



# **NFC Digital Protocol**

Technical Specification

NFC Forum™

DIGITAL 1.0

NFCForum-TS-DigitalProtocol-1.0

2010-11-17

## **RESTRICTIONS ON USE**

This specification is copyright © 2005-2010 by the NFC Forum, and was made available pursuant to a license agreement entered into between the recipient (Licensee) and NFC Forum, Inc. (Licensor) and may be used only by Licensee, and in compliance with the terms of that license agreement (License). If you are not the Licensee, you may read this Specification, but are not authorized to implement or make any other use of this specification. However, you may obtain a copy of this Specification and implementation rights at the following page of Licensor's website: [http://www.nfc-forum.org/specs/spec\\_license](http://www.nfc-forum.org/specs/spec_license) after entering into and agreeing to such license terms as Licensor is then requiring. On the date that this specification was downloaded by Licensee, the non-implementation terms of that license were as follows:

### **1. LICENSE GRANT.**

Licensor hereby grants Licensee the right, without charge, to copy (for internal purposes only) and share this Specification with Licensee's members, employees and (to the extent related to Licensee's use of this Specification) consultants. This license grant does not include the right to sublicense, modify or create derivative works based upon the Specification.

### **2. NO WARRANTIES.**

THE SPECIFICATION IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, ACCURACY, COMPLETENESS AND NON-INFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL LICENSOR, ITS MEMBERS OR ITS CONTRIBUTORS BE LIABLE FOR ANY CLAIM, OR ANY DIRECT, SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THE SPECIFICATION.

### **3. THIRD PARTY RIGHTS.**

Without limiting the generality of Section 2 above, LICENSOR ASSUMES NO RESPONSIBILITY TO COMPILE, CONFIRM, UPDATE OR MAKE PUBLIC ANY THIRD PARTY ASSERTIONS OF PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS THAT MIGHT NOW OR IN THE FUTURE BE INFRINGED BY AN IMPLEMENTATION OF THE SPECIFICATION IN ITS CURRENT, OR IN ANY FUTURE FORM. IF ANY SUCH RIGHTS ARE DESCRIBED ON THE SPECIFICATION, LICENSOR TAKES NO POSITION AS TO THE VALIDITY OR INVALIDITY OF SUCH ASSERTIONS, OR THAT ALL SUCH ASSERTIONS THAT HAVE OR MAY BE MADE ARE SO LISTED.

### **4. TERMINATION OF LICENSE.**

In the event of a breach of this Agreement by Licensee or any of its employees or members, Licensor shall give Licensee written notice and an opportunity to cure. If the breach is not cured within thirty (30) days after written notice, or if the breach is of a nature that cannot be cured, then Licensor may immediately or thereafter terminate the licenses granted in this Agreement.

### **5. MISCELLANEOUS.**

All notices required under this Agreement shall be in writing, and shall be deemed effective five days from deposit in the mails. Notices and correspondence to the NFC Forum address as it appears below. This Agreement shall be construed and interpreted under the internal laws of the United States and the Commonwealth of Massachusetts, without giving effect to its principles of conflict of law.

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

# Contents

<b>1</b>	<b>Introduction.....</b>	<b>1</b>
1.1	Scope.....	1
1.2	Audience.....	1
1.3	Applicable Documents or References .....	1
1.4	Administration.....	3
1.5	Name and Logo Usage .....	3
1.6	Intellectual Property .....	4
1.7	Acknowledgements .....	4
1.8	Special Word Usage .....	4
1.9	Requirement Numbering .....	4
1.10	Implementation of Optional Items.....	5
1.11	Notational Conventions .....	5
1.11.1	Notations.....	5
1.11.2	Value of Parameters.....	5
1.12	Abbreviations .....	6
1.13	Glossary.....	8
1.13.1	Device and Communication.....	8
1.13.2	Protocol and Mode.....	10
1.13.3	Errors .....	10
<b>2</b>	<b>Overview .....</b>	<b>12</b>
<b>3</b>	<b>Bit Duration.....</b>	<b>14</b>
<b>4</b>	<b>NFC-A Technology .....</b>	<b>15</b>
4.1	Sequence Format .....	15
4.1.1	Poll→Listen Modulation .....	15
4.1.2	Listen→Poll Modulation .....	17
4.1.3	Synchronization .....	18
4.1.4	De-synchronization.....	19
4.2	Bit Level Coding .....	19
4.2.1	Poll→Listen Coding Scheme.....	19
4.2.2	Listen→Poll Coding Scheme.....	21
4.3	Frame Format .....	21
4.3.2	Short Frame .....	22
4.3.3	Standard Frame .....	22
4.3.4	Bit Oriented SDD Frame .....	23
4.4	Data and Payload Format .....	24
4.5	Command Set .....	25
4.6	ALL_REQ and SENS_REQ.....	25
4.6.1	ALL_REQ Command.....	26
4.6.2	SENS_REQ Command.....	26
4.6.3	SENS_RES Response.....	26
4.7	SDD_REQ.....	27
4.7.1	SDD_REQ Command.....	28
4.7.2	SDD_RES Response.....	31
4.8	SEL_REQ.....	34
4.8.1	SEL_REQ Command.....	34
4.8.2	SEL_RES Response.....	35
4.9	SLP_REQ.....	36

4.9.1	SLP_REQ Command.....	36
4.9.2	SLP_REQ Response .....	37
4.10	Timing Requirements .....	37
4.10.1	Frame Delay Time Poll→Listen.....	37
4.10.2	Frame Delay Time Listen→Poll.....	40
4.10.3	Guard Time.....	42
<b>5</b>	<b>NFC-B Technology .....</b>	<b>43</b>
5.1	Sequence Format .....	43
5.1.1	Poll→Listen Modulation .....	43
5.1.2	Listen→Poll Modulation .....	44
5.1.3	Synchronization .....	45
5.1.4	Pattern Synchronization.....	48
5.1.5	De-synchronization.....	48
5.2	Bit Level Coding .....	50
5.2.1	Poll→Listen Coding Scheme.....	50
5.2.2	Listen→Poll Coding Scheme.....	50
5.3	Frame Format .....	51
5.4	Data and Payload Format .....	51
5.5	Command Set .....	52
5.6	ALLB_REQ and SENSB_REQ .....	52
5.6.1	ALLB_REQ and SENSB_REQ Command .....	52
5.6.2	SENSB_RES Response .....	55
5.7	SLOT_MARKER.....	62
5.7.1	SLOT_MARKER Command.....	62
5.7.2	SLOT_MARKER Response .....	63
5.8	SLPB_REQ .....	63
5.8.1	SLPB_REQ Command .....	63
5.8.2	SLPB_RES Response .....	63
5.9	Timing Requirements .....	63
5.9.1	Frame Delay Time Poll→Listen.....	63
5.9.2	Frame Delay Time Listen→Poll.....	66
5.9.3	Guard Time.....	68
<b>6</b>	<b>NFC-F Technology.....</b>	<b>70</b>
6.1	Sequence Format .....	70
6.1.1	Modulation.....	70
6.1.2	Synchronization .....	71
6.1.3	De-synchronization.....	73
6.2	Bit Level Coding .....	73
6.3	Frame Format .....	73
6.4	Data and Payload Format .....	74
6.5	Command Set .....	75
6.6	SENSF_REQ .....	75
6.6.1	SENSF_REQ Command.....	75
6.6.2	SENSF_RES Response.....	78
6.7	Timing Requirements .....	83
6.7.1	Frame Delay Time Poll→Listen.....	83
6.7.2	Frame Delay Time Listen→Poll.....	85
6.7.3	Guard Time.....	87

<b>7</b>	<b>Half-duplex Protocols .....</b>	<b>88</b>
<b>8</b>	<b>Type 1 Tag Platform .....</b>	<b>89</b>
8.1	Sequence Format .....	89
8.2	Bit Level Coding .....	89
8.3	Frame Format .....	89
8.4	Data and Payload Format .....	90
8.5	Command Set .....	91
8.6	Read Identifier (RID) .....	92
8.6.1	RID Command.....	92
8.6.2	RID Response .....	93
8.7	Timing Requirements .....	93
8.7.1	Reader-Reader Data Delay .....	93
8.7.2	Frame Delay Time .....	94
<b>9</b>	<b>Type 2 Tag Platform .....</b>	<b>96</b>
9.1	Sequence Format .....	96
9.2	Bit Level Coding .....	96
9.3	Frame Format .....	96
9.4	Data and Payload Format .....	97
9.5	Command Set .....	98
9.6	READ .....	98
9.6.1	Command.....	98
9.6.2	Response .....	98
9.7	WRITE .....	99
9.7.1	Command.....	99
9.7.2	Response .....	99
9.8	SECTOR SELECT .....	100
9.8.1	SECTOR SELECT Command Packet 1 .....	100
9.8.2	SECTOR SELECT Command Packet 2 .....	100
9.8.3	Response .....	100
9.9	Timing Requirements .....	102
<b>10</b>	<b>Type 3 Tag Platform .....</b>	<b>103</b>
10.1	Sequence Format .....	103
10.2	Bit Level Coding .....	103
10.3	Frame Format .....	103
10.4	Data and Payload Format .....	103
10.5	Command Set .....	104
10.6	Timing Requirements .....	104
<b>11</b>	<b>Type 4A Tag Platform.....</b>	<b>106</b>
11.1	Sequence Format .....	106
11.2	Bit Level Coding .....	106
11.3	Frame Format .....	106
11.4	Data and Payload Format .....	106
11.5	Command Set .....	107
11.6	Request for Answer to Select (RATS) .....	107
11.6.1	RATS Command.....	107
11.6.2	RATS Response (Answer To Select).....	109
11.7	Timing Requirements .....	115
11.7.1	FWT.....	115

11.7.2	SFGT.....	117
<b>12</b>	<b>Type 4B Tag Platform.....</b>	<b>118</b>
12.1	Sequence Format .....	118
12.2	Bit Level Coding .....	118
12.3	Frame Format .....	118
12.4	Data and Payload Format .....	118
12.5	Command Set .....	119
12.6	ATTRIB.....	119
12.6.1	ATTRIB Command .....	119
12.6.2	ATTRIB Response.....	125
12.7	Timing Requirements .....	127
<b>13</b>	<b>ISO-DEP Protocol .....</b>	<b>128</b>
13.1	Block Format.....	128
13.1.1	Block.....	128
13.1.2	SoD .....	129
13.1.3	Payload .....	131
13.1.4	EoD.....	131
13.2	Protocol Operation.....	132
13.2.1	General Rules.....	132
13.2.2	Frame Waiting Time Extension.....	132
13.2.3	Chaining.....	135
13.2.4	Block Numbering Rules.....	135
13.2.5	Block Handling Rules.....	136
13.2.6	Exception Processing.....	138
13.2.7	De-activation Rules.....	140
13.3	Timing Requirements .....	142
<b>14</b>	<b>NFC-DEP Protocol.....</b>	<b>143</b>
14.1	Sequence Format .....	143
14.2	Bit Level Coding .....	143
14.3	Frame Format .....	144
14.4	Data and Payload Format .....	144
14.5	Command Set .....	145
14.6	Attribute Request (ATR_REQ) .....	146
14.6.1	ATR_REQ Length.....	146
14.6.2	ATR_REQ Command.....	146
14.6.3	ATR_RES Response.....	149
14.7	Parameter Selection Request (PSL_REQ).....	152
14.7.1	PSL_REQ Command.....	152
14.7.2	PSL_RES Response.....	154
14.8	Data Exchange Protocol Request (DEP_REQ) .....	154
14.8.1	DEP_REQ Command .....	154
14.8.2	DEP_RES Response .....	155
14.8.3	Protocol Format Byte (PFB).....	156
14.8.4	Response Timeout Extension.....	157
14.9	Deselect Request (DSL_REQ) .....	159
14.9.1	DSL_REQ Command .....	160
14.9.2	DSL_RES Response .....	160
14.10	Release Request (RLS_REQ).....	161
14.10.1	RLS_REQ Command .....	161

14.10.2	RLS_RES Response .....	162
14.11	Timing Requirements .....	163
14.12	NFC-DEP Protocol Operation .....	165
14.12.1	General Rules.....	165
14.12.2	Chaining.....	165
14.12.3	PDU Numbering Rules .....	166
14.12.4	PDU Handling Rules .....	167
14.12.5	Exception Processing.....	168
<b>A.</b>	<b>Values.....</b>	<b>170</b>
A.1	NFC-A Technology .....	170
A.2	NFC-B Technology .....	171
A.3	NFC-F Technology.....	172
A.4	Type 1 Tag Platform.....	172
A.5	Type 2 Tag Platform.....	172
A.6	Type 3 Tag Platform.....	173
A.7	Type 4A Tag Platform.....	173
A.8	Type 4B Tag Platform.....	173
A.9	ISO-DEP Protocol .....	174
A.10	NFC-DEP Protocol.....	174
<b>B.</b>	<b>Revision History .....</b>	<b>175</b>

## Figures

Figure 1: Modified Miller Coding with ASK 100%.....	15
Figure 2: Manchester Coding with OOK .....	17
Figure 3: Short Frame.....	22
Figure 4: Standard Frame (Poll→Listen Communication) .....	22
Figure 5: Bit Oriented SDD Frame (with Split after the First Bit of the Second Byte) .....	23
Figure 6: Data and Payload Format – NFC-A Standard Frame.....	24
Figure 7: Example: SDD_REQ with 14 Data Bits of NFCID1 CL1 (4 Bytes Size) and SDD_RES .....	33
Figure 8: $FDT_{A,LISTEN}$ .....	38
Figure 9: $t_{nn,min}$ for NFC-A.....	40
Figure 10: $FDT_{A,POLL}$ .....	41
Figure 11: NRZ-L Coding.....	43
Figure 12: NRZ-L Coding with BPSK.....	44
Figure 13: Synchronization and Timing Parameters between a Listen Frame and a Poll Frame..	45
Figure 14: Synchronization and Timing Parameters between a Poll Frame and a Listen Frame..	46
Figure 15: NFC-B – Character Format.....	51
Figure 16: NFC-B – Frame Format .....	51
Figure 17: Data and Payload Format – NFC-B .....	52
Figure 18: $t_{nn,min}$ for NFC-B .....	66
Figure 19: Manchester Coding with ASK Modulation .....	70
Figure 20: Signal Synchronization and Timing Parameters.....	72
Figure 21: NFC-F – Character Format .....	74
Figure 22: NFC-F – Frame Format .....	74
Figure 23: Data and Payload Format – NFC-F.....	74
Figure 24: Collision Resolution (Example).....	78
Figure 25: Frame Format – Type 1 Tag Platform in Poll Mode.....	89
Figure 26: Data and Payload Format – Type 1 Tag .....	90
Figure 27: Reader-Reader Data Delay (RRDD).....	94
Figure 28: Data and Payload Format – Type 2 Tag (except for ACK and NACK Response).....	97
Figure 29: Data and Payload Format – Type 4A Tag.....	107
Figure 30: Block Format .....	128
Figure 31: Data and Payload Format – NFC-DEP Protocol.....	144



## Tables

Table 1: Sample Requirement .....	4
Table 2: Notational Conventions.....	5
Table 3: Activities versus Technology / Device Platform.....	13
Table 4: NFC-A – Command Set .....	25
Table 5: Format of ALL_REQ .....	26
Table 6: Format of SENS_REQ .....	26
Table 7: Byte 1 of SENS_RES.....	26
Table 8: Byte 2 of SENS_RES.....	27
Table 9: Format of SDD_REQ Command .....	28
Table 10: Format of SEL_CMD.....	28
Table 11: Format of SEL_PAR (Upper 4 Bits).....	28
Table 12: Format of SEL_PAR (Lower 4 Bits).....	29
Table 13: SDD_RES Response (NFCID1 CLn + BCC) .....	31
Table 14: nfcid1 <sub>0</sub> for Single-size NFCID1 .....	31
Table 15: Format of SEL_REQ Command .....	34
Table 16: NFCID1 CL <sub>n</sub> .....	35
Table 17: Format of SEL_RES Response .....	35
Table 18: Format of SLP_REQ Command.....	36
Table 19: FDT <sub>A,LISTEN</sub> and Logic State of Last Data Bit .....	38
Table 20: FDT <sub>A,LISTEN,MIN</sub> and Command Type .....	38
Table 21: NFC-B – Command Set .....	52
Table 22: ALLB_REQ and SENSB_REQ Command Format .....	53
Table 23: Format of PARAM Byte Included in ALLB_REQ and SENSB_REQ Command .....	53
Table 24: Coding of N.....	54
Table 25: SENSB_RES Format.....	55
Table 26: Application Data Format .....	56
Table 27: Protocol Info Format .....	56
Table 28: Bit Rates Supported by the NFC Forum Device in Listen Mode.....	57
Table 29: FSCI to FSC Conversion.....	58
Table 30: Protocol_Type .....	59
Table 31: Minimum TR2 Coding.....	59
Table 32: ADC Coding.....	61
Table 33: FO – NAD .....	61

Table 34: FO – DID.....	61
Table 35: SLOT_MARKER Command Format.....	62
Table 36: Coding of Slot Number .....	62
Table 37: Format of the SLPB_REQ Command.....	63
Table 38: SLPB_RES Response Format .....	63
Table 39: NFC-F – Command Set.....	75
Table 40: SENSF_REQ Command Format.....	76
Table 41: Coding of RC .....	76
Table 42: Coding of TSN .....	78
Table 43: SENSF_RES Format .....	79
Table 44: NFCID2 Format .....	79
Table 45: RD Format Advanced Protocol Features (Byte 18) .....	82
Table 46: RD Format Advanced Protocol Features (Byte 19) .....	83
Table 47: Command Set.....	92
Table 48: RID Command Format.....	92
Table 49: RID Response Format .....	93
Table 50: Command Set.....	98
Table 51: READ Command Format.....	98
Table 52: READ Response Format .....	99
Table 53: NACK Response Format.....	99
Table 54: WRITE Command Format .....	99
Table 55: ACK Response Format.....	100
Table 56: SECTOR SELECT Command Packet 1 Format .....	100
Table 57: SECTOR SELECT Command Packet 2 Format .....	100
Table 58: Command Set.....	104
Table 59: Format of MRTI <sub>CHECK</sub> and MRTI <sub>UPDATE</sub> .....	105
Table 60: Command Set.....	107
Table 61: Format of RATS Command .....	107
Table 62: Format of RATS Parameter Byte (PARAM) .....	108
Table 63: FSDI to FSD Conversion .....	108
Table 64: Structure of the ATS .....	109
Table 65: Coding of Format Byte T0 .....	110
Table 66: FSCI to FSC Conversion.....	111
Table 67: Coding of Interface Byte TA(1).....	112
Table 68: Coding of Interface Byte TB(1) .....	113

Table 69: Coding of Interface Byte TC(1) .....	115
Table 70: Command Set .....	119
Table 71: ATTRIB Command Format .....	119
Table 72: Format of Param 1 of the ATTRIB Command.....	120
Table 73: Format of Param 2 of the ATTRIB Command.....	122
Table 74: FSDI to FSD Conversion .....	122
Table 75: Coding of b8 and b7 of Param 2 .....	123
Table 76: Coding of b6 and b5 of Param 2 .....	123
Table 77: Format of Param 3 of the ATTRIB Command.....	124
Table 78: Format of Param 4 of the ATTRIB Command.....	125
Table 79: ATTRIB Response Format.....	125
Table 80: Format of I-block PCB .....	129
Table 81: Format of R-block PCB.....	129
Table 82: Format of S-block PCB .....	130
Table 83: Format of DID Field.....	130
Table 84: Format of INF Field of an S(WTX) Request .....	132
Table 85: Format of INF Field of an S(WTX) Response .....	133
Table 86: NFC-DEP Protocol - Command Set.....	145
Table 87: ATR_REQ Format .....	146
Table 88: Bit Rates Supported by Initiator in Sending Direction ( $BS_I$ ) .....	147
Table 89: Bit Rates Supported by Initiator in Receiving Direction ( $BR_I$ ).....	147
Table 90: $PP_I$ Format .....	148
Table 91: $LR_I$ Coding .....	148
Table 92: ATR_RES Format .....	149
Table 93: Bit Rates Supported by Target in Sending Direction ( $BS_T$ ) .....	150
Table 94: Bit Rates Supported by Target in Receiving Direction ( $BR_T$ ) .....	150
Table 95: TO Format .....	150
Table 96: $PP_T$ Format .....	151
Table 97: $LR_T$ Coding .....	151
Table 98: Format of PSL_REQ Command.....	152
Table 99: Format of BRS .....	153
Table 100: Coding of DSI and DRI.....	153
Table 101: Format of FSL .....	153
Table 102: Format of the PSL_RES Response.....	154
Table 103: Format of the DEP_REQ Command.....	155

