underlying layers e.g. TS 102 613 [2] where the restrictions for low-power mode and power saving mode still apply.

The host controller activates one or more RF technologies as requested by the host to the external reader. Further details of this activation process are out of scope for the present document.

NOTE 2: Application selection of a contactless card application (using e.g. AID) is out of scope for the present document.

Figure 14 illustrates a valid contactless platform.

NOTE: Non-standard applications refer to type A contactless card applications that indicate "not compliant with ISO/IEC 14443-4 [7]" in the SAK according to ISO/IEC 14443-3 [6].

Figure 14
9.2 Void

9.3 Gates

9.3.1 Void

Table 24: Void

9.3.2 Identity management gate

The following parameter shall be added to the identity management gate of the host controller as defined in clause 7.1.3.

Table 25

<table>
<thead>
<tr>
<th>Identifier</th>
<th>Parameter</th>
<th>Access Right</th>
<th>Description</th>
<th>Length</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>'07'</td>
<td>LOW_POWER_SUPPORT</td>
<td>RO</td>
<td>Indicates the power mode that is supported; where '01' indicates that low power mode is supported and '00' that it is not supported.</td>
<td>1</td>
<td>N/A</td>
</tr>
</tbody>
</table>

9.3.3 Card RF gates

9.3.3.1 Overview

Card RF gates are located in the host controller. A card RF gate controls the behaviour of a RF technology. Protocol and parameter settings related to the RF technology are available in the RF gate's registry.

The following clauses define the commands and events that a card RF gate supports in addition to those of a generic gate as defined in clause 7.2.

Table 26 lists the G_ID for the defined card RF gates.

Table 26

<table>
<thead>
<tr>
<th>Gate</th>
<th>G_ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type B card RF gate</td>
<td>'21'</td>
</tr>
<tr>
<td>Type B' card RF gate</td>
<td>'22'</td>
</tr>
<tr>
<td>Type A card RF gate</td>
<td>'23'</td>
</tr>
<tr>
<td>Type F card RF gate</td>
<td>'24'</td>
</tr>
<tr>
<td>Reserved for proprietary use</td>
<td>'25' to '29'</td>
</tr>
</tbody>
</table>

9.3.3.2 Commands

There are no additional commands defined.

9.3.3.3 Events

The card RF gates supports the event listed in table 27. The events are described in the following clauses.

Table 27

<table>
<thead>
<tr>
<th>Value</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>'10'</td>
<td>EVT_SEND_DATA</td>
</tr>
</tbody>
</table>
9.3.3.3.1 EVT_SEND_DATA

This event allows the sending of data to the CLF.

The event has parameters as follows:

**Table 28**

<table>
<thead>
<tr>
<th>Description</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>data that are sent</td>
<td>N</td>
</tr>
</tbody>
</table>

9.3.3.4 Registry

All registries shall be persistent.

9.3.3.4.1 RF technology type A

Table 29 defines the registry entries for a card RF gate for RF technology type A.

**Table 29**

<table>
<thead>
<tr>
<th>Identifier</th>
<th>Parameter</th>
<th>Access</th>
<th>Comment</th>
<th>Length</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>'01'</td>
<td>MODE</td>
<td>RW</td>
<td>Type A card emulation enabled indicator</td>
<td>1</td>
<td>'FF'</td>
</tr>
<tr>
<td>'02'</td>
<td>UID_REG</td>
<td>WO</td>
<td>UID as defined in ISO/IEC14443-3 [6] for type A</td>
<td>$N_0=0, 4, 7, 10$</td>
<td>$N_0=0$</td>
</tr>
<tr>
<td>'03'</td>
<td>SAK</td>
<td>RW</td>
<td>Preformatted SAK as defined in ISO/IEC 14443-3 [6] for type A</td>
<td>1</td>
<td>'00'</td>
</tr>
<tr>
<td>'04'</td>
<td>ATQA</td>
<td>RW</td>
<td>Preformatted ATQA; byte 1 shall contain b8..1, byte 2 b16..9 of the ATQA coded as defined in ISO/IEC 14443-3 [6] for type A</td>
<td>2</td>
<td>'0000'</td>
</tr>
<tr>
<td>'05'</td>
<td>APPLICATION_DATA</td>
<td>RW</td>
<td>The historical bytes as defined in ISO/IEC 14443-4 [7] for type A</td>
<td>$N_1$</td>
<td>$N_1=0$</td>
</tr>
<tr>
<td>'06'</td>
<td>FWI, SFGI</td>
<td>RW</td>
<td>Frame waiting time and Start-up frame guard time as defined in ISO/IEC 14443-4 [7] for type A</td>
<td>1</td>
<td>'EE'</td>
</tr>
<tr>
<td>'07'</td>
<td>CID_SUPPORT</td>
<td>RW</td>
<td>support for CID as defined ISO/IEC 14443-4 [7]</td>
<td>1</td>
<td>'00'</td>
</tr>
<tr>
<td>'08'</td>
<td>CLT_SUPPORT</td>
<td>RO</td>
<td>Support of tunnelling mode for ISO/IEC 14443-4 [7] non compliant protocols</td>
<td>1</td>
<td>'00'</td>
</tr>
<tr>
<td>'09'</td>
<td>DATARATE_MAX</td>
<td>RW</td>
<td>Maximum data rate supported</td>
<td>3</td>
<td>'030300'</td>
</tr>
</tbody>
</table>

Coding for the MODE is as follows:

- 'FF' = Type A card emulation not enabled; and
- '02' = Type A card emulation enabled; and
- All other values are RFU.

Coding for UID_REG is as follows:

- If Length equals 0 then the CLF shall generate a single size UID with uid0 = '08' and uid1 to uid3 as random number. The random number shall be generated only on state transitions from POWER-OFF to IDLE state (state definitions according to ISO/IEC 14443-3 [6]). The CLF shall interpret the absence of an RF-field as POWER-OFF state.
- If Length equals 4, 7 or 10 then the CLF shall use UID_REG as UID.

Coding for the CID_SUPPORT parameter:

- '01' = the support of the CID is required; the CLF shall set CID support in the ATS; and
- '00' = the support of the CID is not required; the CLF may or may not set CID support in the ATS.
• All other values are RFU.

Coding for the CLT_SUPPORT parameter:

• ’01’ = the CLF contains a tunnelling mode capability for type A ISO/IEC 14443-4 [7] non compliant protocol support; and

• ’00’ = the CLF does not contain any tunnelling mode capability for type A ISO/IEC 14443-4 [7] non compliant protocol support.

DATARATE_MAX codes the maximum divisor supported, with the coding as follows:

• Byte 1 defines the maximum divisor supported in direction PCD to PICC:
  - ’00’ = the maximum divisor supported is 1 (~106 kbit/s);
  - ’01’ = the maximum divisor supported is 2 (~212 kbit/s);
  - ’02’ = the maximum divisor supported is 4 (~424 kbit/s);
  - ’03’ = the maximum divisor supported is 8 (~848 kbit/s);
  - All other values are RFU.

• Byte 2 defines the maximum divisor supported in direction PICC to PCD:
  - ’00’ = the maximum divisor supported is 1 (~106 kbit/s);
  - ’01’ = the maximum divisor supported is 2 (~212 kbit/s);
  - ’02’ = the maximum divisor supported is 4 (~424 kbit/s);
  - ’03’ = the maximum divisor supported is 8 (~848 kbit/s);
  - All other values are RFU.

• Byte 3 defines a limitation to support different divisors for each direction:
  - ’00’ = different divisors for each directions (PICC to PCD and PCD to PICC) is supported;
  - ’01’ = only the same divisor in both direction (PICC to PCD and PCD to PICC) is supported;
  - All other values are RFU.