Table 104: Format of the DEP_RES Response .....	155
Table 105: PFB Coding of Information PDU .....	157
Table 106: PFB Coding of ACK/NACK PDU .....	157
Table 107: PFB Coding of Supervisory PDU .....	157
Table 108: Coding of Transport Data Byte of an RTOX Request .....	158
Table 109: Format of Transport Data Byte of an RTOX Response .....	158
Table 110: Format of the DSL_REQ Command .....	160
Table 111: Format of the DSL_RES Response .....	160
Table 112: Format of the RLS_REQ Command .....	161
Table 113: Format of the RLS_RES Response .....	162
Table 114: NFC-A Technology Poll Mode and Listen Mode Parameter Values .....	170
Table 115: NFC-B Technology Poll Mode and Listen Mode Parameter Values .....	171
Table 116: NFC-F Technology Poll Mode and Listen Mode Parameter Values .....	172
Table 117: Type 1 Tag Platform Poll Mode and Listen Mode Parameter Values .....	172
Table 118: Type 2 Tag Platform Poll Mode and Listen Mode Parameter Values .....	172
Table 119: Type 3 Tag Platform Poll Mode and Listen Mode Parameter Values .....	173
Table 120: Type 4A Tag Platform Poll Mode and Listen Mode Parameter Values .....	173
Table 121: Type 4B Tag Platform Poll Mode and Listen Mode Parameter Values .....	173
Table 122: ISO-DEP Protocol Poll Mode and Listen Mode Parameter Values .....	174
Table 123: NFC-DEP Protocol Poll Mode and Listen Mode Parameter Values .....	174
Table 124: Revision History .....	175

# Requirements

Requirements 1: Implementation of Optional Items .....	5
Requirements 2: Reserved for Future Use.....	6
Requirements 3: Signal Patterns Poll→Listen – NFC-A .....	16
Requirements 4: Modulation Listen→Poll – NFC-A.....	17
Requirements 5: Signal Patterns Listen→Poll – NFC-A .....	18
Requirements 6: De-synchronization Poll→Listen – NFC-A.....	19
Requirements 7: De-synchronization Listen→Poll – NFC-A .....	19
Requirements 8: Bit Level Coding Poll→Listen – NFC-A.....	20
Requirements 9: Bit Level Coding Listen→Poll – NFC-A.....	21
Requirements 10: Frame Format – NFC-A Technology .....	22
Requirements 11: Short Frame Format – NFC-A Technology .....	22
Requirements 12: Standard Frame Format – NFC-A Technology .....	23
Requirements 13: Bit Oriented SDD Frame Format .....	24
Requirements 14: Bit Oriented SDD Frame Format (Continued).....	24
Requirements 15: Data and Payload Format – NFC-A .....	25
Requirements 16: NFC-A – Command Set .....	25
Requirements 17: NFCID1 Length .....	27
Requirements 18: Type 1 Tag Platform Configuration.....	27
Requirements 19: Format of SDD_REQ Command .....	30
Requirements 20: NFCID1 .....	32
Requirements 21: Format of SDD_RES Response (Complete and Incomplete NFCID1).....	32
Requirements 22: Format of SDD_RES Response (Complete NFCID1 Only) .....	32
Requirements 23: Format of SDD_RES Response (Incomplete NFCID1 Only).....	34
Requirements 24: SEL_RES Response .....	36
Requirements 25: SLP_REQ Response.....	37
Requirements 26: $FDT_{A,LISTEN}$ .....	39
Requirements 27: Tolerance.....	39
Requirements 28: $FDT_{A,LISTEN,MIN}$ .....	39
Requirements 29: $t_{nn,min}$ - NFC-A .....	40
Requirements 30: $FDT_{A,POLL,MIN}$ .....	41
Requirements 31: $FDT_{A,REACTIVATION}$ .....	42
Requirements 32: Guard Time – NFC-A .....	42
Requirements 33: Signal Patterns Poll→Listen – NFC-B.....	44

Requirements 34: Signal Patterns Listen→Poll – NFC-B.....	45
Requirements 35: Synchronization Poll→Listen – NFC-B.....	46
Requirements 36: Synchronization Listen→Poll – NFC-B.....	47
Requirements 37: NFC-B Pattern Group Separation .....	48
Requirements 38: NFC-B Pattern Boundaries.....	48
Requirements 39: De-synchronization Poll→Listen – NFC-B .....	49
Requirements 40: De-synchronization Listen→Poll – NFC-B .....	49
Requirements 41: Bit Level Coding Poll→Listen – NFC-B .....	50
Requirements 42: Bit Level Coding Listen→Poll – NFC-B .....	50
Requirements 43: Character Format – NFC-B Technology.....	51
Requirements 44: Frame Format – NFC-B Technology .....	51
Requirements 45: Data and Payload Format – NFC-B .....	52
Requirements 46: AFI .....	53
Requirements 47: Number of Slots (N).....	54
Requirements 48: Listen Mode Handling of RFU Values of N .....	54
Requirements 49: Support for Extended SENSB_RES.....	55
Requirements 50: NFCID0 in SENSB_RES .....	55
Requirements 51: Bit Rates Supported by the NFC Forum Device in Listen Mode.....	57
Requirements 52: Poll Mode Handling of RFU Value of b4 of Bit_Rate_Capability .....	57
Requirements 53: FSC.....	58
Requirements 54: FSCI .....	58
Requirements 55: Poll Mode Handling of RFU Values of FSCI .....	58
Requirements 56: Protocol Type Supported by the NFC Forum Device in Listen Mode.....	59
Requirements 57: Minimum TR2 Coding.....	60
Requirements 58: Poll Mode Handling of RFU Value of b4 of Protocol_Type .....	60
Requirements 59: Maximum Value of FWI.....	60
Requirements 60: Poll Mode Handling of RFU Value of FWI.....	60
Requirements 61: NAD .....	61
Requirements 62: SFGI.....	62
Requirements 63: Poll Mode Handling of RFU Value of SFGI.....	62
Requirements 64: $FDT_{B,LISTEN,MIN}$ .....	64
Requirements 65: Frame Waiting Time .....	65
Requirements 66: $t_{nn,min}$ - NFC-B .....	66
Requirements 67: $FDT_{B,POLL,MIN}$ .....	67

Requirements 68: SFGT – NFC-B .....	68
Requirements 69: $FDT_{B,REACTIVATION}$ .....	68
Requirements 70: Guard Time – NFC-B.....	69
Requirements 71: Creation of Signal Patterns – NFC-F .....	71
Requirements 72: Reading of Signal Patterns – NFC-F.....	71
Requirements 73: NFC-F Transition Boundaries.....	71
Requirements 74: Synchronization - NFC-F Sequence.....	72
Requirements 75: De-synchronization - NFC-F.....	73
Requirements 76: Bit Level Coding – NFC-F.....	73
Requirements 77: Bit Level Decoding – NFC-F.....	73
Requirements 78: NFC-F – Frame Format.....	74
Requirements 79: Data and Payload Format – NFC-F.....	75
Requirements 80: System Code (SC).....	76
Requirements 81: Request Code (RC) .....	77
Requirements 82: Time Slot Number (TSN).....	78
Requirements 83: NFCID2 in SENSEF_RES .....	79
Requirements 84: PAD0.....	79
Requirements 85: PAD1.....	80
Requirements 86: $MRTI_{CHECK}$ .....	80
Requirements 87: $MRTI_{UPDATE}$ .....	81
Requirements 88: PAD2.....	81
Requirements 89: Request Data (RD) .....	82
Requirements 90: $FDT_{F,LISTEN,MIN}$ (except SENSEF_REQ).....	84
Requirements 91: $FDT_{F,LISTEN,SENSEF_REQ}$ .....	85
Requirements 92: $FDT_{F,POLL,MIN}$ .....	86
Requirements 93: $FDT_{F,REACTIVATION}$ .....	86
Requirements 94: Guard Time – NFC-F.....	87
Requirements 95: General Rules for Half-duplex Transmission Protocol.....	88
Requirements 96: Sequence Format – Type 1 Tag Platform.....	89
Requirements 97: Bit Level Coding – Type 1 Tag Platform.....	89
Requirements 98: Frame Format – Type 1 Tag Platform.....	90
Requirements 99: Data and Payload Format – Type 1 Tag.....	91
Requirements 100: RID Command .....	92
Requirements 101: RID Response.....	93
Requirements 102: Reader-Reader Data Delay (RRDD).....	94

Requirements 103: Frame Delay Times – Type 1 Tag Platform.....	95
Requirements 104: Sequence Format – Type 2 Tag Platform.....	96
Requirements 105: Bit Level Coding – Type 2 Tag Platform.....	96
Requirements 106: Frame Format – Type 2 Tag Platform.....	96
Requirements 107: Frame Format – ACK and NACK Response .....	97
Requirements 108: Data and Payload Format – Type 2 Tag.....	97
Requirements 109: READ Response.....	98
Requirements 110: NACK Response .....	99
Requirements 111: WRITE Response.....	99
Requirements 112: SECTOR SELECT Command Packet 1 Response .....	101
Requirements 113: Passive ACK Response .....	101
Requirements 114: Frame Delay Times – Type 2 Tag Platform.....	102
Requirements 115: Sequence Format – Type 3 Tag Platform.....	103
Requirements 116: Bit Level Coding – Type 3 Tag Platform.....	103
Requirements 117: Frame Format – Type 3 Tag Platform.....	103
Requirements 118: Data and Payload Format – Type 3 Tag.....	103
Requirements 119: MRT <sub>CHECK</sub> .....	105
Requirements 120: MRT <sub>UPDATE</sub> .....	105
Requirements 121: Sequence Format – Type 4A Tag Platform.....	106
Requirements 122: Bit Level Coding – Type 4A Tag Platform.....	106
Requirements 123: Frame Format – Type 4A Tag Platform.....	106
Requirements 124: Data and Payload Format – Type 4A Tag.....	107
Requirements 125: Frame Size for NFC Forum Device in Poll Mode (FSD).....	108
Requirements 126: FSDI <sub>MIN</sub> .....	108
Requirements 127: Listen Mode Handling of RFU Values of FSDI.....	109
Requirements 128: Support of DID.....	109
Requirements 129: Listen Mode Handling of RFU Value of DID.....	109
Requirements 130: Length Byte TL of the ATS .....	110
Requirements 131: FSC.....	111
Requirements 132: FSCI .....	111
Requirements 133: Poll Mode Handling of RFU Values of FSCI .....	111
Requirements 134: Format Byte T0 of the ATS.....	112
Requirements 135: Format Byte TA(1) of the ATS .....	113
Requirements 136: Poll Mode Handling of RFU Value of b4 of Interface Byte TA(1) .....	113
Requirements 137: Interface Byte TB(1) of the ATS.....	114



Requirements 138: Poll Mode Handling of RFU Value of FWI.....	114
Requirements 139: Interface Byte TB(1) of the ATS.....	114
Requirements 140: Poll Mode Handling of RFU value of SFGI .....	114
Requirements 141: NAD .....	115
Requirements 142: Historical Bytes of the ATS .....	115
Requirements 143: Frame Timing – Type 4A Tag Platform.....	116
Requirements 144: SFGT – Type 4A Tag Platform.....	117
Requirements 145: Sequence Format – Type 4B Tag Platform.....	118
Requirements 146: Bit Level Coding – Type 4B Tag Platform.....	118
Requirements 147: Frame Format – Type 4B Tag Platform.....	118
Requirements 148: Data and Payload Format – Type 4B Tag Platform.....	118
Requirements 149: NFCID0 in ATTRIB Command.....	119
Requirements 150: Format of Param 1 of the ATTRIB Command.....	121
Requirements 151: Listen Mode Handling of RFU Values of Minimum TR0 and TR1 .....	122
Requirements 152: Frame Size for Poll Frame (FSD) .....	122
Requirements 153: FSDI.....	123
Requirements 154: Listen Mode Handling of RFU Values of FSDI.....	123
Requirements 155: Setting the Bit Rate .....	124
Requirements 156: Format of Param 3 of the ATTRIB Command.....	124
Requirements 157: Listen Mode Handling of RFU Values of b8 to b4 of Param 3.....	124
Requirements 158: Format of Param 4 of the ATTRIB Command.....	125
Requirements 159: Listen Mode Handling of RFU Value of DID.....	125
Requirements 160: Higher Layer – INF.....	125
Requirements 161: DID in ATTRIB Response.....	126
Requirements 162: Higher Layer – Response .....	126
Requirements 163: Frame Timing – Type 4B Tag Platform.....	127
Requirements 164: Block Length.....	128
Requirements 165: DID.....	131
Requirements 166: Power Level Indicator .....	131
Requirements 167: Handling of RFU Values of DID Field .....	131
Requirements 168: CRC.....	132
Requirements 169: Power Level Indication .....	132
Requirements 170: Handling of RFU Values of INF Field of an S(WTX) Response.....	133
Requirements 171: Frame Waiting Time Extension .....	134
Requirements 172: Chaining Rules .....	135

Requirements 173: Block Sizes during Chaining.....	135
Requirements 174: Last Block .....	135
Requirements 175: Block Numbering Rules .....	136
Requirements 176: General Block Handling Rules.....	136
Requirements 177: Block Handling Rules for the Reader/Writer .....	137
Requirements 178: Block Handling Rules for the Card Emulator .....	138
Requirements 179: Exception Processing – Card Emulator.....	138
Requirements 180: Exception Processing – Reader/Writer .....	139
Requirements 181: Deactivation Rules .....	141
Requirements 182: Error Handling for Deactivation .....	141
Requirements 183: Timing – Type 4A and Type 4B Tag Half-duplex Protocol .....	142
Requirements 184: Sequence Format – NFC-DEP Protocol.....	143
Requirements 185: Bit Level Coding – NFC-DEP Protocol.....	143
Requirements 186: Frame Format – NFC-DEP Protocol.....	144
Requirements 187: Data and Payload Format – NFC-DEP Protocol.....	145
Requirements 188: Attribute Request Length .....	146
Requirements 189: NFCID <sub>3I</sub> Format .....	146
Requirements 190: Initiator Device Identification Number (DID <sub>I</sub> ) .....	147
Requirements 191: BS <sub>I</sub> and BR <sub>I</sub> Coding .....	147
Requirements 192: PP Format.....	148
Requirements 193: NAD .....	148
Requirements 194: NFCID <sub>3T</sub> Format.....	149
Requirements 195: Target Device Identification Number (DID <sub>T</sub> ) .....	149
Requirements 196: BS <sub>T</sub> and BR <sub>T</sub> Format .....	150
Requirements 197: TO Format.....	151
Requirements 198: Initiator Handling of RFU Value of WT .....	151
Requirements 199: PP Format.....	152
Requirements 200: NAD .....	152
Requirements 201: DID – PSL_REQ Command .....	152
Requirements 202: FSL.....	154
Requirements 203: DID – PSL_RES Response .....	154
Requirements 204: DID – DEP_REQ Command.....	155
Requirements 205: DID – DEP_RES Response.....	156
Requirements 206: Response Timeout Extension.....	159
Requirements 207: DID – DSL_REQ Command.....	160

Requirements 208: DID – DSL_RES Response.....	161
Requirements 209: RLS_REQ Command.....	162
Requirements 210: RLS_RES Response.....	162
Requirements 211: Frame Timing – NFC-DEP Protocol.....	163
Requirements 212: Response Waiting Time .....	164
Requirements 213: Chaining Rules .....	165
Requirements 214: Block Sizes during Chaining.....	165
Requirements 215: PDU Numbering Rules .....	166
Requirements 216: General PDU Handling Rules .....	167
Requirements 217: PDU Handling Rules for the Initiator .....	167
Requirements 218: PDU Handling Rules for the Target.....	167
Requirements 219: Exception Processing – Target.....	168
Requirements 220: Exception Processing – Initiator .....	169
Requirements 221: Values.....	170

# 1 Introduction

## 1.1 Scope

The scope of this document covers the digital interface and the half-duplex transmission protocol of the NFC Forum Device in its four roles (Initiator, Target, Reader/Writer, and Card Emulator).

This includes the modulation schemes, bit level coding, bit rates, frame formats, protocols, and command sets.

Within this scope, the Active Communication mode is not described in the current version of this document.

## 1.2 Audience

This document is intended for use by manufacturers wanting to implement an NFC Forum Device.

## 1.3 Applicable Documents or References

The following documents contain provisions that are referenced in this specification. The latest version including all published amendments applies unless a publication date is explicitly stated.

[ACTIVITY]	NFC Activity Specification, Version 1.0, September 2010, NFC Forum
[ANALOG]	NFC Analog, Latest version, NFC Forum
[EMV_CLESS]	EMV Contactless Communication Protocol Specification, Version 2.0 July 2008 EMVCo
[ISO/IEC_13239]	ISO/IEC 13239, Information technology – Telecommunications and information exchange between systems – High-level data link control (HDLC) procedures, 2002, ISO/IEC

- [ISO/IEC\_14443] Identification cards – Contactless integrated circuit cards – Proximity cards  
Includes:
- [ISO/IEC 14443-1:2008], Identification cards – Contactless integrated circuit cards – Proximity cards – Part 1: Physical characteristics
  - [ISO/IEC 14443-2:2010], Identification cards – Contactless integrated circuit cards – Proximity cards – Part 2: Radio frequency power and signal balance
  - [ISO/IEC 14443-3:2001], Identification cards – Contactless integrated circuit cards – Proximity cards – Part 3: Initialization and anticollision
  - [ISO/IEC\_14443-3:2001/Amd.1], Identification cards -- Contactless integrated circuit(s) cards -- Proximity cards -- Part 3: Initialization and Anti-collision, 1 February 2001 with Amendment 1: Bit rates of fc/64, fc/32 and fc/16, 15 June 2005; Amendment 3: Handling of reserved fields and values, 22 March 2006; and Corrigendum 1: Amendment 1 - Corrigendum, 29 August 2006
  - [ISO/IEC 14443-4:2008], Identification cards – Contactless integrated circuit cards – Proximity cards – Part 4: Transmission protocol
- ISO/IEC
- [ISO/IEC\_18092] ISO/IEC 18092:  
Information technology – Telecommunications and information exchange between systems – Near Field Communication – Interface and Protocol (NFCIP-1)  
2004  
ISO/IEC
- [JIS\_X\_6319-4] JIS X 6319-4  
Specification of implementation for integrated circuit(s) cards – Part 4: High speed proximity cards  
2005  
JIS
- [RFC2119] Key words for use in RFCs to Indicate Requirement Levels, RFC 2119,  
S. Bradner,  
March 1997  
Internet Engineering Task Force
- [T1TOP] Type 1 Tag Operation Specification  
Version 1.0,  
July 2007  
NFC Forum
- [T2TOP] Type 2 Tag Operation,  
Version 1.0,  
July 2007  
NFC Forum

- [T3TOP]           Type 3 Tag Operation,  
Version 1.0,  
August 2007  
NFC Forum
- [T4TOP]           Type 4 Tag Operation,  
Version 2.0,  
September 2010  
NFC Forum

## 1.4 Administration

The NFC Digital Protocol is an open specification supported by the Near Field Communication Forum, Inc., located at:

401 Edgewater Place, Suite 600  
Wakefield, MA, 01880

Tel.: +1 781-876-6216

Fax: +1 781-610-9864

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

The NFC Devices Technical Working Group maintains this specification.

## 1.5 Name and Logo Usage

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

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

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

## 1.6 Intellectual Property

The NFC Digital Protocol conforms to the Intellectual Property guidelines specified in the NFC Forum's *Intellectual Property Rights Policy*, as outlined in the NFC Forum *Rules of Procedure*.

## 1.7 Acknowledgements

Extracts are derived from [ISO/IEC\_14443] and [ISO/IEC\_18092].

British Standards can be obtained in PDF or hardcopy formats from the BSI online shop at [www.bsigroup.com/Shop](http://www.bsigroup.com/Shop) or by contacting BSI Customer Services for hardcopies only at +44 (0)20 8996 9001, email: [cservices@bsigroup.com](mailto:cservices@bsigroup.com).

## 1.8 Special Word Usage

The following words have a special meaning in this specification:

- MUST and MAY

The key words MUST and MAY are to be interpreted as described in [RFC2119].

- HAS TO and HAVE TO

The key words HAS TO and HAVE TO indicate functionality that is mandatory yet outside of the scope of this specification. Therefore, statements using HAS TO and HAVE TO are not normative.

- MUST RAISE THE <error> PROTOCOL EXCEPTION

Certain events are identified in the digital protocol that affects upper layer protocols that are not specified in this specification. For the purposes of describing these dependencies between independent specifications, those events and conditions are described through the raising of a protocol exception indicated by the key words: MUST RAISE THE <error> PROTOCOL EXCEPTION.

This does not imply particular implementation requirements or any specific implementation architecture. It is not anticipated that these events will be externally observable or distinguishable and therefore they are not suitable for conformance testing.

## 1.9 Requirement Numbering

Requirements in this document are uniquely numbered with the number appearing next to each requirement. Requirements can include informative statements in the italic font. For example:

**Table 1: Sample Requirement**

---

1.9.1.1	A car <b>MUST</b> have four wheels.
	<i>A car <b>MAY</b> have alloy wheels.</i>

---

A requirement can have different numbers in different versions of the specifications. Hence, all references to a requirement must include the version of the document as well as the requirement's number.

## 1.10 Implementation of Optional Items

This specification contains items declared as optional for an implementation. However, if optional items are implemented, they have to be implemented as specified in this specification.

### Requirements 1: Implementation of Optional Items

---

#### Poll and Listen Mode

---

1.10.1.1 The NFC Forum Device implementing an item declared as optional **MUST** implement it as specified in this specification.