NOTE: The resulting actual maximum supported divisor indicated by the interface byte TA(1) as defined in ISO/IEC 14443-4 [7] is given by the minimum of:

  - the value and limitation as indicated in this registry; and
  - the maximum divisor implemented in the CLF.
9.3.3.4.2 RF technology type B

Table 30 defines the registry entries for a RF gate for RF technology type B.

<table>
<thead>
<tr>
<th>Identifier</th>
<th>Parameter</th>
<th>Access</th>
<th>Comment</th>
<th>Length</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>'01'</td>
<td>MODE</td>
<td>RW</td>
<td>Type B card emulation enabled indicator</td>
<td>1</td>
<td>'FF'</td>
</tr>
<tr>
<td>'02'</td>
<td>PUPI_REG</td>
<td>WO</td>
<td>PUPI as defined in ISO/IEC 14443-3 [6] for type B.</td>
<td>N₀</td>
<td>N₀=0</td>
</tr>
<tr>
<td>'03'</td>
<td>AFI</td>
<td>RW</td>
<td>Application family identifier as defined in ISO/IEC 14443-3 [6] for type B</td>
<td>1</td>
<td>'00'</td>
</tr>
<tr>
<td>'04'</td>
<td>ATQB</td>
<td>RW</td>
<td>Additional data for ATQB</td>
<td>4</td>
<td>See Table 31</td>
</tr>
<tr>
<td>'05'</td>
<td>HIGHER_LAYER_RESPONSE</td>
<td>RW</td>
<td>Higher Layer response in answer to ATTRIB command as defined in ISO/IEC 14443-3 [6] for type B</td>
<td>N₂</td>
<td>N₂=0</td>
</tr>
<tr>
<td>'06'</td>
<td>DATARATE_MAX</td>
<td>RW</td>
<td>Maximum data rate supported</td>
<td>3</td>
<td>'030300'</td>
</tr>
</tbody>
</table>

Coding for the MODE parameter:
- 'FF' = Type B card emulation not enabled; and
- '02' = Type B card emulation enabled; and
- All other values are RFU.

Coding for the PUPI_REG is as follows:
- In case N=0 then the CLF shall generate the PUI as dynamically generated number. The PUI shall only be generated by a state transition from the POWER-OFF to the IDLE state (state definitions according to ISO/IEC 14443-3 [6]). The CLF shall interpret the absence of an RF-field as POWER-OFF state.
- In all other cases, the CLF shall use the PUPI _REG as PUI.

Structure of the ATQB parameter.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Comment</th>
<th>Length</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRC_AID</td>
<td>Compressed CRC AID as described in the ISO/IEC 14443-3 [6]</td>
<td>2</td>
<td>'0000'</td>
</tr>
<tr>
<td>NUMBER_APLI</td>
<td>Number of applications as described in the ISO/IEC 14443-3 [6]</td>
<td>1</td>
<td>'00'</td>
</tr>
<tr>
<td>PROTO_INFO</td>
<td>Third byte of the Protocol information as described in the ISO/IEC 14443-3 [6]</td>
<td>1</td>
<td>'E4'</td>
</tr>
</tbody>
</table>

NOTE: PROTO_INFO also encodes FWI and support for CID, which are separate registry parameters in case of Type A.

DATARATE_MAX codes the maximum bit rates supported, with the coding as follows:
- Byte 1 defines the maximum bit rate supported in direction PCD to PICC:
  - '00' = the maximum bit rate supported is fc/128 (~106 kbps);
  - '01' = the maximum bit rate supported is fc/64 (~212 kbps);
  - '02' = the maximum bit rate supported is fc/32 (~424 kbps);
  - '03' = the maximum bit rate supported is fc/16 (~848 kbps);
  - All other values are RFU.
• Byte 2 defines the maximum bit rate supported in direction PICC to PCD:
  - '00' = the maximum bit rate supported is fc/128 (~106 kbps);
  - '01' = the maximum bit rate supported is fc/64 (~212 kbps);
  - '02' = the maximum bit rate supported is fc/32 (~424 kbps);
  - '03' = the maximum bit rate supported is fc/16 (~848 kbps);
  - All other values are RFU.

• Byte 3 represents the limitation of having the bit rate in both direction compulsory:
  - '00' = different bit rates in both directions (PICC to PCD and PCD to PICC) are supported;
  - '01' = only the same bit rate in both directions (PICC to PCD and PCD to PICC) is supported;
  - All other values are RFU.

NOTE: The resulting actual maximum bit rate supported is indicated in the first byte of the protocol information as defined in ISO/IEC 14443-3 [6] is given by the minimum of:
- the value and limitation as indicated in this registry; and
- the maximum divisor implemented in the CLF.

9.3.3.4.3 RF technology type B'

Table 32 defines the registry entries for a RF gate for the RF technology type 'B'.

<table>
<thead>
<tr>
<th>Identifier</th>
<th>Parameter</th>
<th>Access Right</th>
<th>Comment</th>
<th>Length</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>'01'</td>
<td>MODE</td>
<td>RW</td>
<td>Type B' card emulation enabled indicator</td>
<td>1</td>
<td>'FF'</td>
</tr>
<tr>
<td>'02'</td>
<td>PAT_IN</td>
<td>RW</td>
<td>Pattern to match with the incoming data</td>
<td>N0</td>
<td>N0=0</td>
</tr>
<tr>
<td>'03'</td>
<td>DAT_OUT</td>
<td>RW</td>
<td>Data to reply if the incoming data matches with PAT_IN pattern</td>
<td>N1</td>
<td>N1=0</td>
</tr>
</tbody>
</table>

Coding for the MODE parameter:
- 'FF' = Type B' card emulation not enabled; and
- '02' = Type B' card emulation enabled; and
- All other values are RFU.

9.3.3.4.4 RF technology Type F (ISO/IEC 18092 212 kbps/424 kbps card emulation only)

Table 33 defines the registry entries for a RF gate for the RF technology Type F.

<table>
<thead>
<tr>
<th>Identifier</th>
<th>Parameter</th>
<th>Access Right</th>
<th>Comment</th>
<th>Length</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>'01'</td>
<td>MODE</td>
<td>RW</td>
<td>Type F card emulation enabled indicator</td>
<td>1</td>
<td>'FF'</td>
</tr>
<tr>
<td>'02'</td>
<td>SPEED_CAP</td>
<td>RO</td>
<td>CLF Speed Capabilities for Type F</td>
<td>1</td>
<td>'01'</td>
</tr>
<tr>
<td>'03'</td>
<td>CLT_SUPPORT</td>
<td>RO</td>
<td>Support of tunnelling mode for anti-collision data exchange in type F card emulation</td>
<td>1</td>
<td>'00'</td>
</tr>
</tbody>
</table>
Coding for the MODE parameter:
- 'FF' = Type F card emulation not enabled; and
- '02' = Type F card emulation enabled; and
- All other values are RFU.

Table 34 defines the coding for the SPEED_CAP parameter.

<table>
<thead>
<tr>
<th>SPEED_CAP</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>b8</td>
<td>1: auto bit rate detection supported</td>
</tr>
<tr>
<td></td>
<td>0: auto bit rate detection not supported</td>
</tr>
<tr>
<td>b7..b3</td>
<td>RFU</td>
</tr>
<tr>
<td>b2</td>
<td>1: 424 kbps bit rate supported</td>
</tr>
<tr>
<td></td>
<td>0: 424 kbps bit rate not supported</td>
</tr>
<tr>
<td>b1</td>
<td>1: 212 kbps bit rate supported</td>
</tr>
<tr>
<td></td>
<td>0: 212 kbps bit rate not supported</td>
</tr>
</tbody>
</table>

Coding for the CLT_SUPPORT parameter:
- '01' = the CLF contains a tunnelling mode capability for type F card emulation anti-collision support; and
- '00' = the CLF does not contain any tunnelling mode capability for type F card emulation anti-collision support.