---

NOTE For example, NFC-B in Listen Mode is declared optional. This indicates that an implementation can choose to not implement it, but if NFC-B technology in Listen Mode is implemented, it has to be implemented as specified in this specification.

## 1.11 Notational Conventions

### 1.11.1 Notations

The notations as shown in Table 2 apply in this document:

**Table 2: Notational Conventions**

Notation	Description
XYh	Hexadecimal notation. Values expressed in hexadecimal form are followed by a lower case “h”. For example, 27509 decimal is expressed in hexadecimal as 6B75h.
XYb	Binary notation. Values expressed in binary form are followed by a lower case “b”. For example, 82h hexadecimal is expressed in binary as 10000010b.
[...]	Optional part.
xx	More than one value possible.
<b>STATE</b>	States are written in <b>COURIER FONT</b> to distinguish them from the text.

### 1.11.2 Value of Parameters

Throughout the document, symbols are used to identify the values of parameters. The actual values of the parameters are listed in Appendix A. Symbols referenced in Appendix A are written in **Arial bold** to distinguish them in the text.



## Requirements 2: Reserved for Future Use

### Poll and Listen Mode

- 1.11.2.1 The NFC Forum Device sending bytes or bits defined as RFU MUST set these bytes or bits to the value indicated, or to zero if no value is given.
- 1.11.2.2 The NFC Forum Device receiving bytes or bits defined as RFU MUST disregard these bytes or bits and MUST keep the same interpretation of any other field of the whole response, unless explicitly stated otherwise.

*The NFC Forum Device sending bytes or bits defined as Any Value MAY set these bytes or bits to any specific value.*

*How to process bytes or bits received that are defined as Any Value is out of scope of this specification, where not explicitly stated otherwise.*

## 1.12 Abbreviations

Abbreviation	Description
ASK	Amplitude Shift Keying
ATN	Attention
BCC	UID CL <sub>n</sub> check byte for NFC-A
bd	Bit Duration
CL <sub>n</sub>	Cascade Level n ( $1 \leq n \leq 3$ )
CRC	Cyclic Redundancy Check, a checksum appended within the data segment before transmission, and verified afterwards by the recipient to detect Transmission Errors
CRC_A	CRC error detection code for NFC-A
CRC_B	CRC error detection code for NFC-B
CRC_F	CRC error detection code for NFC-F
DID	Device Identification Number
<b>D</b> <sub>LISTEN→POLL</sub>	Divisor for communication direction Listen→Poll
<b>D</b> <sub>POLL→LISTEN</sub>	Divisor for communication direction Poll→Listen
EoD	End of Data
EoF	End of Frame
EoS	End of Sequence
<b>f</b> <sub>c</sub>	Carrier Frequency
FWI	Frame Waiting time Integer
FWT	Frame Waiting Time
IEC	International Electrotechnical Commission
ISO	International Organization for Standardization

Abbreviation	Description
JIS	Japanese Industrial Standard
LSB	Least Significant Byte
lsb	Least Significant Bit
Max	Maximum
MBL	Maximum Buffer Length
MBLI	Maximum Buffer Length Index
Min	Minimum
MRT	Maximum Response Time
MRTI	Maximum Response Time Information
MSB	Most Significant Byte
msb	Most Significant Bit
n.a.	Not Applicable
NAD	Node Addressing
NDEF	NFC Data Exchange Format
NFC	Near Field Communication
NFC-A	Near Field Communication – Type A Technology
NFC-B	Near Field Communication – Type B Technology
NFC-F	Near Field Communication – Type F Technology
NFCID0	NFC-B identifier of the NFC Forum Device. NFCID0 is always 4 bytes long.
NFCID1	NFC-A identifier of the NFC Forum Device in the Passive Communication mode. NFCID1 can be 4, 7, or 10 bytes long (simple, double, or triple size).
NFCID1 CL $n$	Contains the portion of the NFCID1 relative to the Cascade Level $n$ . NFCID1 CL $n$ is always 4 bytes long.
NFCID2	NFC-F identifier of the NFC Forum Device in the Passive Communication mode. NFCID2 is always 8 bytes long.
NFCID3	NFCIP-1 identifier of the NFC Forum Device. NFCID3 is always 10 bytes long.
NFCIP-1	Near Field Communication Interface and Protocol as specified in [ISO/IEC_18092].
NRZ-L	Non-Return to Zero (L for Level)
OOK	On-Off Keying

Abbreviation	Description
PDU	Protocol Data Unit
RF	Radio Frequency (RF field = magnetic field)
RFU	Reserved for Future Use
RRDD	Reader-Reader Data Delay
RTOX	Response Timeout Extension
RWT	Response Waiting Time
SDD	Single Device Detection
SFGI	Start-up Frame Guard time Integer
SFGT	Start-up Frame Guard Time
SoD	Start of Data
SoF	Start of Frame
SoS	Start of Sequence
WT	Waiting Time, parameter to code RWT

## 1.13 Glossary

### 1.13.1 Device and Communication

#### *Active Communication*

A communication mode in which each device generates its own RF field to send a message to another device.

#### *Command*

An instruction from one device to another device in order to move the other device through a state machine.

#### *Connectionless Transport*

An unacknowledged data transmission service with minimal protocol complexity.

#### *Listen Frame*

A frame sent by an NFC Forum Device in Listen Mode.

#### *NFC Forum Device*

A device that supports the following Modus Operandi: Initiator, Target, and Reader/Writer. It may also support Card Emulator.

#### *NFC Tag*

A contactless tag or (smart) card supporting NDEF over Passive Communication.

#### *Operating Field*

The magnetic field created by an NFC Forum Device in Poll Mode within the Operating Volume.

### *Operating Volume*

The three-dimensional space, as defined by the NFC Forum, in which an NFC Forum Device in Poll Mode can communicate with an NFC Forum Device in Listen Mode.

### *Passive Communication*

A communication mode in which one device generates an RF field and sends Commands to a second device. To respond, this second device uses load modulation (i.e., it does not generate an RF field but it draws more or less power from the RF field).

### *Poll Command*

A Command to query an NFC Forum Device in Listen Mode or an NFC Forum Tag:

- ALL\_REQ or SENS\_REQ Command for NFC-A
- ALLB\_REQ or SENSB\_REQ Command for NFC-B
- SENSF\_REQ Command for NFC-F

### *Poll Frame*

A frame sent by an NFC Forum Device in Poll Mode.

### *Response*

Information sent from one device to another device upon receipt of a Command. The information received by the other device should allow this other device to continue the data exchange.

### *Technology*

A group of transmission parameters defined by the NFC standard that make a complete communication protocol. A non-exhaustive list of transmission parameters is: RF carrier, communication mode, bit rate, modulation scheme, bit level coding, frame format, protocol, and Command set. NFC defines three groups and therefore three Technologies: NFC-A, NFC-B, and NFC-F. The three Technologies use the same RF carrier (13.56 MHz). Each Technology uses its own modulation scheme, bit level coding and frame format, but may have the same protocol and Command set.

### *Technology Subset*

A legacy platform supporting a subset of a Technology. A Technology Subset supports at least the Poll Command of the Technology. The four Technology Subsets described in this specification are:

- Type 1 Tag platform, which uses a particular subset of NFC-A, excluding anti-collision.
- Type 2 Tag platform, which uses a particular subset of NFC-A, including anti-collision.
- Type 3 Tag platform, which uses a particular subset of NFC-F, including anti-collision.
- Type 4 Tag platform, which uses a particular subset of NFC-A or NFC-B, including anti-collision.

### 1.13.2 Protocol and Mode

#### *Activity*

A process within an NFC Forum Device with well defined pre-conditions and post-conditions, as defined in [ACTIVITY]. An Activity can only start when its pre-conditions are fulfilled. When an Activity ends, its post-conditions are fulfilled.

#### *Card Emulator*

A role of an NFC Forum Device, reached when an NFC Forum Device in Listen Mode has gone through a number of Activities and in which the NFC Forum Device behaves as one of the Technology Subsets.

#### *Initiator*

A role of an NFC Forum Device reached when an NFC Forum Device in Poll Mode has gone through a number of Activities; in this mode the NFC Forum Device communicates using the NFC-DEP Protocol.

#### *ISO-DEP Protocol*

Half-duplex block transmission protocol defined in Section 13 and based on [ISO/IEC\_14443] and [EMV\_CLESS].

#### *Listen Mode*

Initial mode of an NFC Forum Device when it does not generate a carrier; in this mode the NFC Forum Device listens for the RF field of another device.

#### *NFC-DEP Protocol*

Half-duplex block transmission protocol defined in Section 14 and based on [ISO/IEC\_18092].

#### *Poll Mode*

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

#### *Reader/Writer*

Role of an NFC Forum Device reached when an NFC Forum Device in Poll Mode has gone through a number of Activities. In this mode, the NFC Forum Device behaves like a legacy contactless reader and uses Commands from one of the Technology Subsets.

#### *Target*

Role of an NFC Forum Device, reached when the NFC Forum Device has gone through a number of Activities in which the NFC Forum Device communicates using the NFC-DEP Protocol.

### 1.13.3 Errors

#### *Protocol Error*

A Semantic Error or Syntax Error.

*Semantic Error*

A Correct Frame with no Syntax Error is received when it is not expected.<sup>1</sup>

*Syntax Error*

A Correct Frame is received with an invalid content. In this case, the coding of the Command or the block within the frame is not consistent with this specification.

*Timeout Error*

No Response has been received within the Response Waiting Time (RWT).

*Transmission Error*

An incorrect frame is received. In this case, the signal modulation, the bit coding, the frame format, the timing, or the checksum is not consistent with this specification.

*Correct Frame*

A frame without Transmission Error.

*Valid Block, Valid PDU*

A block or PDU without Protocol Error within a Correct Frame.

*Valid Command, Valid Response*

A Command or Response without Protocol Error within a Correct Frame.

---

<sup>1</sup> A semantic error does not occur in this document. It is only included for consistency.

## 2 Overview

This specification covers the digital protocol of NFC Forum Device communication in Passive Mode.

An NFC Forum device exchanges information, referred to as “payload” in this document, by transmitting data packages. A data package optionally begins with a Start of Data (SoD) consisting of a start byte and/or a length byte, followed by the payload, and optionally ends with an End of Data (EoD). If present, the EoD consists of a two-byte checksum.

Data packages, which are digital information composed of the values Logic “0” and Logic “1”, are embedded in frames. These frames are optionally delimited by a Start of Frame (SoF) and an End of Frame (EoF). SoF, EoF, “0”, and “1” are coded digitally. This means that these elements are coded by using combinations of only two different values (e.g., a high/low signal level or a 0°/180° signal phase).

Frames are embedded in sequences to synchronize and calibrate the sending and receiving devices and to trigger and stop the signal-decoding process in the receiving device. A sequence contains a frame with a particular leading signal pattern called Start of Sequence (SoS) and a particular closing signal pattern called End of Sequence (EoS).

Table 3 summarizes the structure of this document. The left column contains the different Activities. The upper row of the right column shows that the behavior of the NFC Forum Device during the Mode Switch Activity, the Collision Detection Activity, and the Collision Resolution Activity depends on its Technology only. The lower row of the right column illustrates that the behavior of the NFC Forum Device during the Device Activation Activity, the Data Exchange Activity, and the Device Deactivation Activity also depends on its Technology Subset.

**Table 3: Activities versus Technology / Device Platform**

Activity	Technology / Device Platform						
Listen, RF Collision Avoidance, Technology Detection, Collision Resolution	NFC-A Section 4				NFC-B Section 5	NFC-F Section 6	
Device Activation	Type 1 Tag Platform Section 8		Type 2 Tag Platform Section 9	Type 4A Tag Platform Section 11	Type 4B Tag Platform Section 12	Type 3 Tag Platform Section 10	
Data Exchange	NFC-DEP Protocol Section 14	Type 1, 2, and 3 Tag Half-duplex Protocol Section 7		ISO-DEP Protocol Section 13		Type 1, 2, and 3 Tag Half-duplex Protocols Section 7	NFC-DEP Protocol Section 14
Device Deactivation							



### 3 Bit Duration

The bit duration indicates the timing of a digital signal. For this document, the bit duration (bd) is the time it takes to transmit one unit of information.

For communication direction Poll→Listen, the bit duration is defined as follows:

$$1 \text{ bd} = 128 / (f_c \times D_{\text{POLL} \rightarrow \text{LISTEN}})$$

For communication direction Listen→Poll, the bit duration is defined as follows:

$$1 \text{ bd} = 128 / (f_c \times D_{\text{LISTEN} \rightarrow \text{POLL}})$$

where  $f_c$  is the constant carrier frequency generated by the NFC Forum Device in Poll Mode, and the integers  $D_{\text{POLL} \rightarrow \text{LISTEN}}$  and  $D_{\text{LISTEN} \rightarrow \text{POLL}}$  are divisors that have the value 1, 2, or 4.

For NFC-A and NFC-B communication initiation, the value of  $D_{\text{POLL} \rightarrow \text{LISTEN}}$  and  $D_{\text{LISTEN} \rightarrow \text{POLL}}$  is equal to 1.

For NFC-F communication initiation, the value of  $D_{\text{POLL} \rightarrow \text{LISTEN}}$  and  $D_{\text{LISTEN} \rightarrow \text{POLL}}$  is equal to 2 or 4, depending on the configuration.

The communication initiation covers the Mode Switch Activity, Collision Detection Activity, Collision Resolution Activity, and Device Activation Activity (see [ACTIVITY]).

## 4 NFC-A Technology

This section specifies the NFC-A Technology-related features of the NFC Forum Device. Information defined elsewhere is included for readability reasons.

If not explicitly stated otherwise, [ISO/IEC\_18092] is the normative reference for this section. Deviations from and restrictions of [ISO/IEC\_18092] are indicated within the text on the topics for which deviations and restrictions apply.

### 4.1 Sequence Format

This section describes the sequence format for NFC-A Technology.

#### 4.1.1 Poll→Listen Modulation

In Poll Mode, the analog signal is modulated using Modified Miller coding with ASK 100% modulation, as illustrated in Figure 1.

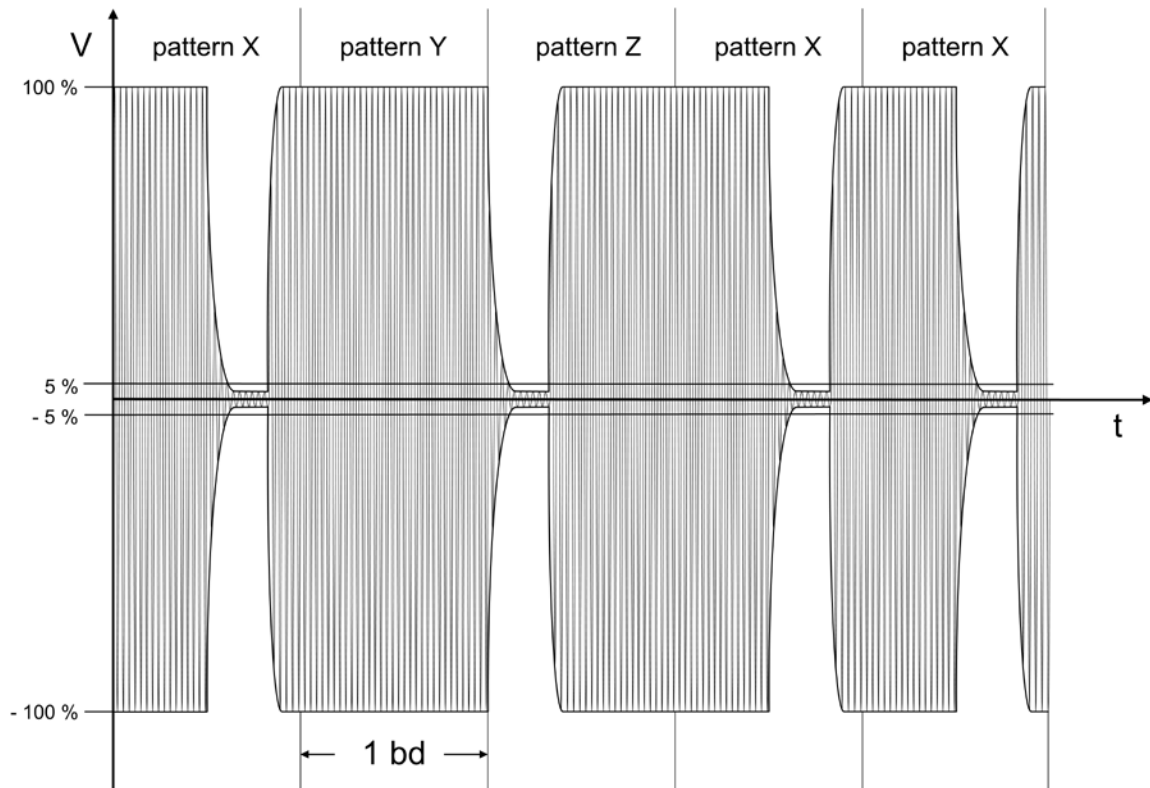


Figure 1: Modified Miller Coding with ASK 100%

The modulation principle of ASK 100% of the Operating Field is used to create  $V_2$ . Refer to [ANALOG] for the definition of  $V_2$ .

Modified Miller coding uses these  $V_2$  to define three particular patterns: X, Y, and Z.

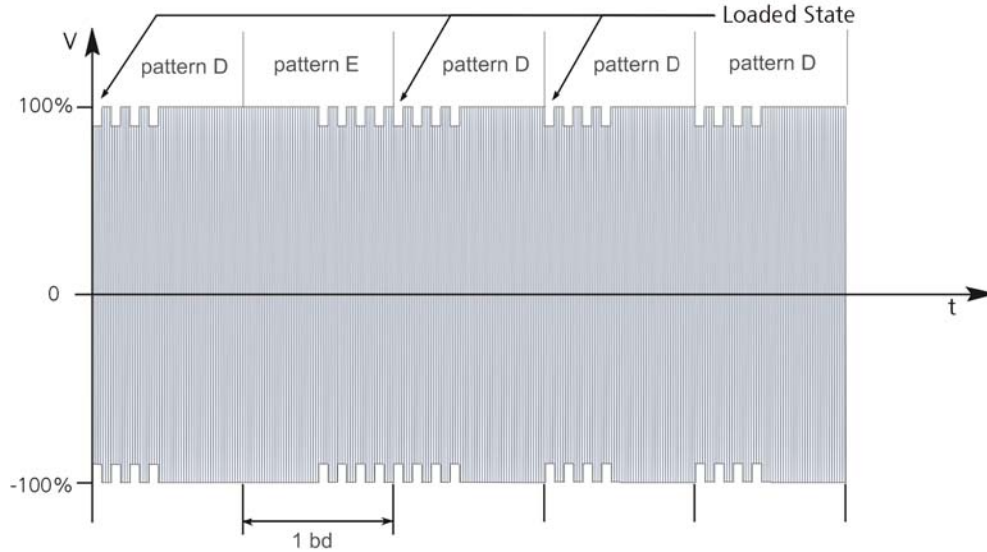
## Requirements 3: Signal Patterns Poll→Listen – NFC-A

Poll Mode		Listen Mode	
4.1.1.1	<p>The NFC Forum Device MUST build signal patterns X, Y, and Z, as follows:</p> <ul style="list-style-type: none"> <li>• Pattern X: at the beginning of the bit duration, no modulation MUST occur for a time of half the bit duration. After a time of half the bit duration, a <math>V_2</math> MUST occur.</li> <li>• Pattern Y: for the full bit duration, no modulation MUST occur.</li> <li>• Pattern Z: at the beginning of the bit duration, a <math>V_2</math> MUST occur. For the rest of the bit duration, no modulation MUST occur.</li> </ul>	4.1.1.2	<p>The NFC Forum Device MUST read signal patterns X, Y, and Z, as follows:</p> <ul style="list-style-type: none"> <li>• If the NFC Forum Device detects no modulation for a time of half the bit duration and a <math>V_2</math> after a time of half the bit duration, the NFC Forum Device MUST read this as pattern X.</li> <li>• If the NFC Forum Device detects a <math>V_2</math> at the beginning of the bit duration and no modulation for the rest of the bit duration, the NFC Forum Device MUST read this as pattern Z.</li> <li>• If the NFC Forum Device does not detect any modulation during the entire bit period, the NFC Forum Device MUST read this as pattern Y.</li> <li>• All other patterns MUST be treated as invalid patterns.</li> </ul>

NOTE The term “Pulse” as used in [ISO/IEC\_18092] is referred to as  $V_2$  in this document.

### 4.1.2 Listen→Poll Modulation

In Listen Mode, the analog signal is modulated using Manchester coding with OOK subcarrier modulation (see Figure 2).



**Figure 2: Manchester Coding with OOK**

Manchester coding uses OOK subcarrier modulation to define three particular patterns: E, D, and F.

#### Requirements 4: Modulation Listen→Poll – NFC-A

---

##### Listen Mode

---

4.1.2.1 The NFC Forum Device **MUST** modulate the carrier with the subcarrier such that the bit period starts with the loaded state of the subcarrier (see Figure 2).

---

### Requirements 5: Signal Patterns Listen→Poll – NFC-A

Poll Mode	Listen Mode
<p>4.1.2.2 The NFC Forum Device MUST read the patterns D, E, and F as follows:</p> <ul style="list-style-type: none"> <li>• If the carrier is modulated with the subcarrier for the first half of the bit duration and is not modulated for the second half of the bit duration, the NFC Forum Device MUST read that as pattern D.</li> <li>• If the carrier is not modulated with the subcarrier for the first half of the bit duration and is modulated for the second half of the bit duration, the NFC Forum Device MUST read that as pattern E.</li> <li>• If the carrier is not modulated with the subcarrier during one full bit period, the NFC Forum Device MUST read this as pattern F.</li> <li>• All other patterns MUST be treated as invalid patterns.</li> </ul>	<p>4.1.2.3 The NFC Forum Device MUST build the patterns D, E, and F as follows:</p> <ul style="list-style-type: none"> <li>• Pattern D: the carrier MUST be modulated with the subcarrier for the first half of the bit duration and MUST not be modulated for the remaining part of the bit duration.</li> <li>• Pattern E: the carrier MUST not be modulated with the subcarrier for the first half of the bit duration and MUST be modulated for the second half of the bit duration.</li> <li>• Pattern F: the carrier is not modulated with the subcarrier for one bit duration.</li> </ul>

#### 4.1.3 Synchronization

NFC-A Technology does not require signal synchronization and therefore, SoS is not present.

## 4.1.4 De-synchronization

The EoS indicates the end of a sequence.

### Requirements 6: De-synchronization Poll→Listen – NFC-A

Poll Mode		Listen Mode	
4.1.4.1	The NFC Forum Device MUST code EoS as follows: <ul style="list-style-type: none"> <li>EoS: pattern Y.</li> </ul>	4.1.4.2	The NFC Forum Device MUST decode EoS as follows: <ul style="list-style-type: none"> <li>If the NFC Forum Device detects pattern Y after pattern Y, then it MUST decode the last pattern Y as EoS.</li> <li>If the NFC Forum Device detects pattern Y after pattern Z, then it MUST decode pattern Y as EoS.</li> </ul>

### Requirements 7: De-synchronization Listen→Poll – NFC-A

Poll Mode		Listen Mode	
4.1.4.3	The NFC Forum Device MUST decode EoS as follows: <ul style="list-style-type: none"> <li>If the NFC Forum Device detects pattern F, then it MUST decode this as EoS.</li> </ul>	4.1.4.4	The NFC Forum Device MUST code EoS as follows: <ul style="list-style-type: none"> <li>EoS: pattern F</li> </ul>

## 4.2 Bit Level Coding

### 4.2.1 Poll→Listen Coding Scheme

The patterns X, Y, and Z are used to code the digital alphabet Logic “0” and Logic “1”. Logic “0”s and Logic “1”s are the components of frames.

## Requirements 8: Bit Level Coding Poll→Listen – NFC-A

Poll Mode	Listen Mode
<p>4.2.1.1 The NFC Forum Device MUST code Logic “0” and Logic “1” as follows:</p> <ul style="list-style-type: none"> <li>• Logic “1”: pattern X</li> <li>• Logic “0”: pattern Y</li> </ul> <p>with the following exceptions:</p> <ul style="list-style-type: none"> <li>• Pattern Z MUST be used to code the first Logic “0” (SoF).</li> <li>• If there are two or more contiguous Logic “0”s, pattern Z MUST be used from the second Logic “0” on.</li> </ul>	<p>4.2.1.2 The NFC Forum Device MUST decode Logic “0” and Logic “1” as follows:</p> <ul style="list-style-type: none"> <li>• The first pattern Z MUST be decoded as Logic “0”.</li> <li>• If the NFC Forum Device detects pattern X, then it MUST decode this as Logic “1”.</li> <li>• If the NFC Forum Device detects pattern Y after pattern X, then it MUST decode pattern Y as Logic “0”.</li> <li>• If the NFC Forum Device detects pattern Z after pattern Y, then it MUST decode pattern Z as Logic “0”.</li> <li>• If the NFC Forum Device detects pattern Z after pattern Z, then it MUST decode the last pattern Z as Logic “0”.</li> </ul> <p><i>Invalid patterns MAY be treated as Transmission Errors, or the device MAY use implementation-dependent methods to decode an invalid pattern into a Logic ‘0’ or Logic ‘1’.</i></p>

## 4.2.2 Listen→Poll Coding Scheme

The patterns E and D are used to code the digital alphabet Logic “0” and Logic “1”. Logic “0”s and Logic “1”s, referred to as data bits in the following, are the components of frames.

### Requirements 9: Bit Level Coding Listen→Poll – NFC-A

Poll Mode	Listen Mode
<p>4.2.2.1 The NFC Forum Device MUST decode Logic “0” and Logic “1” as follows:</p> <ul style="list-style-type: none"> <li>• If the NFC Forum Device detects pattern D, then it MUST decode this as Logic “1”.</li> <li>• If the NFC Forum Device detects pattern E, then it MUST decode this as Logic “0”.</li> </ul> <p><i>Invalid patterns MAY be treated as Transmission Errors, or the device MAY use implementation-dependent methods to decode an invalid pattern into a Logic “0” or Logic “1”.</i></p>	<p>4.2.2.2 The NFC Forum Device MUST code Logic “0” and Logic “1” as follows:</p> <ul style="list-style-type: none"> <li>• Logic “1”: pattern D</li> <li>• Logic “0”: pattern E</li> </ul>

## 4.3 Frame Format

Data bits, when transmitted between NFC Forum Devices, are grouped within frames. The format of a frame is different for each Technology. NFC-A Technology groups the data bits together in a frame by adding an SoF and an EoF. A parity bit (P) is added at the end of each 8 data bits.

This section defines the frames of NFC Forum Devices configured for NFC-A Technology. NFC-A Technology uses three types of frames: short frame, standard frame, and bit-oriented SDD frame. The short frame is used to initiate communication (wake-up). The standard frame is used for data exchange. The bit-oriented SDD frame is used for collision resolution.

NOTE Definitions in Section 4.3 overrule deviating definitions in [ISO/IEC\_18092].



**Requirements 10: Frame Format – NFC-A Technology**

**Poll and Listen Mode**

4.3.1.1 A frame MUST start with SoF.

For Poll→Listen communication, the SoF MUST be a Logic “0”.

For Listen→Poll communication, the SoF MUST be a Logic “1”.

4.3.1.2 A frame MUST end with EoF.

For Poll→Listen communication, EoF MUST be a Logic “0”.

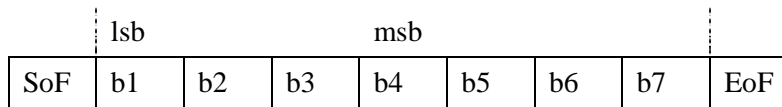
For Listen→Poll communication, EoF MUST be a Pattern F (see Requirement 4.1.2.3).

**4.3.2 Short Frame**

A short frame is used to initiate communication and consists of the following (see Figure 3):

- SoF
- Up to 7 data bits transmitted lsb first
- EoF

No parity is added.



**Figure 3: Short Frame**

**Requirements 11: Short Frame Format – NFC-A Technology**

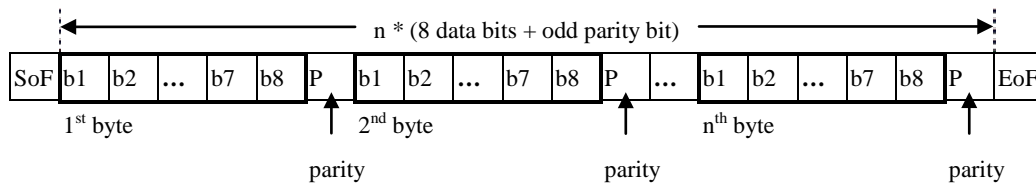
**Poll and Listen Mode**

4.3.2.1 Following the SoF and preceding the EoF, the short frame MUST contain less than 8 data bits.

**4.3.3 Standard Frame**

Standard frames are used for data exchange and consist of the following (see Figure 4):

- SoF
- $n * (8 \text{ data bits} + \text{odd parity bit})$ , with  $n \geq 1$
- EoF (Poll→Listen communication only)



**Figure 4: Standard Frame (Poll→Listen Communication)**

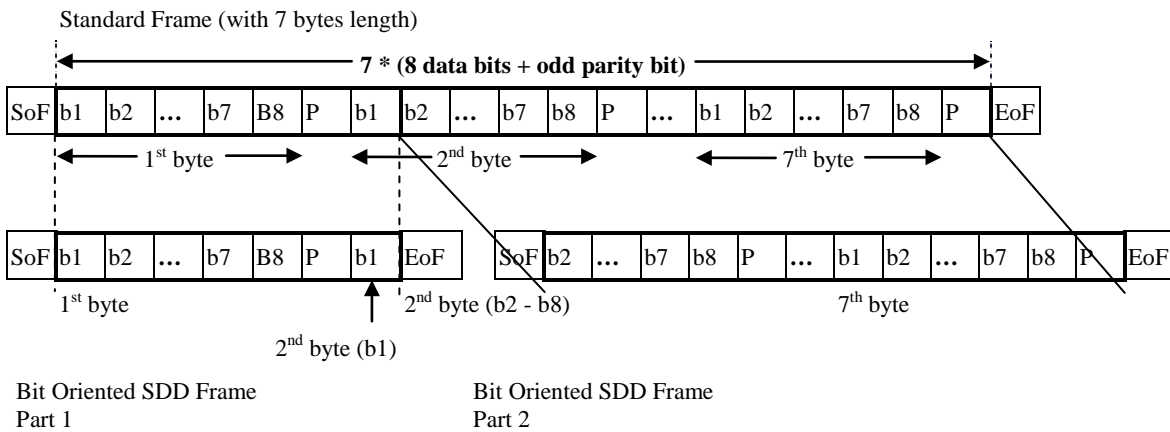
**Requirements 12: Standard Frame Format – NFC-A Technology**

**Poll and Listen Mode**

4.3.3.1 Each 8 data bits in a standard frame **MUST** be followed by an odd parity bit. The parity bit P **MUST** be set such that the number of Logic “1”s is odd in the set consisting of b1 to b8 and P.

**4.3.4 Bit Oriented SDD Frame**

Bit oriented SDD frames are used for collision resolution and result from a standard frame with a length of 7 bytes that is split into two parts. The split can occur after any data bit. Figure 5 shows an example with the split after the first bit of the second byte.



**Figure 5: Bit Oriented SDD Frame (with Split after the First Bit of the Second Byte)**

Bit oriented SDD frames are characterized as follows:

- Bit oriented SDD frames have a SoF and an EoF.
- Part 1 of a bit oriented SDD frame is always sent by an NFC Forum Device in Poll Mode.
- Part 2 of a bit oriented SDD frame is always sent by an NFC Forum Device in Listen Mode in response to part 1.
- The sum of data bits of part 1 and part 2 is 56 (7 bytes with 8 data bits each).
- The minimum length of part 1 is set to 16 data bits.
- The maximum length of part 1 is set to 55 data bits.
- As a result, the minimum length of part 2 are 8 data bits, the maximum length of part 2 is 40 data bits.
- If the split occurs after the eighth data bit of a byte, then the related parity bit is added after the last data bit of part 1.
- If the split occurs at another position within a byte, then no parity bit is added after the last data bit of part 1, and the first parity bit of part 2 is undefined.

**NOTE** If the split occurs after the eighth (i.e., the last) data bit of a byte, then part 1 and part 2 of the bit oriented SDD frame have the same format as standard frames.

**Requirements 13: Bit Oriented SDD Frame Format**

<b>Poll and Listen Mode</b>	
4.3.4.1	Each 8 data bits in a bit oriented SDD frame MUST be followed by an odd parity bit. The parity bit P MUST be set such that the number of Logic “1”s is odd in the set consisting of b1 to b8 and P.

**Requirements 14: Bit Oriented SDD Frame Format (Continued)**

<b>Poll Mode</b>		<b>Listen Mode</b>	
4.3.4.2	The minimum length of part 1 MUST be 16 data bits. The maximum length of part 1 MUST be 55 data bits.	4.3.4.3	The sum of data bits of part 1 and part 2 MUST be 56, as illustrated in Figure 5.
4.3.4.4	If the split occurs after the eighth data bit of a byte, then the related parity bit MUST be added after the last data bit of part 1.	4.3.4.5	If the split occurs at a position within a byte (i.e., the split does not occur after the eighth data bit), then the first parity bit of part 2 is undefined.  <i>The first parity bit of part 2 MAY be set to Any Value.</i>

**4.4 Data and Payload Format**

Data transmitted in an NFC-A standard frame (i.e., the bytes between SoF and EoF), have the following substructure. They consist of the payload and, depending on the particular payload, of an EoD (SoD is not used).

Section 4.5 describes the payload, which consists of the Commands and Responses.

If present, the EoD contains a two-byte checksum referred to as CRC\_A. The CRC\_A is defined as a function of k data bits. The number of bits k is a multiple of eight. Input for CRC\_A calculation is the payload.

Figure 6 illustrates the NFC-A data and payload format for a standard frame. Table 4 shows the cases in which the EoD is present.

Data					
Payload (Command or Response)				[EoD]	
Byte 1	Byte 2	...	Byte n	[CRC_A1]	[CRC_A2]

**Figure 6: Data and Payload Format – NFC-A Standard Frame**

Data embedded in NFC-A short frames or NFC-A bit oriented SDD frames do not have SoD and EoD.

### Requirements 15: Data and Payload Format – NFC-A

Poll and Listen Mode	
4.4.1.1	If required according to Table 4, the payload MUST be followed by an EoD at the position, as indicated in Figure 6. The EoD MUST contain a CRC_A.
4.4.1.2	The CRC_A MUST be calculated as defined in [ISO/IEC_13239], but the initial register content MUST be 6363h and the register content MUST not be inverted after calculation. CRC_A1 is the LSB and CRC_A2 is the MSB.
4.4.1.3	The NFC Forum Device MUST compare the received CRC_A with the CRC_A calculated according to Requirement 4.4.1.2. If different, then the NFC Forum Device MUST resort to exception processing with a Transmission Error.

## 4.5 Command Set

Payload exchanged between NFC Forum Devices consists of Commands and Responses. Table 4 lists the Commands that are available to the NFC Forum Device configured for NFC-A Technology. For each Command, the corresponding Response is indicated. Furthermore, Table 4 shows the frame type that a specific Command is embedded in and whether an EoD is present. Refer to Section 4.3 for the definition of NFC-A frames.

**Table 4: NFC-A – Command Set**

Command	Response	EoD Present	Frame Type
ALL_REQ, SENS_REQ		No	Short Frame
	SENS_RES	No	Standard Frame
SDD_REQ		No	Bit Oriented SDD Frame
	SDD_RES	No	Bit Oriented SDD Frame
SEL_REQ		Yes	Standard Frame
	SEL_RES	Yes	Standard Frame
SLP_REQ		Yes	Standard Frame

### Requirements 16: NFC-A – Command Set

Poll and Listen Mode	
4.5.1.1	Commands and Responses MUST be transmitted in compliance with Table 4.

## 4.6 ALL\_REQ and SENS\_REQ

The ALL\_REQ and SENS\_REQ Commands are sent by the NFC Forum Device in Poll Mode to probe the Operating Field for NFC Forum Devices in Listen Mode configured for NFC-A Technology.

### 4.6.1 ALL\_REQ Command

Table 5 shows the format of the ALL\_REQ Command.

**Table 5: Format of ALL\_REQ**

b7	b6	b5	b4	b3	b2	b1	Meaning
1	0	1	0	0	1	0	52h = ALL_REQ

### 4.6.2 SENS\_REQ Command

The format of the SENS\_REQ Command is specified in Table 6.

**Table 6: Format of SENS\_REQ**

b7	b6	b5	b4	b3	b2	b1	Meaning
0	1	0	0	1	1	0	26h = SENS_REQ

### 4.6.3 SENS\_RES Response

In response to an ALL\_REQ and SENS\_REQ Command from the NFC Forum Device in Poll Mode, an NFC Forum Device in Listen Mode returns a SENS\_RES Response with a length of 2 bytes, depending on its state. (See [ACTIVITY] for more details.) The SENS\_RES Response is coded as specified in Table 7 and Table 8.

**Table 7: Byte 1 of SENS\_RES**

b8	b7	b6	b5	b4	b3	b2	b1	Meaning	
0	0							NFCID1 size: single (4 bytes)	
0	1							NFCID1 size: double (7 bytes)	
1	0							NFCID1 size: triple (10 bytes)	
1	1							RFU	
		0						RFU	
			1	0	0	0	0	Bit frame SDD	
			0	1	0	0	0	Bit frame SDD	
			0	0	1	0	0	Bit frame SDD	
			0	0	0	1	0	Bit frame SDD	
			0	0	0	0	1	Bit frame SDD	
			0	0	0	0	0	Bit frame SDD (identifies a Type 1 Tag platform)	
			all other values						RFU

**Table 8: Byte 2 of SENS\_RES**

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	0	0	0					RFU
				1	1	0	0	Type 1 Tag platform configuration
				all other values				Other than Type 1 Tag platform configuration

**Requirements 17: NFCID1 Length**

Poll Mode		Listen Mode	
4.6.3.1	The NFC Forum Device MUST accept an NFCID1 of 4, 7, or 10 bytes.	4.6.3.2	The NFCID1 of the NFC Forum Device MUST have a length of 4, 7, or 10 bytes.

NOTE In [ISO/IEC\_18092], the NFCID1 is specified to be only 4 bytes long. Requirement 4.6.3.2 overrules this restriction.

**Requirements 18: Type 1 Tag Platform Configuration**

Poll Mode		Listen Mode	
4.6.3.3	<p>The NFC Forum Device MUST treat a SENS_RES Response as Protocol Error:</p> <ul style="list-style-type: none"> <li>• With b5-b1 of Byte 1 set to 00000b, but b4-b1 of Byte 2 set to a value different from 1100b or</li> <li>• With b4-b1 of Byte 2 set to 1100b, but b5-b1 of Byte 1 set to a value different from 00000b</li> </ul>	<p><i>The NFC Forum Device MAY be configured for Type 1 Tag platform and set b5-b1 of Byte 1 to 00000b and b4-b1 of Byte 2 to 1100b accordingly.</i></p>	

NOTE Requirement 4.6.3.3 extends [ISO/IEC\_18092].

**4.7 SDD\_REQ**

The SDD\_REQ Command is used to obtain the NFCID1 of an NFC Forum Device in Listen Mode and to detect whether more than one device of the same Technology is in the Operating Field of the NFC Forum Device in Poll Mode. Furthermore, the SDD\_REQ Command is used for collision resolution if there is more than one NFC Forum Device in Listen Mode in the Operating Field. Refer to [ACTIVITY] for details regarding the collision resolution mechanism.

### 4.7.1 SDD\_REQ Command

Table 9 specifies how the SDD\_REQ Command is coded.

**Table 9: Format of SDD\_REQ Command**

Byte 1	Byte 2	n data bits
SEL_CMD	SEL_PAR	data bit 1 ... data bit n

The SEL\_CMD byte indicates the cascade level (CL) of the NFCID1 requested by the NFC Forum Device in Poll Mode and is coded as shown in Table 10.

**Table 10: Format of SEL\_CMD**

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1	0	0	1	0	0	1	1	93h: SDD_REQ CL1
1	0	0	1	0	1	0	1	95h: SDD_REQ CL2
1	0	0	1	0	1	1	1	97h: SDD_REQ CL3
all other values								RFU

The SEL\_PAR byte indicates the length of the SDD\_REQ Command, including the SEL\_CMD and SEL\_PAR byte, and therefore, it also indicates the number of data bits following the SEL\_PAR byte.

The upper 4 bits are called “byte count” and specify the integer part of the number of data bits transmitted by the NFC Forum Device in Poll Mode (including SEL\_CMD and SEL\_PAR) divided by 8. Consequently, the minimum value of “Byte count” is 2. A complete NFCID1, including checksum, has a length of 5 bytes, and therefore, the maximum value of “Byte count” is 7. The byte count format is shown in Table 11.

The lower 4 bits are called “bit count” and specify the number of data bits transmitted by the NFC Forum Device in Poll Mode modulo 8. The bit count format is shown in Table 12.

For example, if SEL\_PAR equals 20h, then no data bits are following. If SEL\_PAR equals 35h, then 13 data bits are following.

**Table 11: Format of SEL\_PAR (Upper 4 Bits)**

b8	b7	b6	b5	Meaning
0	0	1	0	Byte count = 2
0	0	1	1	Byte count = 3
0	1	0	0	Byte count = 4
0	1	0	1	Byte count = 5
0	1	1	0	Byte count = 6
0	1	1	1	Byte count = 7

**Table 12: Format of SEL\_PAR (Lower 4 Bits)**

<b>b4</b>	<b>b3</b>	<b>b2</b>	<b>b1</b>	<b>Meaning</b>
0	0	0	0	Bit count = 0
0	0	0	1	Bit count = 1
0	0	1	0	Bit count = 2
0	0	1	1	Bit count = 3
0	1	0	0	Bit count = 4
0	1	0	1	Bit count = 5
0	1	1	0	Bit count = 6
0	1	1	1	Bit count = 7

The data bits following the SEL\_PAR contain the first n bits of an NFCID1 in the cascade level as specified by the SEL\_CMD byte. n is calculated by the value of SEL\_PAR. The SDD\_REQ Command contains a maximum of 39 data bits, excluding the SEL\_CMD and the SEL\_PAR.