9.3.4 Card application gates

9.3.4.1 Overview

A card application gate controls the access towards contactless card applications.

The next clauses define the commands and events that a card application gate supports in addition to those of a generic gate as defined in clause 7.2.

The G\textsubscript{ID} for the card application gates are dynamically assigned by the host.

9.3.4.2 Commands

No additional command defined.

9.3.4.3 Events

The card application gates support the events listed in table 35. The events shall be described in the following clauses.

<table>
<thead>
<tr>
<th>Value</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>'10'</td>
<td>EVT_SEND_DATA</td>
</tr>
<tr>
<td>'11'</td>
<td>EVT_FIELD_ON</td>
</tr>
<tr>
<td>'12'</td>
<td>EVT_CARD_DEACTIVATED</td>
</tr>
<tr>
<td>'13'</td>
<td>EVT_CARD_ACTIVATED</td>
</tr>
<tr>
<td>'14'</td>
<td>EVT_FIELD_OFF</td>
</tr>
</tbody>
</table>
9.3.4.3.1 EVT_FIELD_ON

This event signals that the CLF detected an RF carrier from an external reader. This event shall be sent by the host controller within 2ms after the detection of an RF field.

In case of an underlying data link layer according to TS 102 613 [2], the following exception applies: if SWP is in DEACTIVATED state, the CLF shall activate the interface instead of sending the EVT_FIELD_ON.

The event has no parameters.

9.3.4.3.2 EVT_CARD_DEACTIVATED

This event signals that the CLF is deactivated as defined in ISO/IEC 14443-3 [6] for type B or ISO/IEC 14443-4 [7] for type A.

The event has no parameters.

9.3.4.3.3 EVT_CARD_ACTIVATED

This event signals that the CLF is activated as defined in ISO/IEC 14443-3 [6] for type B or ISO/IEC 14443-4 [7] for type A.

The event has no parameters.

9.3.4.3.4 EVT_FIELD_OFF

This event signals that the CLF detected the loss of the RF carrier from the external reader.

The event has no parameters.

9.3.4.3.5 EVT_SEND_DATA

This event allows the sending of data to the host from the CLF. The event has parameters as follows:

<table>
<thead>
<tr>
<th>Description</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>data that are sent</td>
<td>N</td>
</tr>
<tr>
<td>RF error indicator</td>
<td>1</td>
</tr>
</tbody>
</table>

Coding for the RF error indicator:

- '00' = no error; and
- '01' = error; and
- All other values are RFU.

NOTE: The end of the data and the beginning of the RF error indicator can unambiguously be derived from the message size. The RF error indicator is the last parameter, as the first segments of a long segmented message may already have been sent on the interface to the host before the sender (CLF) can do the integrity check of the original message received on the contactless interface.

The event shall be discarded by the host when the error indicator is set to '01'.

9.3.4.4 Registry

The card application gate does not have registry entries defined.
9.4 Procedures

The following procedures are described for full power mode. If the procedures are different in low power mode then these exceptions are described.

The CLF shall only send events defined in Table 35 to the Card application gate or open a CLT session as defined in TS 102 613 [2] for the corresponding RF technology if the MODE parameter in the associated Card RF gate registry is set to ‘02’ (enabled).

9.4.1 Use of contactless card application

Figure 15 illustrates the messages that type A and type B card RF gates and a card application gate exchange when entering a reader field.

![Figure 15](image-url)

The sequence flow is as follows:

1) When the CLF detects a RF field, the card RF gate shall send the event EVT_FIELD_ON to the card application gate as specified in clause 9.3.4.3.1. When there are multiple open card RF gates the CLF shall send the EVT_FIELD_ON to the open card application gate with the lowest GID. Next the CLF starts the initialization and anti-collision process as defined in ISO/IEC 14443-3 [6].

In low power mode, the event EVT_FIELD_ON may be omitted.

2) At the end of the activation sequence as defined ISO/IEC 14443-4 [7] the card RF gate shall send EVT_CARD_ACTIVATED to the card application gate to indicate that APDUs may be exchanged.

3) The card RF gate shall forward the C-APDUs from the external contactless reader to the card application gate using the EVT_SEND_DATA.

4) The host shall reply by sending the R-APDU in an EVT_SEND_DATA event to the card RF gate. The CLF may forward an empty C-APDU. The host shall reply to this with either an empty R-APDU or an R-APDU containing an error code.

5) The steps 3 and 4 may be repeated.

6) At the end of the deactivation sequence the card RF gate shall send an EVT_CARD_DEACTIVATED to signal that no more APDUs can be exchanged. This event will only be sent when the external contactless reader supports it.
When the CLF detects at any time during the sequence above that the RF field is off, the card RF gate shall send EVT_FIELD_OFF to the card application gate. When there are multiple open card RF gates the CLF shall send the EVT_FIELD_OFF to the card application gate used during the transaction or to the open card application gate with the lowest GID. In low power mode the event EVT_FIELD_OFF may be omitted if the host is powered down instead.

9.4.2 Non ISO/IEC 14443-4 type A

For type A contactless card applications that indicate "not compliant with ISO/IEC14443-4" in the SAK according to ISO/IEC 14443-3 [6] the flow is as follows:

- The card RF gate shall send the EVT_FIELD_ON as specified in clause 9.4.1. The CLF handles initialization and anti-collision. In low power mode the event EVT_FIELD_ON may be omitted.
- Any other communications are done using the CLT mode as defined in TS 102 613 [2].

When the CLF detects at any time during the sequence above that the RF field is off, the card RF gate shall send EVT_FIELD_OFF as specified in clause 9.4.1. In low power mode the event EVT_FIELD_OFF may be omitted if the host is powered down instead.

9.4.3 Type B' RF technology

For type B' contactless card applications the flow is as follows:

- The card RF gate shall send the EVT_FIELD_ON as specified in clause 9.4.1. In low power mode the event EVT_FIELD_ON may be omitted.
- The RF incoming raw frames are matched with the pattern in the parameter PAT_IN. The rules for matching are as follows:
  - PAT_IN shall have an even number of bytes. The first half is the reference data REF, the second half is the mask M. If RFIN represents the received frame (the data bytes between SOF and the CRC), matching is successful if the following bitwise calculation has an all ones result:
    - \((\text{RFIN} \oplus \text{REF}) \lor M\).
- If the matching is successful then the CLF shall prefix the first byte of the received frame to the bytes of the parameter DAT_OUT, add the CRC and send the result to the RF reader. The time between the end of the RF incoming frame and the beginning of the answer to the reader shall be less than 1 ms.
- Before the successful matching the incoming raw frames are not transferred to the host.

When the CLF detects at any time during the sequence above that the RF field is off, the card RF gate shall send EVT_FIELD_OFF to the card application gate as specified in clause 9.4.1. In low power mode the event EVT_FIELD_OFF may be omitted if the host is powered down instead.

9.4.4 Type F RF technology

For ISO/IEC 18092 [4] based contactless card emulation applications in 212 kbps/424 kbps passive mode as used in existing widespread infrastructure, the following applies:

1) The card RF gate sends the EVT_FIELD_ON as specified in clause 9.4.1. In low power mode the event EVT_FIELD_ON may be omitted.
2) In case SWP as defined in TS 102 613 [2] is used as a data link layer, the initialization data exchange is performed using CLT as defined in TS 102 613 [2] The host provides information for the initialization.
3) The card RF gate shall forward the ISO/IEC 18092 [4] 212 kbps/424 kbps frames from the external reader to the card application gate using the EVT_SEND_DATA. The CLF shall not forward RF frames to the host if a response from the host is pending. If an RF frame was received but the response from the host is still pending the received RF frame shall be discarded.
The resulting HCP packet has the following structure:

![Image](image)

**Figure 15a**

4) The host sending a response shall encapsulate the ISO/IEC 18092 [4] 212 kbps/424 kbps frames in an EVT_SEND_DATA event and send it to the card RF gate. The host shall send an empty EVT_SEND_DATA event to the card RF gate if there is no application data to be sent. The CLF shall not send anything to RF but shall be able to receive RF frames after receiving an empty EVT_SEND_DATA from the host.

5) Steps 3 and 4 may be repeated.

When the CLF detects at any time during the sequence above that the RF field is off, the card RF gate shall send EVT_FIELD_OFF to the card application gate as specified in clause 9.4.1. In low power mode the event EVT_FIELD_OFF may be omitted if the host is powered down instead.

ISO/IEC 18092 [4], 212 kbps/424 kbps frames, except initialization command and response (command code '00' and '01), shall be exchanged using the appropriate gate depending on the command code of the frame as described in table 37. The command code (CMD0) is the byte after LEN as defined in ISO/IEC 18092 [4]. The command codes reserved for the NFCIP-1 protocol shall not be forwarded.

<table>
<thead>
<tr>
<th>Command Code (CMD0)</th>
<th>Send via</th>
</tr>
</thead>
<tbody>
<tr>
<td>'00' (POLLING_REQUEST), '01' (POLLING_RESPONSE)</td>
<td>CLT mode</td>
</tr>
<tr>
<td>Other</td>
<td>ISO/IEC 18092 [4] 212 kbps/424 kbps card emulation application gate</td>
</tr>
</tbody>
</table>

### 9.4.5 Update RF technology settings

To avoid inconsistencies in parameter settings, the following procedure is recommended when updating the settings in the CLF for the different RF technologies:

1) The host first sets the MODE parameter for the RF technology to be updated to 'FF' to disable the use of this technology by the CLF for the host.

2) Now all other parameters of the RF technology in the CLF can be updated by the host.

3) Finally, the host can enable the RF technology again by setting the MODE parameter in the CLF to '02'.

### 9.4.6 Identity check

For the data link layer specified in TS 102 613 [2] the following shall apply:

In addition to what is defined in clause 8.4, in case the lower layer identity check fails, the host controller shall inhibit card emulation mode for the UICC host, i.e. the host controller shall not respond to an external contactless reader with any parameter from the card emulation registries related to the UICC host.
10 Contactless reader

10.1 Overview

This clause specifies how a contactless reader supporting multiple RF technologies is emulated on top of HCI core.

The RF technologies that are supported are:

- type A as defined in ISO/IEC 14443 parts 2 [5], 3 [6] and 4 [7]; or
- type B as defined in ISO/IEC 14443 parts 2 [5], 3 [6] and 4 [7].

The contactless platform for reader mode is realized as a host network with two hosts; the host controller and the host (e.g. the UICC host). The CLF shall run the host controller of the host network. Additional hosts using the services of the CLF are out of the scope of the present document.

The platforms for reader and card emulation mode can coexist.

The host operates contactless reader applications and the CLF handles the RF communication layers towards the external contactless card(s).

The host controller has one reader RF gate for each RF technology it supports. For each reader RF gate it wants to use the host has one reader application gate. The host shall not create more than one pipe to each reader RF gate.

The CLF handles the RF layers of the communications as defined in ISO/IEC 14443-2 [5].

The anti-collision and activation as defined in ISO/IEC 14443-3 [6] is handled by the CLF under the control of the host.

The RF protocol as defined in ISO/IEC 14443-4 [7] is handled by the CLF.

The reader RF gate and reader application gate exchange APDUs defined in ISO/IEC 7816-4 [8] over their pipe.

The host is allowed to consume a current up to the maximum defined by the host controller in its identity management gate registry between the notification of a target discovery and the end of operation on this target unless restricted by the underlying layers e.g. TS 102 613 [2] where the restrictions for low-power mode and power saving mode still apply.

Figure 16 illustrates a valid contactless platform.

![Figure 16](image-url)
10.2 Reader RF gates

10.2.1 Overview

A reader RF gate controls the behaviour of a RF technology. Protocol and parameter settings related to the reader RF technology are available in the reader RF gate's registry. The reader RF gates may accept a single target in the field of the reader irrespective of the RF technology.

The following clauses define the commands and events that a reader RF gate supports in addition to those of a generic gate as defined in clause 7.2.

Table 38 lists the G_ID for the defined reader RF gates.

<table>
<thead>
<tr>
<th>Gate</th>
<th>G_ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISO/IEC 14443 Type B reader RF gate</td>
<td>'11'</td>
</tr>
<tr>
<td>reserved for proprietary use</td>
<td>'12'</td>
</tr>
<tr>
<td>ISO/IEC 14443 Type A reader RF gate</td>
<td>'13'</td>
</tr>
<tr>
<td>reserved for proprietary use</td>
<td>'14' to '1B', '7E', '7F'</td>
</tr>
</tbody>
</table>

10.2.2 Command

10.2.2.1 WR_XCHG_DATA

This command allows the sending of data from a reader application gate to the reader RF gate; the data sent in this command is intended for transmission over the RF. A reply from an external device is returned to the reader application gate in the response to this command.

Table 39

<table>
<thead>
<tr>
<th>Value</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>'10'</td>
<td>WR_XCHG_DATA</td>
</tr>
</tbody>
</table>

This command has following parameters:

<table>
<thead>
<tr>
<th>Description</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>CTR</td>
<td>1</td>
</tr>
<tr>
<td>data sent</td>
<td>N</td>
</tr>
</tbody>
</table>

The CTR field is defined as follows:

- b8 to b6: RFU.
- b5: Time-out activation: If b5 is set to zero, application level time-out is deactivated. If b5 is set to one, the time-out is given in b4 to b1.
- b4 to b1: Time-out. This Value (0 up to 14) is used to calculate the application level time-out by the following formula:

\[
\text{Time-out} = \left(256 \times 16 / 13.56 \text{ MHz}\right) \times 2 ^ \text{Value}.
\]
If b5 of the CTR field is set to one and no response is received by the CLF from a target within the time-out period commencing after the error free transmission of this command by the CLF to the target, the CLF shall respond to the host with ANY_E_TIMEOUT with no parameters and shall discard data received from the target thereafter.

If the CLF detects an RF transmission error and the error detection and recovery procedure defined by ISO/IEC 14443-4 [7] does not succeed then the CLF shall respond to the host with ANY_OK with the error indicator set to '01'.

When successful the host shall respond with ANY_OK with parameter as follows:

<table>
<thead>
<tr>
<th>Description</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data received</td>
<td>N</td>
</tr>
<tr>
<td>RF error indicator</td>
<td>1</td>
</tr>
</tbody>
</table>

Coding for the RF error indicator:

- '00' = no error; and
- '01' = non-recoverable error; and
- All other values are RFU.

NOTE: The first segment of a long segmented message may already have been sent on the interface to the host before the host controller can do the integrity check of the original message received on the contactless interface. The same response is to be used if the transmission of the response over the contactless interface is interrupted.

10.2.3 Registries

10.2.3.1 Type A reader RF gate

Table 42 defines the registry for the RF gate ISO/IEC 14443 Type A reader RF gate. The registry is not persistent. The values '02' to '06' are updated after each target activation. The value '07' is updated when the operating status changes.