An NFC Forum Device in Listen Mode that receives the SDD\_REQ Command compares its NFCID1 with the data bits received. If the first n bits of its NFCID1 in the respective cascade level are equal to the n data bits of the SDD\_REQ Command, then this NFC Forum Device responds to the SDD\_REQ Command. Otherwise, the SDD\_REQ Command is ignored by the NFC Forum Device in Listen Mode.



**Requirements 19: Format of SDD\_REQ Command**

Poll Mode	Listen Mode
<p>4.7.1.1 The NFC Forum Device MUST set SEL_CMD to a value that is compliant with Table 10.</p>	<p>4.7.1.2 The NFC Forum Device MUST be ready to receive a SEL_CMD value that is compliant with Table 10. The NFC Forum Device MUST treat a SEL_CMD value not compliant with Table 10 as Protocol Error.</p>
<p>4.7.1.3 If the NFC Forum Device does not support collision resolution, then the following applies:</p> <ul style="list-style-type: none"> <li>The NFC Forum Device MUST set SEL_PAR to a value that is compliant with the length of the SDD_REQ Command. The SEL_PAR value MUST be 20h.</li> </ul> <p>If the NFC Forum Device supports collision resolution, then the following applies:</p> <ul style="list-style-type: none"> <li>The NFC Forum Device MUST set SEL_PAR to a value that is compliant with the length of the SDD_REQ Command. The minimum SEL_PAR value MUST be 20h. The maximum SEL_PAR value MUST be 67h.</li> </ul>	<p>4.7.1.4 The NFC Forum Device MUST be ready to receive a SEL_PAR value that is compliant with the length of the SDD_REQ Command. The NFC Forum Device MUST be ready to receive a minimum SEL_PAR value of 20h. For collision resolution, the NFC Forum Device MUST be ready to receive a maximum SEL_PAR value of 67h. The NFC Forum Device MUST assign a Protocol Error to a SEL_PAR value that is not allowed.</p>

## 4.7.2 SDD\_RES Response

In response to a SDD\_REQ Command with a SEL\_PAR value equal to 20h, all NFC Forum Devices in Listen Mode in the Operating Field transmit the requested cascade level of their NFCID1 (NFCID1 CL<sub>n</sub>, with n=1, 2 or 3). The NFCID1 of an NFC Forum Device in Listen Mode consists of 4, 7, or 10 bytes. The length of the Response containing a complete NFCID1 cascade level (i.e., NFCID1 CL1, or NFCID1 CL2, or NFCID1 CL3) is always 5 bytes. The coding of the Response depends on the value of the SEL\_CMD byte and the size of the NFCID1. Table 13 specifies how to code the SDD\_RES Response containing a complete NFCID1 cascade level.

**Table 13: SDD\_RES Response (NFCID1 CL<sub>n</sub> + BCC)**

SEL_CMD	NFCID1 Size	SDD_RES Response
93h	4	CL1: <b>nfcid1<sub>0</sub> nfcid1<sub>1</sub> nfcid1<sub>2</sub> nfcid1<sub>3</sub></b> BCC
93h	> 4	CL1: CT <b>nfcid1<sub>0</sub> nfcid1<sub>1</sub> nfcid1<sub>2</sub></b> BCC
95h	7	CL2: <b>nfcid1<sub>3</sub> nfcid1<sub>4</sub> nfcid1<sub>5</sub> nfcid1<sub>6</sub></b> BCC
95h	> 7	CL2: CT <b>nfcid1<sub>3</sub> nfcid1<sub>4</sub> nfcid1<sub>5</sub></b> BCC
97h	10	CL3: <b>nfcid1<sub>6</sub> nfcid1<sub>7</sub> nfcid1<sub>8</sub> nfcid1<sub>9</sub></b> BCC

- **nfcid1<sub>n</sub>** is the n<sup>th</sup> byte of the complete NFCID1, with **nfcid1<sub>0</sub>** being the most significant byte.
- CT is the cascade tag.
- BCC is an exclusive-OR over the first 4 bytes of the SDD\_RES Response.

Table 14 specifies the content of nfcid1<sub>0</sub> in the case of a single size NFCID1.

**Table 14: nfcid1<sub>0</sub> for Single-size NFCID1**

nfcid1 <sub>0</sub>	Description
08h	<b>nfcid1<sub>1</sub> to nfcid1<sub>3</sub></b> are dynamically generated
x0h to x7h x9h to xEh	Fixed diversified number (the assignment is out of scope of this specification)
18h, 28h, 38h, 48h, 58h, 68h, 78h, 88h, 98h, A8h, B8h, C8h, D8h, E8h, F8h xFh	RFU

**Requirements 20: NFCID1**

<b>Listen Mode</b>	
4.7.2.1	The NFCID1 can be dynamically generated by the NFC Forum Device. If dynamically generated by the NFC Forum Device, the length of the NFCID1 MUST be limited to 4 bytes.
4.7.2.2	A dynamically generated NFCID1 MUST be generated only on state transition from <b>POWER-OFF</b> to <b>IDLE</b> state and if an RLS_REQ Command is received (see [ACTIVITY] for the NFC Forum Device state machine).
4.7.2.3	The <b>nfcid1<sub>0</sub></b> for a single-size NFCID1 MUST be coded as specified in Table 14.
4.7.2.4	The NFC Forum Device MUST set <b>nfcid1<sub>0</sub></b> of a single-size NFCID1 and <b>nfcid1<sub>3</sub></b> of a double-size NFCID1 to a value different from 88h.

**Requirements 21: Format of SDD\_RES Response (Complete and Incomplete NFCID1)**

<b>Poll Mode</b>	<b>Listen Mode</b>
<i>The NFC Forum Device MAY treat a SDD_RES Response with a CT value that is inconsistent with the length of the NFCID1 as Protocol Error.</i>	4.7.2.5 The NFC Forum Device MUST set CT to a value of 88h.

**Requirements 22: Format of SDD\_RES Response (Complete NFCID1 Only)**

<b>Poll Mode</b>	<b>Listen Mode</b>
4.7.2.6 The NFC Forum Device MUST verify the BCC included in the SDD_RES Response. The NFC Forum Device MUST treat an incorrect BCC as a Transmission Error.	4.7.2.7 The NFC Forum Device MUST calculate a BCC as exclusive-or over the first 4 bytes of the SDD_RES Response (NFCID1 CLn).

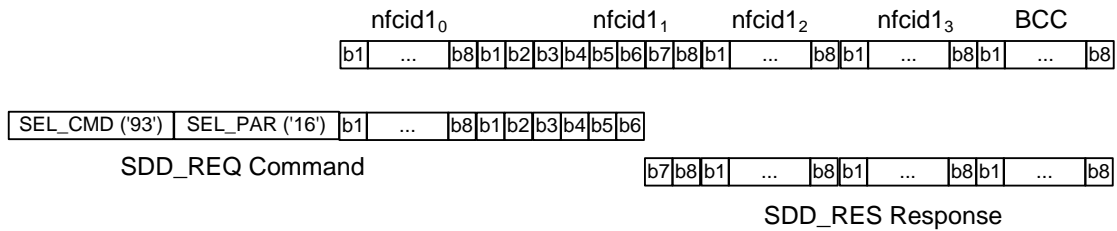
In response to an SDD\_REQ Command with a SEL\_PAR value different from 20h only those NFC Forum Devices in Listen Mode in the Operating Field respond to the Command that fulfill the following requirement:

- The first n bits of the NFCID1 in the respective cascade level are equal to the n data bits of the SDD\_REQ Command following the SEL\_PAR.

Otherwise, the SDD\_REQ Command is ignored by the NFC Forum Device in Listen Mode.

NFC Forum Devices in Listen Mode fulfilling this requirement return the remaining bits of the requested cascade level of their NFCID1 (NFCID1 CL<sub>n</sub>, with n=1, 2, or 3).

Figure 7 illustrates an example. The NFC Forum Device in Poll Mode transmits an SDD\_REQ Command with SEL\_CMD set to 93h, indicating the request for NFCID1 in cascade level 1. SEL\_PAR, set to 36h, indicates that 14 data bits are following (1 byte + 6 bits). NFC Forum Devices in Listen Mode, having an NFCID1 with the first 14 bits being equal to the 14 data bits sent in the SDD\_REQ Command, respond by sending the 26 remaining bits of their NFCID1.



**Figure 7: Example: SDD\_REQ with 14 Data Bits of NFCID1 CL1 (4 Bytes Size) and SDD\_RES**

**Requirements 23: Format of SDD\_RES Response (Incomplete NFCID1 Only)**

<b>Poll Mode</b>		<b>Listen Mode</b>	
4.7.2.8	<p>The NFC Forum Device MUST be ready to receive an SDD_RES Response containing 40-n data bits, where n is the number of data bits following the SEL_PAR in the SDD_REQ Command previously sent.</p> <p>The NFC Forum Device MUST treat an SDD_RES Response containing a different number of data bits as Protocol Error.</p>	4.7.2.9	<p>If the first n bits of the NFC Forum Device's NFCID1 in the respective cascade level are equal to the n data bits following the SEL_PAR in the preceding SDD_REQ Command, then the NFC Forum Device MUST return the remaining 40-n bits of its NFCID1 in the respective cascade level.</p> <p>Otherwise, the NFC Forum Device MUST ignore the SDD_REQ Command.</p>
4.7.2.10	<p>The NFC Forum Device MUST verify the BCC included in the SDD_RES Response. Verification data MUST be the first 4 bytes of the NFCID1 CLn (i.e., the concatenation of the data bits following the SEL_PAR in the preceding SDD_REQ Command and the SDD_RES Response excluding the BCC).</p> <p>The NFC Forum Device MUST treat an incorrect BCC as a Transmission Error.</p>	4.7.2.11	<p>The NFC Forum Device MUST calculate a BCC as exclusive-or over the first 4 bytes of the NFCID1 CLn (from which a part is returned in the SDD_RES Response).</p>

**4.8 SEL\_REQ**

Use the SEL\_REQ Command to select the NFC Forum Device in Listen Mode by means of its NFCID1.

**4.8.1 SEL\_REQ Command**

Table 15 specifies the code for the SEL\_REQ Command.

**Table 15: Format of SEL\_REQ Command**

<b>Byte 1</b>	<b>Byte 2</b>	<b>Bytes 3 - 6</b>	<b>Byte 7</b>
SEL_CMD	70h	NFCID1 CLn	BCC

The SEL\_CMD byte is coded as is shown in Table 10.

BCC is a checksum calculated as exclusive-or over the 4 bytes of the NFCID1 CLn.

The format of NFCID1 CLn is specified in Table 16.

Table 16: NFCID1 CL $n$ 

SEL_CMD	NFCID1 Size	NFCID1 CL $n$
93h	4	CL1: <b>nfcid1</b> <sub>0</sub> <b>nfcid1</b> <sub>1</sub> <b>nfcid1</b> <sub>2</sub> <b>nfcid1</b> <sub>3</sub>
93h	> 4	CL1: CT <b>nfcid1</b> <sub>0</sub> <b>nfcid1</b> <sub>1</sub> <b>nfcid1</b> <sub>2</sub>
95h	7	CL2: <b>nfcid1</b> <sub>3</sub> <b>nfcid1</b> <sub>4</sub> <b>nfcid1</b> <sub>5</sub> <b>nfcid1</b> <sub>6</sub>
95h	> 7	CL2: CT <b>nfcid1</b> <sub>3</sub> <b>nfcid1</b> <sub>4</sub> <b>nfcid1</b> <sub>5</sub>
97h	10	CL3: <b>nfcid1</b> <sub>6</sub> <b>nfcid1</b> <sub>7</sub> <b>nfcid1</b> <sub>8</sub> <b>nfcid1</b> <sub>9</sub>

**nfcid1** <sub>$n$</sub>  is the  $n^{\text{th}}$  byte of the complete NFCID1, with **nfcid1**<sub>0</sub> being the most significant byte.

#### 4.8.2 SEL\_RES Response

The NFC Forum Device in Listen Mode transmits the SEL\_RES Response in response to a SEL\_REQ Command. The length of the SEL\_RES Response is one byte. Table 17 specifies the format of the SEL\_RES Response.

Table 17: Format of SEL\_RES Response

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
x								Any value
	x	x						If b3 is set to 1b: <ul style="list-style-type: none"> <li>Any value</li> </ul> If b3 is set to 0b: <ul style="list-style-type: none"> <li>00b: Configured for Type 2 Tag platform</li> <li>01b: Configured for Type 4A Tag platform, compliant with [ISO/IEC_14443]</li> <li>10b: Configured for the NFC-DEP Protocol</li> <li>11b: Configured for the NFC-DEP Protocol and Type 4A Tag platform</li> </ul>
			x	x				Any value
					x			Cascade bit <ul style="list-style-type: none"> <li>1b: NFCID1 not complete</li> <li>0b: NFCID1 complete</li> </ul>
						x	x	Any value

NOTE The definition of SEL\_RES in Table 17 extends the definition of SEL\_RES in [ISO/IEC\_18092].

**Requirements 24: SEL\_RES Response**

<b>Poll Mode</b>		<b>Listen Mode</b>	
4.8.2.1	The NFC Forum Device MUST treat a SEL_RES Response with b3 set to 0b and b6-b7 set to 00b as coming from an NFC Forum Device configured for the Type 2 Tag platform, and MUST act as specified in [ACTIVITY].	4.8.2.2	The NFC Forum Device configured for the Type 2 Tag platform MUST set b6-b7 of the SEL_RES Response to 00b in the last cascade level (i.e., when the NFCID1 is complete).
4.8.2.3	The NFC Forum Device MUST treat a SEL_RES Response with b3 set to 0b and b6-b7 set to 01b as coming from an NFC Forum Device configured for the Type 4A Tag platform, and MUST act as specified in [ACTIVITY].	4.8.2.4	The NFC Forum Device configured for the Type 4A Tag platform MUST set b6-b7 of the SEL_RES Response to 01b in the last cascade level (i.e., when the NFCID1 is complete).
4.8.2.5	The NFC Forum Device MUST treat a SEL_RES Response with b3 set to 0b and b6-b7 set to 10b as coming from an NFC Forum Device configured for the NFC-DEP Protocol, and MUST act as specified in [ACTIVITY].	4.8.2.6	The NFC Forum Device configured for the NFC-DEP Protocol MUST set b6-b7 of the SEL_RES Response to 10b in the last cascade level (i.e., when the NFCID1 is complete).
4.8.2.7	The NFC Forum Device MUST treat a SEL_RES Response with b3 set to 0b and b6-b7 set to 11b as coming from an NFC Forum Device configured for both the Type 4A Tag platform and the NFC-DEP Protocol, and MUST act as specified in [ACTIVITY].	4.8.2.8	The NFC Forum Device configured for both the Type 4A Tag platform and the NFC-DEP Protocol MUST set b6-b7 of the SEL_RES Response to 11b in the last cascade level (i.e., when the NFCID1 is complete).

**4.9 SLP\_REQ**

Use the SLP\_REQ Command to put the NFC Forum Device in Listen Mode in the **SLEEP** state (see [ACTIVITY] for the NFC Forum Device state machine).

**4.9.1 SLP\_REQ Command**

The SLP\_REQ Command consists of 2 bytes. Table 18 specifies the format of the SLP\_REQ Command.

**Table 18: Format of SLP\_REQ Command**

<b>Byte 1</b>	<b>Byte 2</b>
50h	00h

## 4.9.2 SLP\_REQ Response

The NFC Forum Device in Listen Mode does not respond to an SLP\_REQ Command.

### Requirements 25: SLP\_REQ Response

Poll Mode		Listen Mode	
4.9.2.1	The NFC Forum Device MUST always treat the SLP_REQ Command as acknowledged by the NFC Forum Device in Listen Mode.	4.9.2.2	The NFC Forum Device MUST not respond to an SLP_REQ Command.

## 4.10 Timing Requirements

This section specifies the requirements for the different guard times and Frame Delay Times for NFC-A Technology.

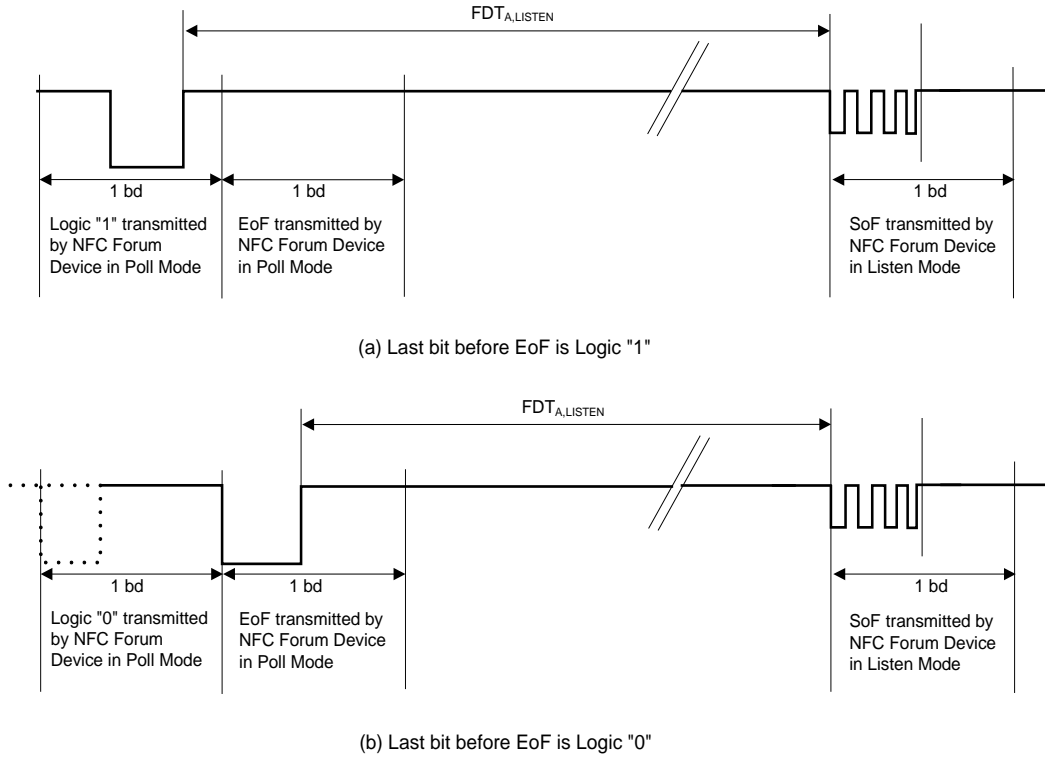
NOTE The term “Frame Response Time” as used in [ISO/IEC\_18092] is referred to as “Frame Delay Time” in this document.

### 4.10.1 Frame Delay Time Poll→Listen

The Frame Delay Time Poll→Listen ( $FDT_{A,LISTEN}$ ) is the time between a Poll Frame and a Listen Frame. The time between the minimum value  $FDT_{A,LISTEN,MIN}$  and the maximum value  $FDT_{A,LISTEN,MAX}$  defines the time interval in which a Listen Frame is allowed to be sent by an NFC Forum Device in Listen Mode in response to a Poll Frame of an NFC Forum Device in Poll Mode.

For NFC-A Technology,  $FDT_{A,LISTEN}$  depends on the logic value of the last bit before the EoF transmitted by the NFC Forum Device in Poll Mode, as illustrated in Figure 8.





**Figure 8:  $FDT_{A,LISTEN}$**

Table 19 shows the values  $FDT_{A,LISTEN}$  can take depending on the logic value of the last data bit transmitted by the NFC Forum Device in Poll Mode. The value of n is an integer and depends on the Command type transmitted within a Poll Frame. Section 4.5 defines the commands.

**Table 19:  $FDT_{A,LISTEN}$  and Logic State of Last Data Bit**

Logic State	$FDT_{A,LISTEN}$
"0"	$n \text{ bd} + 20/f_C$
"1"	$n \text{ bd} + 84/f_C$

$FDT_{A,LISTEN,MIN}$  follows the equations in Table 19 with a specific value of n, as shown in Table 20.

**Table 20:  $FDT_{A,LISTEN,MIN}$  and Command Type**

Command Type	n
ALL_REQ	9
SENS_REQ	
SDD_REQ	
SEL_REQ	
All other Commands	$\geq 9$

If the Poll Frame contains an ALL\_REQ, SENS\_REQ, SDD\_REQ, or SEL\_REQ Command, then  $FDT_{A,LISTEN,MAX}$  is equal to  $FDT_{A,LISTEN,MIN}$ . In this case, the Listen Frame is sent by the NFC Forum Device in Listen Mode at a specific point of time.

**Requirements 26:  $FDT_{A,LISTEN}$**

Poll Mode		Listen Mode	
4.10.1.1	Following the end of a Poll Frame, the NFC Forum Device MUST be ready to receive the start of a Listen Frame at a time aligned to the grid as defined in Figure 8, Table 19, and Table 20.	4.10.1.2	Following the receipt of a Poll Frame, the NFC Forum Device MUST align the first modulation edge within the start bit of a Listen Frame to the grid as defined in Figure 8, Table 19, and Table 20.
4.10.1.3	For the Commands ALL_REQ, SENS_REQ, SDD_REQ, and SEL_REQ, the NFC Forum Device MUST treat receipt of a Listen Frame at a time after $FDT_{A,LISTEN,MIN}$ as a Timeout Error (i.e., $FDT_{A,LISTEN,MAX}$ equals $FDT_{A,LISTEN,MIN}$ ).	4.10.1.4	For the Commands ALL_REQ, SENS_REQ, SDD_REQ, and SEL_REQ, the NFC Forum Device MUST always respond exactly at $FDT_{A,LISTEN,MIN}$ , as defined in Table 19 and Table 20 (i.e., $FDT_{A,LISTEN,MAX}$ equals $FDT_{A,LISTEN,MIN}$ ).

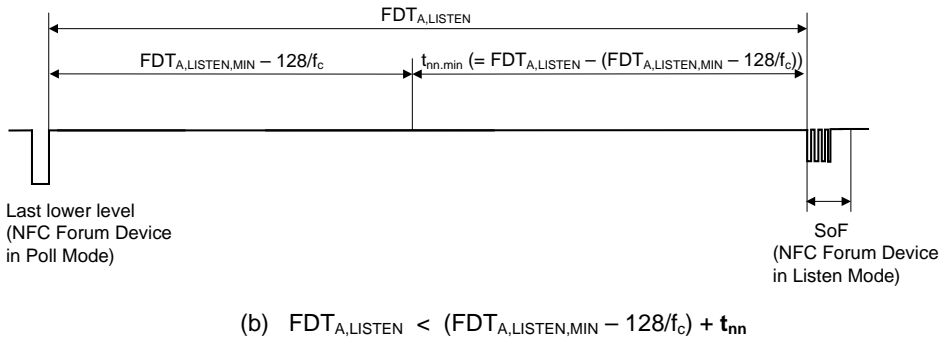
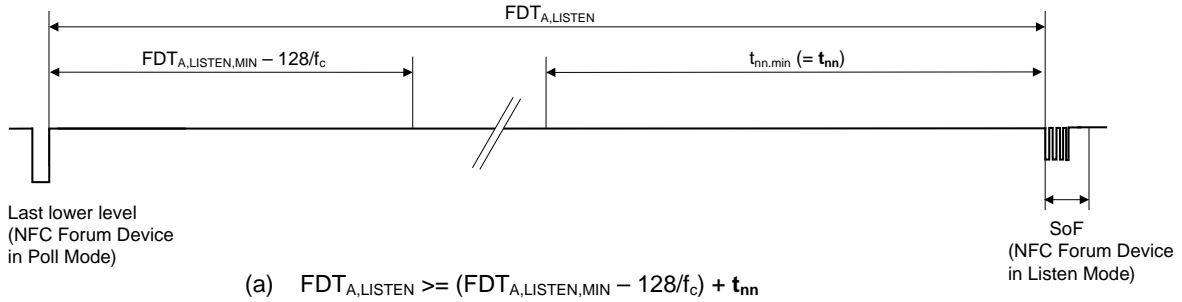
**Requirements 27: Tolerance**

Listen Mode	
4.10.1.5	The NFC Forum Device in Listen Mode MUST detect the end of the last $V_2$ transmitted by the NFC Forum Device in Poll Mode within the FDT tolerance $t_4$ . Refer to [ANALOG] for the definition of $t_4$ .

**Requirements 28:  $FDT_{A,LISTEN,MIN}$**

Poll Mode	
4.10.1.6	Following the end of a Poll Frame, the NFC Forum Device MUST ignore any response during a time $FDT_{A,LISTEN,MIN} - 128/f_C$ .

The NFC Forum Device in Listen Mode is not allowed to produce any detectable disturbance during a minimum time period  $t_{nn,min}$  before sending a Response.



**Figure 9:  $t_{nn,min}$  for NFC-A**

**Requirements 29:  $t_{nn,min}$  - NFC-A**

**Listen Mode**

4.10.1.7 The NFC Forum Device **MUST** not produce any detectable disturbance during a period of at least  $t_{nn,min}$ , defined as the minimum of  $FDT_{A,LISTEN} - (FDT_{A,LISTEN,MIN} - 128/f_c)$  and  $t_{nn}$ , before sending a Response.  $t_{nn,min}$  **MUST** be measured before the start of the SoF of the Listen Frame, as shown in Figure 9.

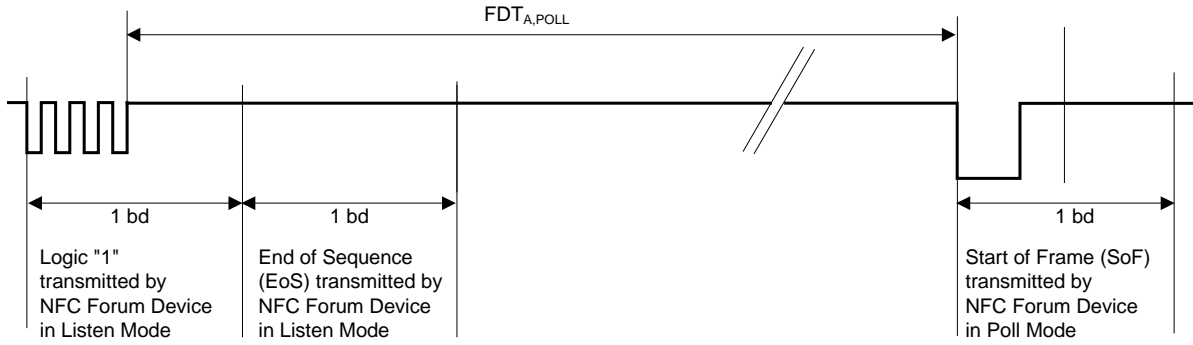
Refer to Appendix A.1 for the value of  $t_{nn}$ .

*The exact quantity of what constitutes ‘any detectable disturbance’ is not defined by this version of the specification.*

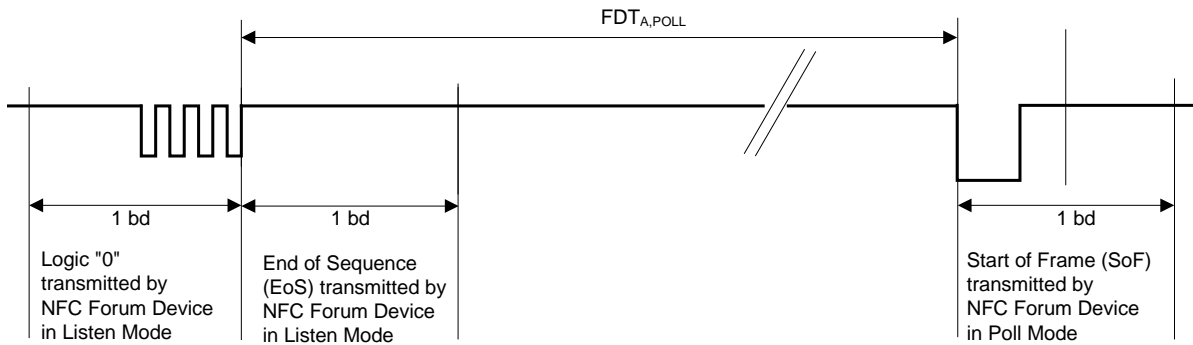
**4.10.2 Frame Delay Time Listen→Poll**

The Frame Delay Time Listen→Poll ( $FDT_{A,POLL}$ ) is the time between a Listen Frame and a Poll Frame. The minimum value  $FDT_{A,POLL,MIN}$  defines the time an NFC Forum Device in Poll Mode has to wait before sending a new Poll Frame after receipt of a Listen Frame. A maximum value  $FDT_{A,POLL,MAX}$  is not defined.

For NFC-A Technology, the definition of  $FDT_{A,POLL}$  depends on the logic value of the last data bit of the Listen Frame transmitted by the NFC Forum Device, as shown in Figure 10. However,  $FDT_{A,POLL}$  is not restricted to certain discrete values like  $FDT_{A,LISTEN}$ .



(a) Last transmitted data bit is Logic "1"



(b) Last transmitted data bit is Logic "0"

**Figure 10:  $FDT_{A,POLL}$**

**Requirements 30:  $FDT_{A,POLL,MIN}$**

Poll Mode	Listen Mode
<p>4.10.2.1 Following the end of a Listen Frame, the NFC Forum Device <b>MUST</b> wait at least for <b><math>FDT_{A,POLL,MIN}</math></b> before transmitting the start of a new Poll Frame. Refer to Appendix A.1 for the value of <b><math>FDT_{A,POLL,MIN}</math></b>.</p>	<p>4.10.2.2 The NFC Forum Device <b>MUST</b> be ready to receive the start of a new Poll Frame no later than <b><math>FDT_{A,POLL,MIN}</math></b> after the end of a Listen Frame.</p> <p><i>If the start of a new Poll Frame is received before <b><math>FDT_{A,POLL,MIN}</math></b>, then the NFC Forum Device <b>MAY</b> treat this as a Transmission Error.</i></p>

A specific value for  **$FDT_{A,POLL,MIN}$**  is defined for the re-activation of the NFC Forum Device in Listen Mode following a DSL\_RES Response (see Section 14.9.2), an RLS\_RES Response (see Section 14.10.2), or an S(DESELECT) Response (see Section 13.2.7).

**Requirements 31:  $FDT_{A,REACTIVATION}$**

<b>Poll Mode</b>	<b>Listen Mode</b>
<p>4.10.2.3 Following the end of a DSL_RES Response, an RLS_RES Response, or an S(DESELECT) Response, the NFC Forum Device MUST wait at least for <math>FDT_{A,REACTIVATION}</math> before transmitting the start of a new Poll Frame.</p> <p>Refer to Appendix A.1 for the value of <math>FDT_{A,REACTIVATION}</math>.</p>	<p>4.10.2.4 The NFC Forum Device MUST be ready to receive the start of a new Poll Frame no later than <math>FDT_{A,REACTIVATION}</math> after the end of a DSL_RES Response, an RLS_RES Response, or an S(DESELECT) Response.</p> <p><i>If the start of a new Poll Frame is received before <math>FDT_{A,REACTIVATION}</math>, then the NFC Forum Device MAY treat this as a Transmission Error.</i></p>

**4.10.3 Guard Time**

This section specifies the guard time of Unmodulated Carrier after which the NFC Forum Device in Listen Mode must be ready to receive an ALL\_REQ or SENS\_REQ Command.

**Requirements 32: Guard Time – NFC-A**

<b>Listen Mode</b>
<p>4.10.3.1 When an NFC Forum Device in Listen Mode is exposed to an Unmodulated Carrier (see [ANALOG]), it MUST be ready to receive an ALL_REQ or SENS_REQ Command within a guard time <math>GT_A</math>.</p> <p>Refer to Appendix A for the value of <math>GT_A</math>.</p>

## 5 NFC-B Technology

This section specifies the NFC-B Technology-related features of the NFC Forum Device. Information defined elsewhere has been included for readability reasons.

If not explicitly stated otherwise, [ISO/IEC\_14443] is normative reference for this section. Deviations from and restrictions of [ISO/IEC\_14443] are indicated within the text on the topics for which deviations and restrictions apply.

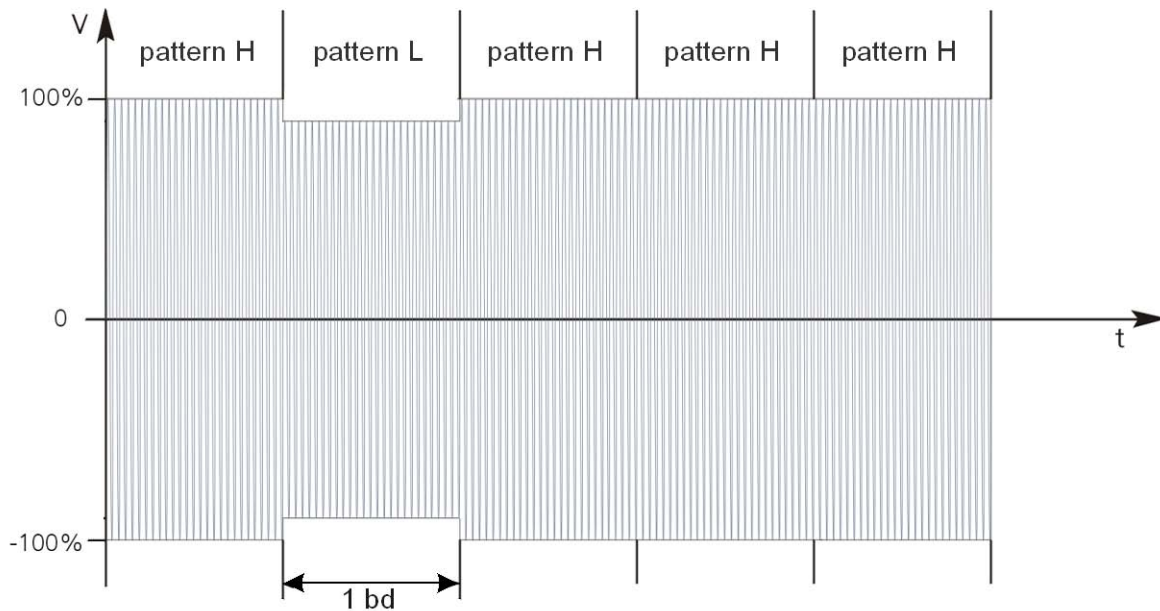
Implementation of NFC-B Listen Mode is optional. However, if implemented, then it has to be implemented as specified here (see Requirements 1.10.1.1).

### 5.1 Sequence Format

This section describes the sequence format for NFC-B Technology.

#### 5.1.1 Poll→Listen Modulation

In Poll Mode, the analog signal is modulated using NRZ-L coding based on the ASK 10% modulation principle. For details about the Modulation Index, refer to [ANALOG]. Figure 11 illustrates NRZ-L coding.



**Figure 11: NRZ-L Coding**

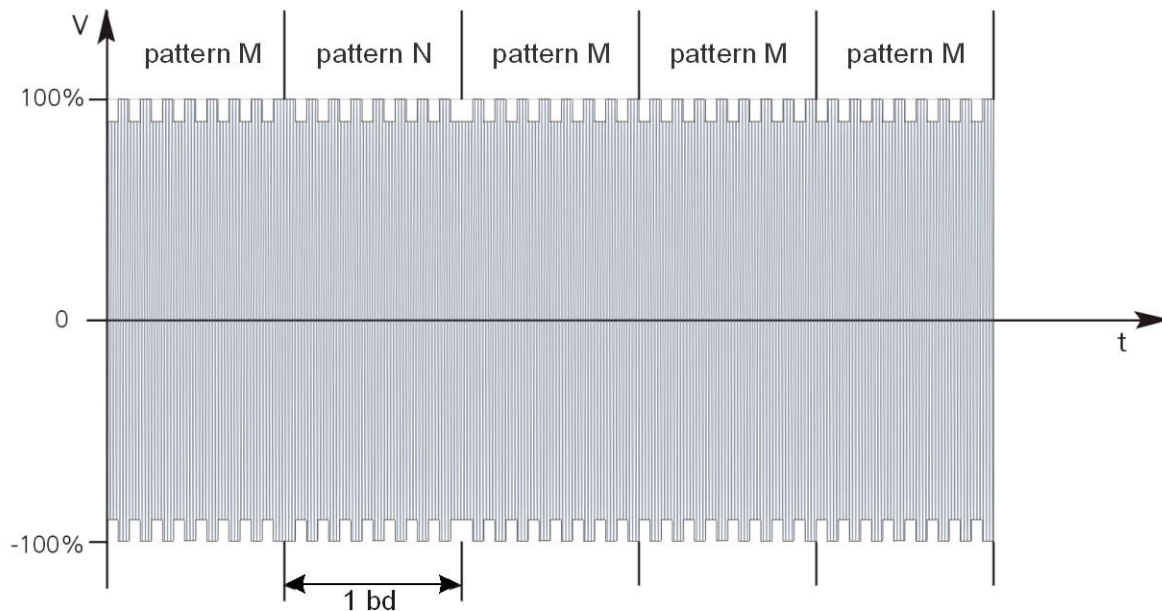
Using the modulation principle of NRZ-L coding with ASK modulation, the carrier amplitude is varied in order to define two particular patterns: L and H.

**Requirements 33: Signal Patterns Poll→Listen – NFC-B**

<b>Poll Mode</b>		<b>Listen Mode</b>	
5.1.1.1	The NFC Forum Device <b>MUST</b> build pattern L and pattern H as follows: <ul style="list-style-type: none"> <li>• Pattern L: carrier low (modulation applied) for the full bit duration</li> <li>• Pattern H: carrier high (no modulation applied) for the full bit duration.</li> </ul>	5.1.1.2	The NFC Forum Device <b>MUST</b> read pattern L and pattern H as follows: <ul style="list-style-type: none"> <li>• If the carrier is low (modulation applied) for the full bit duration, then the NFC Forum Device <b>MUST</b> read this pattern L.</li> <li>• If the carrier is high (no modulation applied) for the full bit duration, then the NFC Forum Device <b>MUST</b> read this as pattern H.</li> <li>• All other patterns <b>MUST</b> be treated as invalid patterns.</li> </ul>

**5.1.2 Listen→Poll Modulation**

In Listen Mode, the analog signal is modulated using NRZ-L with BPSK modulation (see Figure 12).



**Figure 12: NRZ-L Coding with BPSK**

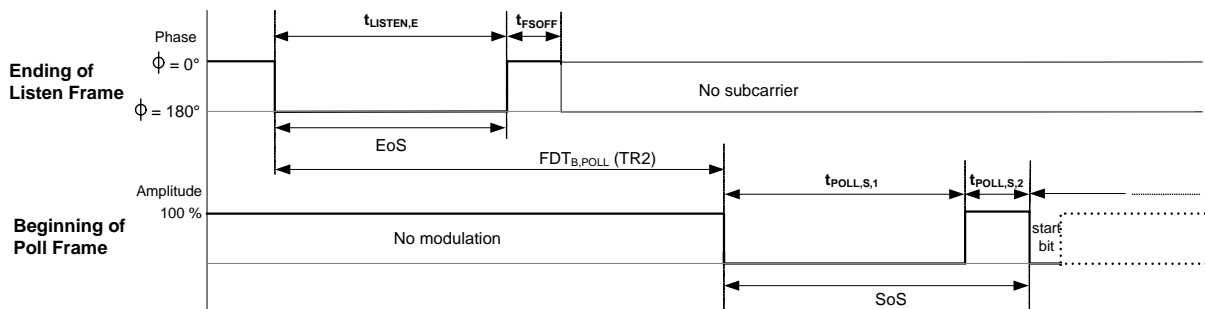
NRZ-L with BPSK modulation uses a phase shift ( $180^\circ$ ) of the subcarrier to define two particular patterns: N and M.

**Requirements 34: Signal Patterns Listen→Poll – NFC-B**

Poll Mode		Listen Mode	
5.1.2.1	The NFC Forum Device <b>MUST</b> read a subcarrier with phase $\emptyset$ for the full bit duration as pattern M. If there is a phase transition, then this marks the beginning of the pattern.	5.1.2.2	The NFC Forum Device <b>MUST</b> build the following as pattern M: <ul style="list-style-type: none"> <li>• A subcarrier with phase <math>\emptyset</math> for the full bit duration.</li> </ul> If this requires a phase transition, the phase transition <b>MUST</b> be at the beginning of the pattern.
5.1.2.3	The NFC Forum Device <b>MUST</b> read a subcarrier with phase $\emptyset 180^\circ$ for the full bit duration as pattern N. If there is a phase transition, then this marks the beginning of the pattern. All patterns different from pattern M and pattern N <b>MUST</b> be treated as invalid patterns.	5.1.2.4	The NFC Forum Device <b>MUST</b> build the following as pattern N: <ul style="list-style-type: none"> <li>• A subcarrier with phase <math>\emptyset 180^\circ</math> for the full bit duration.</li> </ul> If this requires a phase transition, the phase transition <b>MUST</b> be at the beginning of the pattern.

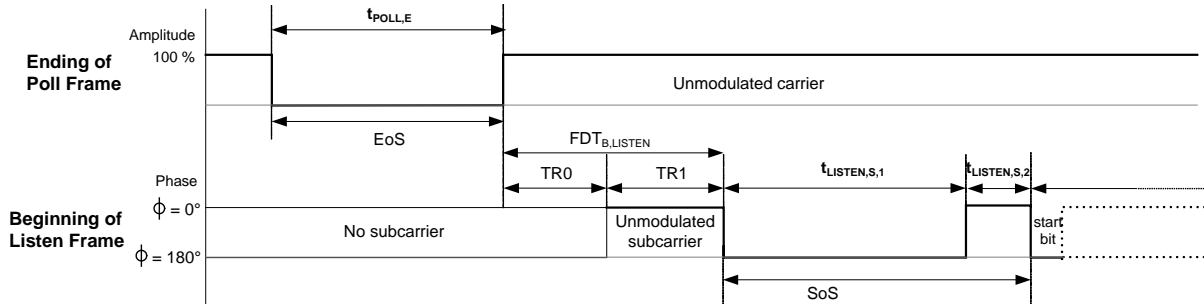
**5.1.3 Synchronization**

Figure 13 and Figure 14 illustrate the different parameters used for NFC-B Technology signal synchronization and related signal timing parameters.