<table>
<thead>
<tr>
<th>Type</th>
<th>Identifier</th>
<th>Parameter</th>
<th>Access Right</th>
<th>Comment</th>
<th>Length</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>MANDATORY</td>
<td>'02'</td>
<td>UID</td>
<td>RO</td>
<td>UID as defined in ISO/IEC 14443-3 [6] for type A</td>
<td>4, 7, 10</td>
<td>080000000'</td>
</tr>
<tr>
<td></td>
<td>'04'</td>
<td>ATQA</td>
<td>RO</td>
<td>Preformatted ATQA; byte 1 shall contain b8..1, byte 2 b16..9 of the ATQA coded as defined in ISO/IEC 14443-3 [6] for type A</td>
<td>2</td>
<td>0000'</td>
</tr>
<tr>
<td></td>
<td>'05'</td>
<td>APPLICATION_DATA</td>
<td>RO</td>
<td>The historical bytes as defined in ISO/IEC 14443-4 [7] for type A</td>
<td>N</td>
<td>N=0</td>
</tr>
<tr>
<td></td>
<td>'03'</td>
<td>SAK</td>
<td>RO</td>
<td>Preformatted SAK as defined in ISO/IEC 14443-3 [6] for type A</td>
<td>1</td>
<td>'00'</td>
</tr>
<tr>
<td></td>
<td>'06'</td>
<td>FWI, SFGT</td>
<td>RO</td>
<td>Frame waiting time and start-up frame guard time as defined in ISO/IEC 14443-4 [7] for type A</td>
<td>1</td>
<td>'EE'</td>
</tr>
<tr>
<td></td>
<td>'01'</td>
<td>DATARATE_MAX</td>
<td>RW</td>
<td>Maximum data rate supported</td>
<td>1</td>
<td>'00'</td>
</tr>
<tr>
<td></td>
<td>'07'</td>
<td>OPERATING_STATUS</td>
<td>RO</td>
<td>Indication of the current operating status of the reader</td>
<td>1</td>
<td>'00'</td>
</tr>
<tr>
<td></td>
<td>'08'</td>
<td>STATUS_EVENT_EN</td>
<td>RW</td>
<td>Enable for reader status events</td>
<td>1</td>
<td>'00'</td>
</tr>
</tbody>
</table>

Coding for the DATARATE_MAX is as follows:
only the same divisor (ISO/IEC 14443-4 [7]) for both directions (PICC to PCD and PCD to PICC) is supported:

- '00' = the maximum divisor supported is 1.
- '01' = the maximum divisor supported is 2.
- '02' = the maximum divisor supported is 4.
- '03' = the maximum divisor supported is 8.

NOTE: The actual maximum supported divisor is given by the minimum of:

- the value as indicated in this registry; and
- the maximum divisor implemented in the CLF.
- All other values are RFU.

Coding of the OPERATING_STATUS is as follows:

b8-2 : RFU
b1 : Reader status

The CLF shall set b1 to 0 when the Type A reader mode is not available for the host (e.g. the CLF is exclusively active for another host).

The CLF shall set b1 to 1 when the Type A reader mode is available for the host.

Coding of the STATUS_EVENT_EN is as follows:

b8-2 : RFU
b1 : Reader status event enable

The Reader status event enable (b1) indicates to the CLF that the host requests events to be sent when the reader status changes and by the same method may indicate to the CLF that it does not wish to receive such events as specified in detail in clause 10.3.4.2.
10.2.3.2 Type B reader RF gate

Table 43 defines the registry for the RF gate ISO/IEC 14443 Type B reader RF gate. The registry is not persistent. The values '01', '03', '04' and '02' (read value) are updated after each target activation. The value '07' is updated when the operating status changes.

| Type | Identifier     | Parameter          | Access Right | Comment                                                  | Length | Default |
|------|----------------|--------------------|--------------|---------------|----------------------------------------------------------|--------|---------|
| MANDATORY | '03' | PUPI               | RO           | PUPI as defined in ISO/IEC 14443-3 [6] for type B       | N_0    | N_0=0   |
|       | '04' | APPLICATION_DATA  | RO           | Application data as defined in ISO/IEC 14443-3 [6] for ATQB | N_1    | N_1=0   |
|       | '02' | AFI                | RW           | Application family identifier as defined in ISO/IEC 14443-3 [6] for type B used by the reader to poll the target | 1      | '00'    |
|       | '01' | HIGHER_LAYER_RESPONSE | RO      | Higher layer response as defined in ISO/IEC 14443-3 [6] for answer to ATTRIB command | N_2    | N_2=0   |
|       | '05' | HIGHER_LAYER_DATA  | RW           | Higher layer data as defined in ISO/IEC 14443-3 [6] for ATTRIB command | N_3    | N_3=0   |
| MANDATORY, not present in Rel-10 and earlier | '07' | OPERATING_STATUS  | RO           | Indication of the current operating status of the reader | 1      | '00'    |
|       | '08' | STATUS_EVENT_EN   | RW           | Enable for reader status events | 1      | '00'    |

Coding of the AFI is as follows:
- The value written to the registry shall be used by the reader to poll the target.
- Once a target is activated, the registry value shall reflect the AFI value of the target.

Coding of the OPERATING_STATUS is as follows:
- b8-2 : RFU
- b1 : Reader status

The CLF shall set b1 to 0 when the Type B reader mode is not available for the host (e.g. the CLF is exclusively active for another host).

The CLF shall set b1 to 1 when the Type B reader mode is available for the host.

Coding of the STATUS_EVENT_EN is as follows:
- b8-2 : RFU
- b1 : Reader status event enable

The Reader status event enable (b1) indicates to the CLF that the host requests events to be sent when the reader status changes and by the same method may indicate to the CLF that it does not wish to receive such events as specified in detail in clause 10.3.4.2.

10.2.4 Events

The reader RF gates support the events listed in table 44. The events shall be described in the following clauses.
Table 44

<table>
<thead>
<tr>
<th>Value</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>'10'</td>
<td>EVT_READER_REQUESTED</td>
</tr>
<tr>
<td>'11'</td>
<td>EVT_END_OPERATION</td>
</tr>
</tbody>
</table>

10.2.4.1 EVT_READER_REQUESTED

With this event the host signals to the CLF that a reader application gate is requesting to activate the RF polling (turn on the RF carrier). The method and the algorithm used by the reader RF gates to achieve target polling are implementation dependant and out of the scope of the present document. This event has no parameters and may be sent over any open pipe of any reader RF gate.

10.2.4.2 EVT_END_OPERATION

The host sends this event to inform the CLF Controller of the end of operation. Upon reception of the event the CLF controller shall turn the RF field OFF if the EVT_TARGET_DISCOVERED has been previously sent to that specific host, otherwise the CLF may turn the RF field OFF and move forward in its internal processing.

10.2.5 Responses

For the command specified in clause 10.2.2.1, table 45 lists the possible response codes. Unless otherwise stated these responses carry no additional parameters.

Table 45

<table>
<thead>
<tr>
<th>Value</th>
<th>Response</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>'00'</td>
<td>ANY_OK</td>
<td>command completed successfully (with optional parameters)</td>
</tr>
<tr>
<td>'03'</td>
<td>ANY_E_NOK</td>
<td>command was rejected and/or not completed</td>
</tr>
<tr>
<td>'09'</td>
<td>ANY_E_TIMEOUT</td>
<td>an application level time-out occurred</td>
</tr>
<tr>
<td>'10'</td>
<td>WR_RF_ERROR</td>
<td>The target has returned an RF error</td>
</tr>
</tbody>
</table>

10.3 Reader application gates

10.3.1 Overview

A reader application gate controls the access to contactless reader applications.

The following clauses define the commands and events that a reader application gate supports in addition to those of a generic gate as defined in clause 7.2.

The GID for the reader application gates are dynamically assigned by the host.

10.3.2 Command

No additional command defined.

10.3.3 Registry

The reader application gate does not have registry entries defined.

10.3.4 Events

The reader application gates support the events listed in table 46. The event shall be described in the following clause.
Table 46

<table>
<thead>
<tr>
<th>Value</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>'10'</td>
<td>EVT_TARGET_DISCOVERED</td>
</tr>
<tr>
<td>'11'</td>
<td>EVT_READER_STATUS</td>
</tr>
</tbody>
</table>

10.3.4.1 EVT_TARGET_DISCOVERED

This event signals to the reader application gate the existence of an RF target in the field of the activated RF technology. The event notifies the reader application gate about the availability of the RF information in the registry. The activation of the target is completed and the associated registry may inform the application about the RF feature of the application.

This event has one parameter:

Table 47

<table>
<thead>
<tr>
<th>Description</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>STATUS</td>
<td>1</td>
</tr>
</tbody>
</table>

Coding for the status field:

- '00' = when there is a single target in the reader field and the activation of the target is completed; and
- '03' = when there are several targets in the field irrespective of the RF technology.

10.3.4.2 EVT_READER_STATUS

This event indicates to a host that the availability status of the reader RF gate for a particular RF technology has changed.

The CLF shall send this event (see also Annex A)

- once after the reader status event enable bit (b1) in the STATUS_EVENT_EN is changed to 1, and
- as long as the reader status event enable bit (b1) in the corresponding STATUS_EVENT_EN remains at 1 whenever the availability status of a reader RF gate has changed.

Even if multiple pipes from one host to reader RF gates are open, this event shall be sent over one of these pipes only.

If different types of the reader mode become available/unavailable for the host at different points in time, the CLF shall send one event for each change.

The CLF shall not send the EVT_READER_STATUS if the reader status event enable bit (b1) in the corresponding STATUS_EVENT_EN is set to 0.

This event has one parameter:

Table 47a

<table>
<thead>
<tr>
<th>Description</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>READER_STATUS</td>
<td>1</td>
</tr>
</tbody>
</table>

Coding for the reader status field:

b8-3 : RFU

b1 : Type A reader status

0 = reader mode for Type A targets is not available for this host.

1 = reader mode for Type A targets is available for this Host.
b2 : Type B reader status

0 = reader mode for Type B targets is not available for this host.
1 = reader mode for Type B targets is available for this host.

10.4 Procedures

The following procedures are described for the reader mode.

10.4.1 Use of contactless reader application

Figure 17 illustrates the messages that type A and type B reader RF gates and a reader application gate exchange when detecting a target.

![Figure 17](image)

The sequence flow is as follows:

1) When the host sends the EVT_READER_REQUESTED to the CLF to enable the RF polling. The CLF starts the detecting of a target according to all reader RF gates of the host that have an open pipe. The host shall send the EVT_READER_REQUESTED on a single pipe only.

2) A target has been detected and activated. The CLF notifies the host via the event EVT_TARGET_DISCOVERED. If there are several targets in the field then the procedure shall stop. The host may read the reader RF gate registry to get RF specific information (e.g. AFI, etc.).

3) The reader application gate may send C-APDU over the pipe by using the command WR_XCHG_DATA.

4) The reader RF gate replies in sending back an R-APDU.

The steps 3 and 4 may be repeated.

5) When the operation is completed, the host shall send the EVT_END_OPERATION to turn off the RF carrier as defined above. The host shall send the EVT_END_OPERATION on a single pipe only.
10.4.2 Contactless reader not available

Figure 17a illustrates the messages that type A and type B reader RF gates and a reader application gate may exchange when the reader is not available.

The sequence flow is as follows:

1) Before starting reader mode, the host checks if the reader is available.
2) If the reader is not available, the host enables the reader status event.

NOTE: Step 1 is optional. The host may also start directly with enabling of the reader status event.

3) In response the CLF reports that the reader is (still not) available.
4) The host sends the event reader requested and waits for the reader to become available.
5) Once the reader becomes available for the host, the CLF activates the reader for the host and the reader RF gates signals this via the event reader status.

A host may decide to always activate the reader status event and skip checking of the status in step 1. If the reader is available when the event reader requested is sent, the host will then receive the event reader status as a confirmation immediately.
10.4.3 Error management

Figure 17b illustrates the sequence the Host should follow in case the CLF reports an RF error in the response to a WR_XCHG_DATA command.

The sequence flow is as follows:

1) Assuming that the Host has triggered the reader RF gate by sending the EVT_READER_REQUESTED and that the CLF has notified the host via the event EVT_TARGET_DISCOVERED, the Host can send the WR_XCHG_DATA command.

2) The CLF detected a non-recoverable RF error and the reader RF gate replies in sending the response ANY_OK with parameter "Error indicator" set to '01'.

3) The Host stops the reader operation by sending the EVT_END_OPERATION.

4) The Host may want to restart the reader operation by sending the EVT_READER_REQUESTED.

11 Connectivity

11.1 Overview

This clause defines:

- A mechanism that allows a UICC to start a proactive session as defined in TS 102 223 [3] whenever this is required in the context of a contactless transaction.
- A mechanism for a host to launch an application in the terminal host that is related to the application running in the host.
- Commands and events needed by the terminal host in order to perform power management of the system.

From the host perspective these commands and events are addressed to the connectivity gate in the terminal host. However it is implementation dependent whether there is a real HCI to the terminal or whether it is implemented virtually in the CLF and proprietary mechanisms between the CLF and the terminal are used to trigger the sending of the command.

Support of the connectivity gate is optional for the terminal host.
11.2 Connectivity gate

The connectivity gate is a generic gate as defined in clause 7.2. As destination gate, the connectivity gate shall accept at least one pipe from each host in its WHITELIST.

Table 48 lists the GID for the connectivity gate.

<table>
<thead>
<tr>
<th>Gate</th>
<th>GID</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connectivity gate</td>
<td>'41'</td>
</tr>
</tbody>
</table>

The command and the events defined in the following clauses shall be sent to this gate.

11.2.1 Commands

11.2.1.1 PRO_HOST_REQUEST

This command allows a host to request the terminal host to activate during a limited time one or several other hosts in order to interact with them. This procedure shall not apply to request activation of the host controller or terminal host.

<table>
<thead>
<tr>
<th>Value</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>'10'</td>
<td>PRO_HOST_REQUEST</td>
</tr>
</tbody>
</table>

This command has following parameters:

<table>
<thead>
<tr>
<th>Description</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>Activation Duration (ms)</td>
<td>2</td>
</tr>
<tr>
<td>List of host identifiers</td>
<td>N</td>
</tr>
</tbody>
</table>

When successful for all requested hosts, the terminal host shall respond with ANY_OK with no parameters. The host shall not interact with another host after the expired activation duration time.

11.2.2 Events

The connectivity gate supports the events listed in table 51. The events shall be described in the following clauses.

<table>
<thead>
<tr>
<th>Value</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>'10'</td>
<td>EVT_CONNECTIVITY</td>
</tr>
<tr>
<td>'12'</td>
<td>EVT_TRANSACTION</td>
</tr>
<tr>
<td>'13'</td>
<td>EVT_OPERATION_ENDED</td>
</tr>
</tbody>
</table>

11.2.2.1 EVT_CONNECTIVITY

This event notifies the terminal host that it shall send a "HCI connectivity event" as defined in TS 102 223 [3].

This event has no parameters.
11.2.2.2 Void

11.2.2.3 EVT_OPERATION_ENDED

This event notifies the terminal host that the interaction with the hosts listed in the command PRO_HOST_REQUEST is ended before the activation duration time limit.

This event has no parameters.

11.2.2.4 EVT_TRANSACTION

This event notifies the terminal host that it shall launch an application associated to an NFC application in a UICC host identified by the AID on the parameter list.

NOTE: This method to launch an application does not replace the launch application supported by a proactive session as defined in TS 102 223 [3].

This event sends a list of parameters coded in BER-TLV fields as defined in TS 101 220 [9]. The tag and length of these fields are defined in the table 52.