**Figure 13: Synchronization and Timing Parameters between a Listen Frame and a Poll Frame**





**Figure 14: Synchronization and Timing Parameters between a Poll Frame and a Listen Frame**

For the demodulator, the beginning of a signal is indicated by the SoS.

**Requirements 35: Synchronization Poll→Listen – NFC-B**

Poll Mode	Listen Mode
5.1.3.1 The NFC Forum Device MUST code SoS as follows: <ul style="list-style-type: none"> <li>SoS: <math>t_{POLL,S,1}</math> with carrier low (modulation applied), followed by <math>t_{POLL,S,2}</math> with carrier high (no modulation applied)</li> </ul> Refer to Appendix A.2 for the values of $t_{POLL,S,1}$ and $t_{POLL,S,2}$ .	5.1.3.2 The NFC Forum Device MUST decode SoS as follows: <ul style="list-style-type: none"> <li>If the carrier is low (modulation applied) for <math>t_{POLL,S,1}</math>, followed by <math>t_{POLL,S,2}</math> with carrier high (no modulation applied), then the NFC Forum Device MUST decode that as SoS.</li> </ul>

## Requirements 36: Synchronization Listen→Poll – NFC-B

Poll Mode	Listen Mode
<p>5.1.3.3 For establishing phase reference <math>\emptyset</math>, the NFC Forum Device MUST proceed as follows:</p> <ul style="list-style-type: none"> <li>• After any Command from the NFC Forum Device, it MUST ignore any subcarrier generated by the NFC Forum Device in Listen Mode during a time <b>TR0<sub>MIN</sub></b>.</li> <li>• The subcarrier as detected during TR1 MUST be taken as phase reference <math>\emptyset</math>.</li> </ul> <p><i>If a phase transition is detected before <b>TR1<sub>MIN</sub></b>, then the NFC Forum Device MAY resort to exception processing (Transmission Error).</i></p> <p><i>If at <b>TR1<sub>MAX</sub></b>, no phase transition is detected, then the NFC Forum Device MAY resort to exception processing (Transmission Error).</i></p>	<p>5.1.3.4 For establishing phase reference <math>\emptyset</math>, the NFC Forum Device MUST proceed as follows:</p> <ul style="list-style-type: none"> <li>• After any Command from the NFC Forum Device in Poll Mode, a minimum guard time <b>TR0<sub>MIN</sub></b> MUST apply in which the NFC Forum Device in Listen Mode MUST not generate a subcarrier. Refer to Appendix A.2 for the value of <b>TR0<sub>MIN</sub></b>.</li> <li>• Following the guard time TR0, the NFC Forum Device MUST then generate a subcarrier with no phase transition for a synchronization time TR1. This establishes a subcarrier phase reference <math>\emptyset</math>. Refer to Appendix A.2 for the minimum and maximum value of TR1 (<b>TR1<sub>MIN</sub></b> and <b>TR1<sub>MAX</sub></b>).</li> </ul>

Poll Mode		Listen Mode	
5.1.3.5	<p>If, after the synchronization time TR1, the NFC Forum Device detects:</p> <ul style="list-style-type: none"> <li>• a subcarrier phase transition <math>\emptyset 0</math> to <math>\emptyset 0+180^\circ</math></li> <li>• followed by a subcarrier with phase <math>\emptyset 0+180^\circ</math> for <math>t_{LISTEN,S,1}</math></li> <li>• followed by a subcarrier phase transition <math>\emptyset 0+180^\circ</math> to <math>\emptyset 0</math></li> <li>• followed by the subcarrier with phase <math>\emptyset 0</math> for <math>t_{LISTEN,S,2}</math></li> </ul> <p>then the NFC Forum Device MUST decode this as SoS.</p>	5.1.3.6	<p>After the synchronization time TR1, the NFC Forum Device MUST code SoS as follows:</p> <ul style="list-style-type: none"> <li>• subcarrier phase transition <math>\emptyset 0</math> to <math>\emptyset 0+180^\circ</math></li> <li>• followed by a subcarrier with phase <math>\emptyset 0+180^\circ</math> for <math>t_{LISTEN,S,1}</math></li> <li>• followed by a subcarrier phase transition <math>\emptyset 0+180^\circ</math> to <math>\emptyset 0</math></li> <li>• followed by subcarrier with phase <math>\emptyset 0</math> for <math>t_{LISTEN,S,2}</math></li> </ul> <p>Refer to Appendix A.2 for the values of <math>t_{LISTEN,S,1}</math> and <math>t_{LISTEN,S,2}</math>.</p>

### 5.1.4 Pattern Synchronization

Patterns are grouped in a pattern group that are 10 bit durations long. The separation between two pattern groups is defined as the Extra Guard Time (EGT).

#### Requirements 37: NFC-B Pattern Group Separation

Poll and Listen Mode	
5.1.4.1	The time between two consecutive pattern groups sent by the NFC Forum Device in Poll Mode to the NFC Forum Device in Listen Mode MUST be $EGT_{POLL}$ . Refer to Appendix A.2 for the value of $EGT_{POLL}$ .
5.1.4.2	The time between two consecutive pattern groups sent by the NFC Forum Device in Listen Mode to the NFC Forum Device in Poll Mode MUST be $EGT_{LISTEN}$ . Refer to Appendix A.2 for the value of $EGT_{LISTEN}$ .

The separation between two patterns within a pattern group occurs according to the following requirements.

#### Requirements 38: NFC-B Pattern Boundaries

Poll Mode	Listen Mode
5.1.4.3	5.1.4.4
<p>For Poll→Listen communication, pattern boundaries within a pattern group MUST occur between <math>n \text{ bd} \pm 8/f_c</math> where n is the number of pattern boundaries after the start pattern falling edge (<math>1 \leq n \leq 9</math>).</p>	<p>For Listen→Poll communication, pattern boundaries within a pattern group MUST only occur at nominal positions of rising or falling edges of the subcarrier: <math>n \text{ bd} \pm 16/f_c</math>.</p>

### 5.1.5 De-synchronization

The end of a sequence is indicated by the EoS.

**Requirements 39: De-synchronization Poll→Listen – NFC-B**

Poll Mode		Listen Mode	
5.1.5.1	<p>The NFC Forum Device MUST code EoS as follows:</p> <ul style="list-style-type: none"> <li>EoS: <math>t_{\text{POLL,E}}</math> with carrier low (modulation applied), followed by a transition to carrier high.</li> </ul> <p>Refer to Appendix A.2 for the value of <math>t_{\text{POLL,E}}</math>.</p>	5.1.5.2	<p>The NFC Forum Device MUST decode EoS as follows:</p> <ul style="list-style-type: none"> <li>If the carrier is low (modulation applied) for <math>t_{\text{POLL,E}}</math>, followed by a transition to carrier high (no modulation applied), then the NFC Forum Device MUST decode that as EoS.</li> </ul>

**Requirements 40: De-synchronization Listen→Poll – NFC-B**

Poll Mode		Listen Mode	
5.1.5.3	<p>If the NFC Forum Device detects:</p> <ul style="list-style-type: none"> <li>a subcarrier phase transition <math>\emptyset</math> to <math>\emptyset+180^\circ</math></li> <li>followed by the subcarrier with phase <math>\emptyset+180^\circ</math> for <math>t_{\text{LISTEN,E}}</math></li> <li>followed by a subcarrier phase transition <math>\emptyset+180^\circ</math> to <math>\emptyset</math></li> </ul> <p>then the NFC Forum Device MUST decode this as EoS.</p>	5.1.5.4	<p>The NFC Forum Device MUST code the following as EoS:</p> <ul style="list-style-type: none"> <li>a subcarrier phase transition <math>\emptyset</math> to <math>\emptyset+180^\circ</math></li> <li>followed by a subcarrier with phase <math>\emptyset+180^\circ</math> for <math>t_{\text{LISTEN,E}}</math></li> <li>followed by a subcarrier phase transition <math>\emptyset+180^\circ</math> to <math>\emptyset</math></li> </ul> <p>Refer to Appendix A.2 for the value of <math>t_{\text{LISTEN,E}}</math>.</p>
5.1.5.5	<p>The NFC Forum Device MUST accept that the subcarrier is maintained on for a time <math>t_{\text{FSOFF}}</math> after EoS by the NFC Forum Device in Listen Mode.</p> <p><i>After EoS, if the NFC Forum Device in Listen Mode maintains the subcarrier on for a time greater than the maximum value of <math>t_{\text{FSOFF}}</math>, then the NFC Forum Device in Poll Mode MAY resort to exception processing (Transmission Error).</i></p>	5.1.5.6	<p>After EoS, the NFC Forum Device MUST maintain the subcarrier on for a time <math>t_{\text{FSOFF}}</math> and MUST turn the subcarrier off.</p> <p>Refer to Appendix A.2 for the value of <math>t_{\text{FSOFF}}</math>.</p>

## 5.2 Bit Level Coding

### 5.2.1 Poll→Listen Coding Scheme

The patterns L and H are used to code the digital alphabet Logic “0” and Logic “1”.

#### Requirements 41: Bit Level Coding Poll→Listen – NFC-B

Poll Mode		Listen Mode	
5.2.1.1	<p>The NFC Forum Device MUST code Logic “0” and Logic “1” as follows:</p> <ul style="list-style-type: none"> <li>• Logic “0”: pattern L</li> <li>• Logic “1”: pattern H</li> </ul>	5.2.1.2	<p>The NFC Forum Device MUST decode Logic “0” and Logic “1” as follows:</p> <ul style="list-style-type: none"> <li>• If the NFC Forum Device detects pattern L, then it MUST decode this as Logic “0”.</li> <li>• If the NFC Forum Device detects pattern H, then it MUST decode this as Logic “1”.</li> </ul> <p><i>Invalid patterns MAY be treated as Transmission Errors, or the device MAY use implementation dependent methods to decode an invalid pattern into a Logic “0” or Logic “1”.</i></p>

### 5.2.2 Listen→Poll Coding Scheme

Use patterns N and M to code the digital alphabet Logic “0” and Logic “1”.

#### Requirements 42: Bit Level Coding Listen→Poll – NFC-B

Poll Mode		Listen Mode	
5.2.2.1	<p>The NFC Forum Device MUST decode Logic “0” and Logic “1” as follows:</p> <ul style="list-style-type: none"> <li>• If the NFC Forum Device detects pattern N, then it MUST decode this as Logic “0”.</li> <li>• If the NFC Forum Device detects pattern M, then it MUST decode this as Logic “1”.</li> </ul> <p><i>Invalid patterns MAY be treated as Transmission Errors or the device MAY use implementation dependent methods to decode an invalid pattern into a Logic “0” or Logic “1”.</i></p>	5.2.2.2	<p>The NFC Forum Device MUST code Logic “0” and Logic “1” as follows:</p> <ul style="list-style-type: none"> <li>• Logic “0”: pattern N</li> <li>• Logic “1”: pattern M</li> </ul>

### 5.3 Frame Format

To transmit data, the NFC Forum Device configured for NFC-B Technology uses frames that are built from characters. This section defines the format of characters and frames.

A character consists of a Logic “0” start bit, a Logic “1” stop bit and eight data bits. The stop bit, start bit, and each data bit has a length of one bit duration (bd\*\*). Figure 15 shows the character format.

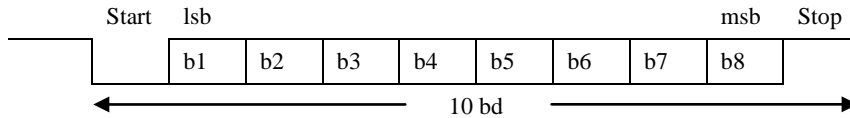


Figure 15: NFC-B – Character Format

#### Requirements 43: Character Format – NFC-B Technology

---

##### Poll and Listen Mode

---

5.3.1.1 A character MUST consist of a start bit (Logic “0”), 8 data bits, and a stop bit (Logic “1”). The stop bit, start bit, and each data bit MUST be one bit duration (bd).

---

Characters are sent as frames, as shown in Figure 16.

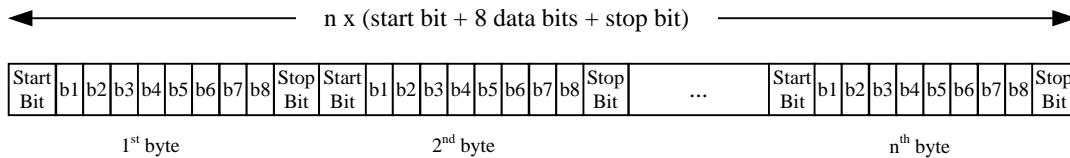


Figure 16: NFC-B – Frame Format

#### Requirements 44: Frame Format – NFC-B Technology

---

##### Poll and Listen Mode

---

5.3.1.2 A frame MUST consist of characters that are aligned as shown in Figure 16.

---

NOTE The frame format defined in Figure 16 deviates from [ISO/IEC\_14443]. SoF and EoF as defined in [ISO/IEC\_14443] are defined by this specification as SoS and EoS at the sequence layer.

### 5.4 Data and Payload Format

Data transmitted in an NFC-B frame have the following substructure. They consist of the payload and of an EoD (SoD is not used).

The payload consists of the Commands and Responses described in Section 5.5.

The EoD contains a 2-byte checksum referred to as CRC\_B. The CRC\_B is a function of k data bits, which is a multiple of 8. Input for CRC\_B calculation is the payload.

Figure 17 illustrates the NFC-B data and payload format.

Data					
Payload (Command or Response)					EoD
Byte 1	Byte 2	...	Byte n	CRC_B1	CRC_B2

**Figure 17: Data and Payload Format – NFC-B**

**Requirements 45: Data and Payload Format – NFC-B**

<b>Poll and Listen Mode</b>	
5.4.1.1	Data MUST be transmitted in frames as defined in Section 5.3.
5.4.1.2	The payload MUST be followed by an EoD at the position, as indicated in Figure 17. The EoD MUST contain a CRC_B.
5.4.1.3	The CRC_B MUST be calculated as defined in [ISO/IEC_13239]. The initial register content MUST be all ones (FFFFh). CRC_B1 is the LSB and CRC_B2 is the MSB.
5.4.1.4	The NFC Forum Device MUST compare the received CRC_B with the CRC_B calculated according to Requirement 5.4.1.3. If different, then the NFC Forum Device MUST resort to exception processing with a Transmission Error.

## 5.5 Command Set

Payload exchanged between NFC Forum Devices consists of Commands and Responses. Table 21 lists the Commands that are supported by the NFC Forum Device configured for NFC-B Technology. For each Command, the corresponding Response is indicated.

**Table 21: NFC-B – Command Set**

<b>Command</b>	<b>Response</b>
ALLB_REQ, SENSB_REQ	SENSB_RES
SLOT_MARKER (optional for Poll Mode)	SENSB_RES
SLPB_REQ	SLPB_RES

NOTE The Commands and Responses listed in Table 21 have different names in [ISO/IEC\_14443]:  
ALLB\_REQ refers to WUPB, SENSB\_REQ refers to REQB, SENSB\_RES refers to ATQB, SLPB\_REQ and SLPB\_RES refer to HLTB and its Response.

## 5.6 ALLB\_REQ and SENSB\_REQ

The ALLB\_REQ and SENSB\_REQ Commands are used by an NFC Forum Device in Poll Mode to probe the Operating Field for NFC Forum Devices in Listen Mode.

### 5.6.1 ALLB\_REQ and SENSB\_REQ Command

Table 22 defines the format of the ALLB\_REQ and SENSB\_REQ Commands.

**Table 22: ALLB\_REQ and SENSB\_REQ Command Format**

Byte 1	Byte 2	Byte 3
05h	AFI	PARAM

The components of this Command are defined as follows:

**AFI**

The AFI indicates the application family being selected.

**Requirements 46: AFI**

Poll Mode	Listen Mode
5.6.1.1 The AFI MUST be set to 00h. This selects all application families.	<i>The NFC Forum Device MAY support an ALLB_REQ and a SENSB_REQ with AFI different from 00h.</i>

**PARAM**

Table 23 specifies the format of PARAM.

**Table 23: Format of PARAM Byte Included in ALLB\_REQ and SENSB\_REQ Command**

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	0							RFU
		x						0b: Advanced protocol features not supported 1b: Advanced protocol features supported
			x					0b: Extended SENSB_RES not supported 1b: Extended SENSB_RES supported
				x				0b: SENSB_REQ 1b: ALLB_REQ
					x	x	x	Number of slots (N)

The Number of slots (N) is used by the anticollision scheme defined in [ISO/IEC\_14443]. The anticollision scheme is based on the definition of time slots in which NFC Forum Devices in Listen Mode are invited to respond. Table 24 specifies the coding of N.



**Table 24: Coding of N**

<b>b3</b>	<b>b2</b>	<b>b1</b>	<b>Meaning</b>
0	0	0	N = 1
0	0	1	N = 2
0	1	0	N = 4
0	1	1	N = 8
1	0	0	N = 16
1	0	1	RFU
1	1	x	RFU

**Requirements 47: Number of Slots (N)**

<b>Poll Mode</b>		<b>Listen Mode</b>	
5.6.1.2	If the NFC Forum Device does not support collision resolution, it MUST set N to 1. If the NFC Forum Device supports collision resolution, it MUST set N to a value as specified in Table 24.	5.6.1.3	The NFC Forum Device MUST be ready to receive a value of N set to 1. For collision resolution, it MUST be ready to receive a value of N as specified in Table 24.

**Requirements 48: Listen Mode Handling of RFU Values of N**

<b>Listen Mode</b>	
5.6.1.4	A received RFU value of N MUST be treated by the NFC Forum Device in Listen Mode as N=16.

Bit 5 indicates whether the NFC Forum Device in Poll Mode supports the extended SENS<sub>B</sub>\_RES byte. The extended SENS<sub>B</sub>\_RES byte is an optional byte included in the SENS<sub>B</sub>\_RES Response, coding the SFGI used by the NFC Forum Device in Listen Mode.

Bit 6 indicates whether the NFC Forum Device in Poll Mode supports advanced protocol features (e.g., higher bit rates).

**Requirements 49: Support for Extended SENSB\_RES**

Poll Mode		Listen Mode	
5.6.1.5	The NFC Forum Device MUST be ready to receive a SENSB_RES Response with or without an extended SENSB_RES byte included.	5.6.1.6	When answering an ALLB_REQ or SENSB_REQ Command with bit b5 set to 0b, indicating that the extended SENSB_RES is not supported, the NFC Forum Device MUST not include the extended SENSB_RES byte in its SENSB_RES Response.  <i>When answering an ALLB_REQ or SENSB_REQ Command with b5 set to 1b, indicating that the extended SENSB_RES is supported, the NFC Forum Device MAY or MAY not include the extended SENSB_RES byte in its SENSB_RES Response.</i>

**5.6.2 SENSB\_RES Response**

Table 25 defines the SENSB\_RES format.

**Table 25: SENSB\_RES Format**

Byte 1	Byte 2 - 5	Byte 6 - 9	Byte 10 – 12 or 13
50h	NFCID0	Application Data	Protocol Info

Byte 13 (i.e., the extended SENSB\_RES byte of the SENSB\_RES Response) is optional. If it is not used, then the SENSB\_RES consists of only 12 bytes.

**NFCID0**

The NFCID0 is used to differentiate between NFC Forum Devices in Listen Mode during the Collision Resolution Activity.

**Requirements 50: NFCID0 in SENSB\_RES**

Listen Mode	
5.6.2.1	The NFCID0 MUST have a length of 4 bytes. The value of the NFCID0 MUST be fixed or dynamically generated by the NFC Forum Device.
5.6.2.2	A dynamically generated NFCID0 MUST be generated only on state transition from <b>POWER-OFF</b> to <b>IDLE</b> state (see [ACTIVITY] for the NFC Forum Device state machine).

**Application Data**

The Application Data Field is used to inform the NFC Forum Device in Poll Mode which applications are installed on the NFC Forum Device in Listen Mode.

The Application Data is defined according to the ADC in the Protocol Info, which defines if either the CRC\_B compressing method (described below) or proprietary coding is used.

When the CRC\_B compressing coding is used, the Application Data field contains the information shown in Table 26:

**Table 26: Application Data Format**

Byte 1	Byte 2 and 3	Byte 4
AFI	CRC_B (AID)	Number of applications

### Protocol Info

The Protocol Info indicates the parameters supported by the NFC Forum Device in Listen Mode, as defined in Table 27.

**Table 27: Protocol Info Format**

Byte 1	Byte 2		Byte 3			Byte 4 (optional)	
Bit_Rate_Capability (8 bits)	FSCI (4 bits)	Protocol_Type (4 bits)	FWI (4 bits)	ADC (2 bits)	FO (2 bits)	SFGI (4 bits)	RFU (4 bits)

- Bit\_Rate\_Capability

Table 28 specifies the bit rates supported by the NFC Forum Device in Listen Mode. Bits b7 to b5 code the bit rate capability of the NFC Forum Device in Listen Mode for the direction from NFC Forum Device in Listen Mode to NFC Forum Device in Poll Mode ( $D_{LISTEN \rightarrow POLL}$ ). The value 000b corresponds with  $D_{LISTEN \rightarrow POLL} = 1$ . The bits b3 to b1 code the bit rate capability of the NFC Forum Device in Listen Mode for the direction from NFC Forum Device in Poll Mode to NFC Forum Device in Listen Mode ( $D_{POLL \rightarrow LISTEN}$ ). The value 000b corresponds with  $D_{POLL \rightarrow LISTEN} = 1$ .