<table>
<thead>
<tr>
<th>Description</th>
<th>Tag</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>AID</td>
<td>’81’</td>
<td>5 to 16</td>
</tr>
<tr>
<td>PARAMETERS</td>
<td>’82’</td>
<td>0 to 255</td>
</tr>
</tbody>
</table>

11.2.3 Registry

There are no registry entries defined.

11.3 Connectivity application gate

The following clauses define the commands and events that a connectivity application gate supports in addition to those of a generic gate as defined in clause 7.2.

The GID for the connectivity application gates are dynamically assigned by the host.

11.3.1 Commands

There are no additional commands.

11.3.2 Events

The connectivity application gate support the event listed in table 53. The event shall be described in the following clause.

<table>
<thead>
<tr>
<th>Value</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>’10’</td>
<td>EVT_STANDBY</td>
</tr>
</tbody>
</table>

11.3.2.1 EVT_STANDBY

This event should be used by the terminal host or the host controller to notify the host that it shall stop any ongoing communication with the other hosts and the host controller after 100 ms. The terminal host shall take care about not interrupting ongoing contactless transactions.

This event has no parameters.
11.3.3 Registry

There are no registry entries defined.

11.4 Procedures

The following procedures are described for the connectivity gate.

11.4.1 Use of connectivity gate

Figure 18 illustrates the messages between a UICC host and a terminal host using the EVT_CONNECTIVITY event. Same shall apply for other events.

![Diagram of message flow between UICC host, terminal host, and connectivity gate]

**Figure 18**

The sequence flow is as follows:

1) The UICC host requests the creation of a pipe to the connectivity gate in the terminal host.
2) The terminal host confirms as specified in clause 8.1.1.
3) UICC host opens the pipe.
4) Terminal host accepts the opening.
5) UICC sends the EVT_CONNECTIVITY to the terminal host.
Annex A (normative):
State diagram for contactless reader mode

Figure A.1 shows the state changes and related events for the operation of reader mode in the CLF.

Figure A.1
## Annex B (informative):
### Change history

This annex lists all Changes Requests (CR) applied to the present document.

<table>
<thead>
<tr>
<th>Date</th>
<th>Meeting</th>
<th>Plenary Doc</th>
<th>CR</th>
<th>Rev</th>
<th>Cat</th>
<th>Subject/Comment</th>
<th>Old</th>
<th>New</th>
</tr>
</thead>
<tbody>
<tr>
<td>2008-04</td>
<td>SCP #37</td>
<td>SCP-080214</td>
<td>001</td>
<td>F</td>
<td>Definition of the reaction to a reset of the underlying data link layer</td>
<td>7.0.0</td>
<td>7.1.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-080214</td>
<td>002</td>
<td>F</td>
<td>Correction and clarification of card emulation registry parameters</td>
<td>7.0.0</td>
<td>7.1.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-080214</td>
<td>003</td>
<td>F</td>
<td>Remove UID length 1 for reader mode</td>
<td>7.0.0</td>
<td>7.1.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-080214</td>
<td>004</td>
<td>D</td>
<td>Correction of the Fig 1, 14 and 16 in order to add consistency in the naming of the hosts.</td>
<td>7.0.0</td>
<td>7.1.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-080214</td>
<td>005</td>
<td>F</td>
<td>Clarification of the reader mode procedure</td>
<td>7.0.0</td>
<td>7.1.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-080214</td>
<td>006</td>
<td>F</td>
<td>Clarification of consequences of failure in identity check in lower layers and update of identity reference</td>
<td>7.0.0</td>
<td>7.1.0</td>
<td></td>
</tr>
<tr>
<td>2008-07</td>
<td>SCP #38</td>
<td>SCP-080318</td>
<td>008</td>
<td>D</td>
<td>Editorial corrections</td>
<td>7.1.0</td>
<td>7.2.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-080318</td>
<td>009</td>
<td>C</td>
<td>Clarification in Identity management gate</td>
<td>7.1.0</td>
<td>7.2.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-080318</td>
<td>010</td>
<td>F</td>
<td>Correction in the reader mode gate</td>
<td>7.1.0</td>
<td>7.2.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-080318</td>
<td>011</td>
<td>F</td>
<td>Clarification of connectivity</td>
<td>7.1.0</td>
<td>7.2.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-080318</td>
<td>012</td>
<td>C</td>
<td>Remove EVT_END_OF_TRANSACTION</td>
<td>7.1.0</td>
<td>7.2.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-080365</td>
<td>014</td>
<td>F</td>
<td>Clarification and correction of Type F card emulation</td>
<td>7.1.0</td>
<td>7.2.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-080366</td>
<td>015</td>
<td>D</td>
<td>Correction of spelling mistakes and improved wording</td>
<td>7.1.0</td>
<td>7.2.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-080318</td>
<td>017</td>
<td>F</td>
<td>Clarification of application level timeout in reader mode</td>
<td>7.1.0</td>
<td>7.2.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-080367</td>
<td>018</td>
<td>F</td>
<td>Correction of EVT_FIELD_OFF</td>
<td>7.1.0</td>
<td>7.2.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-080318</td>
<td>019</td>
<td>F</td>
<td>Modification of the Administration gate and the Link management gate</td>
<td>7.1.0</td>
<td>7.2.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-080318</td>
<td>020</td>
<td>F</td>
<td>Clarification on UID &amp; PUPI values</td>
<td>7.1.0</td>
<td>7.2.0</td>
<td></td>
</tr>
<tr>
<td>2008-10</td>
<td>SCP #39</td>
<td>SCP-080438</td>
<td>013</td>
<td>C</td>
<td>Remove &quot;battery on&quot; event</td>
<td>7.2.0</td>
<td>7.3.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-080432</td>
<td>016</td>
<td>F</td>
<td>Clarify behaviour for unknown commands and events</td>
<td>7.2.0</td>
<td>7.3.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-080432</td>
<td>021</td>
<td>C</td>
<td>Correction of table 2</td>
<td>7.2.0</td>
<td>7.3.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-080432</td>
<td>022</td>
<td>D</td>
<td>Editorial change to table markings</td>
<td>7.2.0</td>
<td>7.3.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-080432</td>
<td>023</td>
<td>D</td>
<td>Explanation of Figure 7</td>
<td>7.2.0</td>
<td>7.3.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-080432</td>
<td>024</td>
<td>F</td>
<td>Parameter length for AID in EVT TRANSACTION</td>
<td>7.2.0</td>
<td>7.3.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-080432</td>
<td>025</td>
<td>F</td>
<td>Clarification for EVT_POST_DATA</td>
<td>7.2.0</td>
<td>7.3.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-080432</td>
<td>026</td>
<td>F</td>
<td>Clarification of host for connectivity</td>
<td>7.2.0</td>
<td>7.3.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-080432</td>
<td>027</td>
<td>F</td>
<td>Change in the WR_XCHGDATA for the RF Reader gate</td>
<td>7.2.0</td>
<td>7.3.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-080432</td>
<td>028</td>
<td>C</td>
<td>Usage of the EVT_FIELD_ON event in the Card emulation procedure</td>
<td>7.2.0</td>
<td>7.3.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-080432</td>
<td>029</td>
<td>F</td>
<td>Extending parameters for type A and B card mode</td>
<td>7.2.0</td>
<td>7.3.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-080432</td>
<td>030</td>
<td>C</td>
<td>Correction of Table 16</td>
<td>7.2.0</td>
<td>7.3.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-080468</td>
<td>031</td>
<td>F</td>
<td>Addition of hot unplug</td>
<td>7.2.0</td>
<td>7.3.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-080483</td>
<td>033</td>
<td>C</td>
<td>Usage of the EVT_FIELD_ON event in the Card emulation procedure (addendum)</td>
<td>7.2.0</td>
<td>7.3.0</td>
<td></td>
</tr>
<tr>
<td>2009-01</td>
<td>SCP #40</td>
<td>SCP-090061</td>
<td>034</td>
<td>F</td>
<td>Correction of EVT_FIELD_ON</td>
<td>7.3.0</td>
<td>7.4.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-090061</td>
<td>035</td>
<td>F</td>
<td>Clarification of HCI registry default values</td>
<td>7.3.0</td>
<td>7.4.0</td>
<td></td>
</tr>
<tr>
<td>2009-05</td>
<td>SCP #41</td>
<td>SCP-090161</td>
<td>037</td>
<td>F</td>
<td>Correction of issues raised by STF</td>
<td>7.4.0</td>
<td>7.5.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-090159</td>
<td>036</td>
<td>F</td>
<td>Addition of a HCI end of operation event</td>
<td>7.4.0</td>
<td>7.5.0</td>
<td></td>
</tr>
<tr>
<td>2009-10</td>
<td>SCP #43</td>
<td>SCP-090331</td>
<td>041</td>
<td>F</td>
<td>Correction of CID support description</td>
<td>7.5.0</td>
<td>7.6.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-090331</td>
<td>040</td>
<td>F</td>
<td>Correction on use of EVT_END_OPERATION</td>
<td>7.5.0</td>
<td>7.6.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-090331</td>
<td>039</td>
<td>D</td>
<td>Several editorial corrections.</td>
<td>8.0.0</td>
<td>9.0.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCP-090356</td>
<td>038</td>
<td>D</td>
<td>Clarification on event and command reception</td>
<td>8.0.0</td>
<td>9.0.0</td>
<td></td>
</tr>
<tr>
<td>Date</td>
<td>Meeting</td>
<td>Plenary Doc</td>
<td>CR</td>
<td>Rev</td>
<td>Cat</td>
<td>Subject/Comment</td>
<td>Old</td>
<td>New</td>
</tr>
<tr>
<td>------------</td>
<td>---------</td>
<td>-------------</td>
<td>----</td>
<td>-----</td>
<td>-----</td>
<td>---------------------------------------------------------------------------------</td>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>2010-03</td>
<td>SCP#44</td>
<td>SCP(10)0027</td>
<td>044</td>
<td>-</td>
<td>A</td>
<td>Correction of ATQB Parameter length</td>
<td>8.0.0</td>
<td>9.0.0</td>
</tr>
<tr>
<td>2010-07</td>
<td>SCP#45</td>
<td>SCP(10)0135</td>
<td>047</td>
<td>-</td>
<td>A</td>
<td>Correction of contactless reader operation</td>
<td>9.0.0</td>
<td>9.1.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Definition of source gate assignment</td>
<td>9.0.0</td>
<td>9.1.0</td>
</tr>
<tr>
<td></td>
<td>SCP#45</td>
<td>SCP(10)0135</td>
<td>052</td>
<td>-</td>
<td>A</td>
<td>Definition of source gate assignment</td>
<td>9.0.0</td>
<td>9.1.0</td>
</tr>
<tr>
<td></td>
<td>SCP#45</td>
<td>SCP(10)0135</td>
<td>048</td>
<td>C</td>
<td>C</td>
<td>Correction of procedure for contactless card emulation</td>
<td>9.0.0</td>
<td>10.0.0</td>
</tr>
<tr>
<td>2010-10</td>
<td>SCP#46</td>
<td>SCP(10)0285r1</td>
<td>057</td>
<td>1</td>
<td>A</td>
<td>Clarification of EVT_HCI_END_OF_OPERATION</td>
<td>10.0.0</td>
<td>10.1.0</td>
</tr>
<tr>
<td>2010-10</td>
<td>SCP#46</td>
<td>SCP(10)0285r1</td>
<td>052</td>
<td></td>
<td></td>
<td>Clarification of contactless reader operation</td>
<td>10.0.0</td>
<td>9.1.0</td>
</tr>
<tr>
<td>2011-03</td>
<td>SCP#48</td>
<td>SCP(11)0158</td>
<td>065</td>
<td>-</td>
<td>A</td>
<td>Correction of a reference to the ISO/IEC 14443 specification</td>
<td>10.2.0</td>
<td>10.3.0</td>
</tr>
<tr>
<td>2011-03</td>
<td>SCP#48</td>
<td>SCP(11)0154</td>
<td>069</td>
<td>A</td>
<td>A</td>
<td>Processing of RF error indicator in card emulation mode</td>
<td>10.2.0</td>
<td>10.3.0</td>
</tr>
<tr>
<td>2011-05</td>
<td>SCP#49</td>
<td>SCP(11)0199</td>
<td>070</td>
<td>D</td>
<td>D</td>
<td>Clarification of the use of WR_XCHG_DATA</td>
<td>10.3.0</td>
<td>11.0.0</td>
</tr>
<tr>
<td>2011-09</td>
<td>SCP#52</td>
<td>SCP(11)0287r1</td>
<td>071</td>
<td>1</td>
<td>C</td>
<td>CR 102 622 R11 #071r1: Clarification of error response after timeout for WR_XCHG_DATA command</td>
<td>11.0.0</td>
<td>11.1.0</td>
</tr>
<tr>
<td>2011-09</td>
<td>SCP#52</td>
<td>SCP(11)0288r1</td>
<td>072</td>
<td>-</td>
<td>C</td>
<td>CR 102 622 R11 #072: Multiple pipes to one gate</td>
<td>11.0.0</td>
<td>11.1.0</td>
</tr>
<tr>
<td>2012-03</td>
<td>SCP#54</td>
<td>SCP(12)000014</td>
<td>073</td>
<td>B</td>
<td>B</td>
<td>CR 102 622 R11 #073: Support of power indication for contactless interface</td>
<td>11.0.0</td>
<td>11.1.0</td>
</tr>
<tr>
<td>2012-06</td>
<td>SCP#55</td>
<td>SCP(12)000124</td>
<td>074</td>
<td>-</td>
<td>B</td>
<td>CR 102 622 R11 #074: Addition of reader gate status indications and events</td>
<td>11.0.0</td>
<td>11.1.0</td>
</tr>
<tr>
<td>2012-12</td>
<td>SCP#57</td>
<td>SCP(12)000270</td>
<td>075</td>
<td>D</td>
<td>D</td>
<td>Correction of REC_ERROR for host link management gate</td>
<td>11.1.0</td>
<td>11.2.0</td>
</tr>
<tr>
<td>2012-12</td>
<td>SCP#57</td>
<td>SCP(12)000271</td>
<td>076</td>
<td>-</td>
<td>D</td>
<td>Editorial correction in the Card emulation architecture figure</td>
<td>11.1.0</td>
<td>11.2.0</td>
</tr>
<tr>
<td>2012-12</td>
<td>SCP#57</td>
<td>SCP(12)000272</td>
<td>077</td>
<td>-</td>
<td>F</td>
<td>Correction of reader gate status indications and events</td>
<td>11.1.0</td>
<td>11.2.0</td>
</tr>
<tr>
<td>2012-12</td>
<td>SCP#57</td>
<td>SCP(12)000274</td>
<td>078</td>
<td>F</td>
<td>F</td>
<td>Clarification of the RF errors handling in reader mode</td>
<td>11.1.0</td>
<td>11.2.0</td>
</tr>
<tr>
<td>2013-07</td>
<td>SCP#60</td>
<td>SCP(13)000150</td>
<td>081</td>
<td>-</td>
<td>C</td>
<td>CR 102 622 R12 #081: Clarification of Type F no response case</td>
<td>11.2.0</td>
<td>12.0.0</td>
</tr>
</tbody>
</table>
## History

<table>
<thead>
<tr>
<th>Document history</th>
</tr>
</thead>
<tbody>
<tr>
<td>V12.0.0</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>