**Table 28: Bit Rates Supported by the NFC Forum Device in Listen Mode**

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
x								If b8 = 1b, then only the same bit rate divisor for both directions is supported ( $D_{LISTEN \rightarrow POLL} = D_{POLL \rightarrow LISTEN}$ ). If b8 = 0b, then a different bit rate divisor for each direction is supported.
	x							$D_{LISTEN \rightarrow POLL} = 8$ supported, if bit is set to 1b.
		x						$D_{LISTEN \rightarrow POLL} = 4$ supported, if bit is set to 1b.
			x					$D_{LISTEN \rightarrow POLL} = 2$ supported, if bit is set to 1b.
				0				RFU
					x			$D_{POLL \rightarrow LISTEN} = 8$ supported, if bit is set to 1b.
						x		$D_{POLL \rightarrow LISTEN} = 4$ supported, if bit is set to 1b.
							x	$D_{POLL \rightarrow LISTEN} = 2$ supported, if bit is set to 1b.
0	0	0	0	0	0	0	0	The NFC Forum Device supports only a bit rate of 106 kbits/s in both directions.

**Requirements 51: Bit Rates Supported by the NFC Forum Device in Listen Mode**

Poll Mode		Listen Mode	
5.6.2.3	The NFC Forum Device <b>MUST</b> support a bit rate of 106 kbits/s in both directions.  <i>The NFC Forum Device MAY support higher bit rates.</i>	5.6.2.4	In response to a SENS <sub>B</sub> _REQ and ALL <sub>B</sub> _REQ with b6 of PARAM byte set to 0b, the NFC Forum Device <b>MUST</b> set bits b7 to b5 and b3 to b1 of the Bit_Rate_Capability equal to 0b, indicating that it supports only a bit rate of 106 kbits/s in both directions.  <i>The NFC Forum Device MAY indicate support for higher bit rates in response to a SENS<sub>B</sub>_REQ and ALL<sub>B</sub>_REQ with b6 of PARAM set to 1b.</i>

**Requirements 52: Poll Mode Handling of RFU Value of b4 of Bit\_Rate\_Capability**

Listen Mode	
5.6.2.5	A received RFU value of b4 = 1b <b>MUST</b> be interpreted by the NFC Forum Device in Poll Mode as if b7 to b1 = 0000000b (only 106 kbits/s in both directions).

NOTE Higher bit rates will be specified in next versions of this specification.

- FSCI

FSCI contains the maximum frame size (FSC), coded as indicated in Table 29. The FSC defines the maximum size of a frame (in bytes) accepted by the NFC Forum Device in Listen Mode.

**Table 29: FSCI to FSC Conversion**

FSCI	0h	1h	2h	3h	4h	5h	6h	7h	8h	9h-Fh
FSC (bytes)	16	24	32	40	48	64	96	128	256	RFU

**Requirements 53: FSC**

Poll Mode		Listen Mode	
5.6.2.6	The NFC Forum Device MUST send frames with a number of data bytes less than or equal to FSC.	5.6.2.7	The NFC Forum Device MUST accept frames with a number of data bytes less than or equal to FSC. The NFC Forum Device MUST resort to exception processing (Protocol Error) if it receives a frame with more than FSC data bytes.
5.6.2.8	The NFC Forum Device MUST be capable of sending frames in accordance with an FSC greater than or equal to <b>FSC<sub>MIN</sub></b> .	5.6.2.9	The FSC supported by the NFC Forum Device MUST be at least <b>FSC<sub>MIN</sub></b> . Refer to Appendix A.2 for the value of <b>FSC<sub>MIN</sub></b> .

**Requirements 54: FSCI**

Listen Mode	
5.6.2.10	The NFC Forum Device MUST set FSCI greater than or equal to the value corresponding to <b>FSC<sub>MIN</sub></b> with a maximum value of 8h. Refer to Appendix A.2 for the value of <b>FSC<sub>MIN</sub></b> . Refer to Table 29 for FSCI to FSC conversion.

**Requirements 55: Poll Mode Handling of RFU Values of FSCI**

Poll Mode	
5.6.2.11	A received value of FSCI = 9h–Fh MUST be treated by the NFC Forum Device as FSCI = 8h.

- Protocol\_Type

Protocol\_Type indicates the protocol type and the minimum TR2 (see Figure 13) supported by the NFC Forum Device in Listen Mode, as shown in Table 30.

**Table 30: Protocol\_Type**

b4	b3	b2	b1	Meaning
0				RFU
	x	x		Minimum TR2
			x	0b: NFC Forum Device in Listen Mode not compliant with [ISO/IEC_14443] 1b: NFC Forum Device in Listen Mode compliant with [ISO/IEC_14443]

**Requirements 56: Protocol Type Supported by the NFC Forum Device in Listen Mode**

Poll Mode		Listen Mode	
5.6.2.12	The NFC Forum Device MUST be ready to receive the Protocol_Type with bit b1 set to 1b.	5.6.2.13	The NFC Forum Device MUST indicate support for [ISO/IEC_14443] by setting bit b1 to 1b.
<i>The NFC Forum Device in Poll Mode MAY support an NFC Forum Device in Listen Mode not indicating conformity to [ISO/IEC_14443].</i>			

Protocol\_Type bits b3,b2 define the minimum value of TR2 supported by the NFC Forum Device in Listen Mode, as specified in Table 31.

**Table 31: Minimum TR2 Coding**

b3	b2	Meaning
0	0	$1792/f_C$ (10 etu + 32/fs)
0	1	$3328/f_C$ (10 etu + 128/fs)
1	0	$5376/f_C$ (10 etu + 256/fs)
1	1	$9472/f_C$ (10 etu + 512/fs)

**Requirements 57: Minimum TR2 Coding**

Poll Mode		Listen Mode	
5.6.2.14	The NFC Forum Device MUST disregard any value returned in bits b3 and b2 in response to a SENS <sub>B</sub> _REQ and ALL <sub>B</sub> _REQ with b6 of PARAM set to 0b. In this case, the NFC Forum Device MUST always use <b>TR2<sub>MIN,DEFAULT</sub></b> as minimum TR2. Refer to Appendix A.2 for the value of <b>TR2<sub>MIN,DEFAULT</sub></b> . The NFC Forum Device MUST use the minimum TR2 value defined by b3 and b2 in response to a SENS <sub>B</sub> _REQ and ALL <sub>B</sub> _REQ with b6 of PARAM set to 1b.	5.6.2.15	In response to a SENS <sub>B</sub> _REQ and ALL <sub>B</sub> _REQ with b6 of PARAM set to 0b, the NFC Forum Device MUST set bits b3,b2 equal to 00b. <i>The NFC Forum Device MAY set b3 and b2 different from 00b in response to a SENS<sub>B</sub>_REQ and ALL<sub>B</sub>_REQ with b6 of PARAM set to 1b.</i>

**Requirements 58: Poll Mode Handling of RFU Value of b4 of Protocol\_Type**

Poll Mode	
5.6.2.16	The NFC Forum Device in Poll Mode MUST not continue communicating with an NFC Forum Device in Listen Mode that sets b4 to 1b.

- FWI – Frame Waiting Time Integer (4 bits)

The FWI codes an integer value used to define the Frame Waiting Time (FWT). The FWT is the maximum time an NFC Forum Device in Listen Mode is allowed to wait before sending the Listen Frame after the end of a Poll Frame. The FWT is calculated as specified in Section 5.9.1. FWI has a value in the range from 1 to 14. The value 15 is RFU.

**Requirements 59: Maximum Value of FWI**

Poll Mode		Listen Mode	
5.6.2.17	The NFC Forum Device in Poll Mode MUST support an FWI less than or equal to <b>FWI<sub>MAX</sub></b> .	5.6.2.18	The NFC Forum Device MUST set FWI less than or equal to <b>FWI<sub>MAX</sub></b> . Refer to Appendix A.2 for the value of <b>FWI<sub>MAX</sub></b> .

**Requirements 60: Poll Mode Handling of RFU Value of FWI**

Poll Mode	
5.6.2.19	The NFC Forum Device MUST treat a received value of FWI = 15 as FWI = 4.

- ADC

The ADC represents the Application Data Coding supported by the NFC Forum Device in Listen Mode and is coded as specified in Table 32.

**Table 32: ADC Coding**

<b>b4</b>	<b>b3</b>	<b>Meaning</b>
x		0b: Advanced protocol features not supported 1b: Advanced protocol features supported
	x	0b: Application is proprietary 1b: Application is coded as described in Table 26.

- FO

Table 33 and Table 34 indicate the Frame Options (FO) supported by the NFC Forum Device in Listen Mode. FO indicates support for NAD and DID.

**Table 33: FO – NAD**

<b>b2</b>	<b>Meaning</b>
x	NAD supported, if bit is set to 1b.

**Requirements 61: NAD**

<b>Poll Mode</b>		<b>Listen Mode</b>	
5.6.2.20	The NFC Forum Device MUST not use NAD and MUST disregard any value returned by the NFC Forum Device in Listen Mode in b2.	5.6.2.21	The NFC Forum Device MUST not support NAD and MUST set b2 equal to 0b.

**Table 34: FO – DID**

<b>b1</b>	<b>Meaning</b>
x	Supports DID, if bit is set to 1b.

- SFGI – Start-up Frame Guard Time Integer (4 bits)

The most significant nibble b8 to b5 of the optional Byte 4 contains SFGI and is used by the NFC Forum Device in Listen Mode to define the SFGT. Refer to Section 5.9.2 for the definition of SFGT. SFGI has a value in the range from 1 to 14. The value 15 is RFU. The default value of SFGI is 0.



**Requirements 62: SFGI**

<b>Poll Mode</b>		<b>Listen Mode</b>	
5.6.2.22	The NFC Forum Device in Poll Mode MUST support an SFGI less than or equal to <b>SFGI<sub>MAX</sub></b> .	5.6.2.23	The NFC Forum Device MUST set SFGI less than or equal to <b>SFGI<sub>MAX</sub></b> . Refer to Appendix A.2 for the value of <b>SFGI<sub>MAX</sub></b> .

**Requirements 63: Poll Mode Handling of RFU Value of SFGI**

<b>Poll Mode</b>	
5.6.2.24	A received value of SFGI = 15 MUST be treated by the NFC Forum Device as SFGI = 0.

**5.7 SLOT\_MARKER**

The SLOT\_MARKER Command is used by an NFC Forum Device in Poll Mode during collision resolution to define the start of the Response time slot for an NFC Forum Device in Listen Mode.

NOTE Implementation of the SLOT\_MARKER Command is optional.

**5.7.1 SLOT\_MARKER Command**

Table 35 defines the format of the SLOT\_MARKER Command.

**Table 35: SLOT\_MARKER Command Format**

<b>Byte 1</b>
APn

The Anticollision Prefix byte (APn) is formatted as follows:

b4-b1 are set to 0101b.

b8-b5 code the slot number, as specified in Table 36.

**Table 36: Coding of Slot Number**

<b>b8</b>	<b>b7</b>	<b>b6</b>	<b>b5</b>	<b>Meaning</b>
0	0	0	0	Not allowed
0	0	0	1	Slot number 2
0	0	1	0	Slot number 3
0	0	1	1	Slot number 4
.....	.....	.....	.....	.....
1	1	1	0	Slot number 15
1	1	1	1	Slot number 16

## 5.7.2 SLOT\_MARKER Response

The Response to the SLOT\_MARKER Command is the SENSB\_RES Response, as specified in Section 5.6.2.

## 5.8 SLPB\_REQ

The SLPB\_REQ Command is used to set an NFC Forum Device in Listen Mode, identified by its NFCID0, to the **SLEEP** state. Refer to [ACTIVITY] for the NFC Forum Device state machine.

### 5.8.1 SLPB\_REQ Command

Table 37 defines the format of the SLPB\_REQ Command.

**Table 37: Format of the SLPB\_REQ Command**

Byte 1	Byte 2 – 5
50h	NFCID0

### 5.8.2 SLPB\_RES Response

Table 38 defines the format of the SLPB\_RES Response.

**Table 38: SLPB\_RES Response Format**

Byte 1
00h

## 5.9 Timing Requirements

This section specifies the requirements for the different Frame Delay Times for NFC-B Technology. The Frame Delay Time (FDT) is the time between two frames transmitted in opposite directions.

### 5.9.1 Frame Delay Time Poll→Listen

The Frame Delay Time Poll→Listen (**FDT<sub>B,LISTEN</sub>**) is the time between a Poll Frame and a Listen Frame, as illustrated in Figure 13. The time between the minimum value **FDT<sub>B,LISTEN,MIN</sub>** and the maximum value **FDT<sub>B,LISTEN,MAX</sub>** defines the time interval in which a Listen Frame is allowed to be sent by an NFC Forum Device in Listen Mode in response to a Poll Frame of an NFC Forum Device in Poll Mode.

$FDT_{B,LISTEN,MIN}$  is the minimum time an NFC Forum Device in Listen Mode has to wait before sending the Listen Frame after the end of a Poll Frame.  $FDT_{B,LISTEN,MIN}$  is defined:

$$FDT_{B,LISTEN,MIN} = TR0_{MIN} + TR1_{MIN}$$

**Requirements 64:  $FDT_{B,LISTEN,MIN}$**

Poll Mode		Listen Mode	
5.9.1.1	The NFC Forum Device MUST be ready to receive the SoS of a Listen Frame no later than $FDT_{B,LISTEN,MIN}$ after the EoS of a Poll Frame.	5.9.1.2	Following the EoS of a Poll Frame, the NFC Forum Device MUST wait at least $FDT_{B,LISTEN,MIN}$ (as defined by $TR0_{MIN} + TR1_{MIN}$ ) before sending the SoS of its Listen Frame. Refer to Appendix A.2 for the values of $TR0_{MIN}$ and $TR1_{MIN}$ .

$FDT_{B,LISTEN,MAX}$  is the maximum time an NFC Forum Device in Listen Mode is allowed to wait before sending the Listen Frame after the end of a Poll Frame. This parameter is also called Frame Waiting Time (FWT). The FWT is calculated by the following formula:

$$FWT = (256 \times 16/f_c) \times 2^{FWI}$$

where, by definition, the integer value of FWI can have a maximum range from 0 to 14. In the SENSB\_RES Response, the NFC Forum Device in Listen Mode informs the NFC Forum Device in Poll Mode of the FWI value. Refer to Section 5.6.2 for the SENSB\_RES Response format.

## Requirements 65: Frame Waiting Time

Poll Mode	Listen Mode
<p>5.9.1.3 The NFC Forum Device <b>MUST</b> wait at least <math>FWT + \Delta FWT</math> for the SoS of a Listen Frame after the EoS of a Poll Frame (except for ALLB_REQ and SENSB_REQ). If the NFC Forum Device does not receive the SoS of a Listen Frame within <math>FWT + \Delta FWT + \Delta T_{POLL}</math>, then the NFC Forum Device <b>MUST</b> treat this as a Timeout Error (i.e., <math>FDT_{B,LISTEN,MAX}</math> equals <math>FWT + \Delta FWT + \Delta T_{POLL}</math>). Refer to Appendix A.2 for the values of <math>\Delta T_{POLL}</math> and <math>\Delta FWT</math>.</p> <p><i>Between <math>FWT + \Delta FWT</math> and <math>FWT + \Delta FWT + \Delta T_{POLL}</math> the NFC Forum Device <b>MAY</b> accept the SoS of a Listen Frame or <b>MAY</b> generate a Timeout Error.</i></p>	<p>5.9.1.4 Following the EoS of a Poll Frame, the NFC Forum Device <b>MUST</b> wait no longer than <math>FWT</math> before sending the SoS of its Listen Frame (except for ALLB_REQ and SENSB_REQ; i.e., <math>FDT_{B,LISTEN,MAX}</math> equals <math>FWT</math>).</p>
<p>5.9.1.5 In the case of the ALLB_REQ, SENSB_REQ, and SLOT_MARKER Commands, the NFC Forum Device <b>MUST</b> wait <math>FWT_{SENSEB}</math> for the SoS of the Listen Frame after the EoS of the Poll Frame. If the NFC Forum Device does not receive the SoS of the Listen Frame within <math>FWT_{SENSEB} + \Delta T_{POLL}</math>, then the NFC Forum Device <b>MUST</b> treat this as a Timeout Error (i.e., <math>FDT_{B,LISTEN,MAX}</math> equals <math>FWT_{SENSEB} + \Delta T_{POLL}</math>). Refer to Appendix A.2 for the values of <math>FWT_{SENSEB}</math> and <math>\Delta T_{POLL}</math>.</p> <p><i>Between <math>FWT_{SENSEB}</math> and <math>FWT_{SENSEB} + \Delta T_{POLL}</math> the NFC Forum Device <b>MAY</b> accept the SoS of a Listen Frame or <b>MAY</b> generate a Timeout Error.</i></p>	<p>5.9.1.6 In the case of the ALLB_REQ, SENSB_REQ, and SLOT_MARKER Commands, following the EoS of the Poll Frame, the NFC Forum Device <b>MUST</b> wait no longer than <math>FWT_{SENSEB}</math> before sending the SoS of the Listen Frame (i.e., <math>FDT_{B,LISTEN,MAX}</math> equals <math>FWT_{SENSEB}</math>. Refer to Appendix A.2 for the values of <math>FWT_{SENSEB}</math>.</p>

The NFC Forum Device in Listen Mode is not allowed to produce any detectable disturbance during a minimum time period  $t_{nn,min}$  before sending a Response.

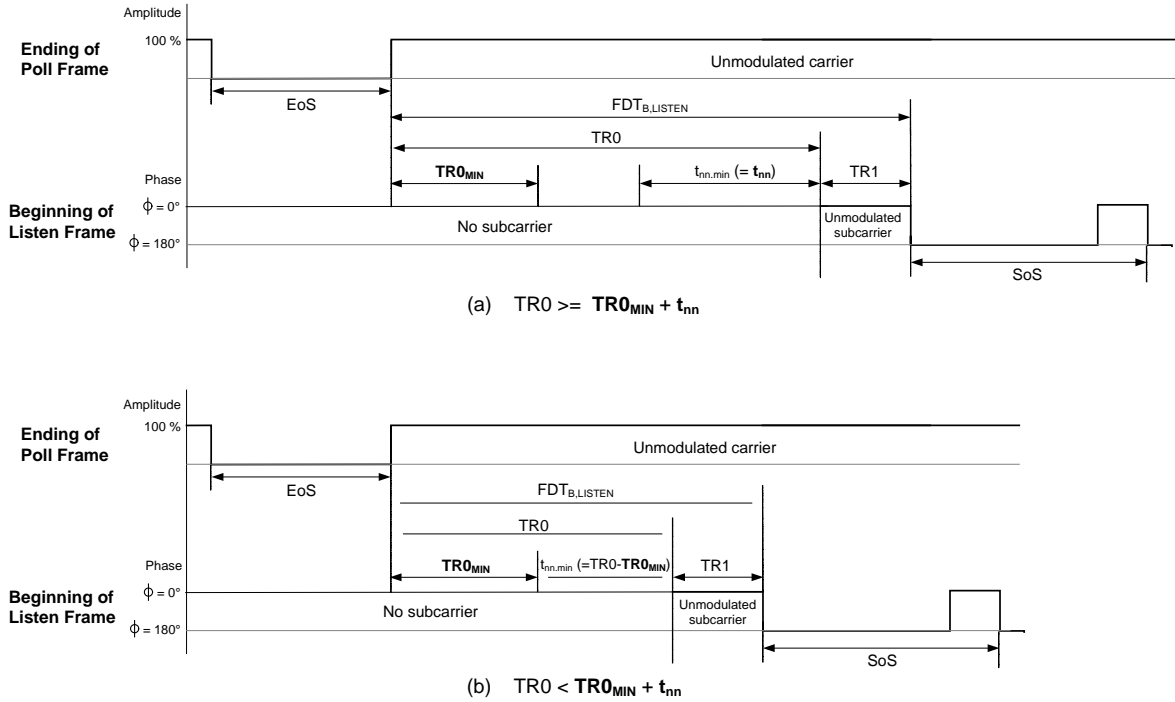


Figure 18:  $t_{nn,min}$  for NFC-B

Requirements 66:  $t_{nn,min}$  - NFC-B

Listen Mode

5.9.1.7 The NFC Forum Device MUST not produce any detectable disturbance during a time period of at least  $t_{nn,min}$ , defined as the minimum of  $TR0 - TR0_{MIN}$  and  $t_{nn}$ , before sending a Response.  $t_{nn,min}$  MUST be measured before the start of  $TR1$ , as shown in Figure 18. For  $TR0_{MIN}$ , the Poll Mode value MUST be used. Refer to Appendix A.2 for the value of  $t_{nn}$ .

*The exact quantity of what constitutes ‘any detectable disturbance’ is not defined by this version of the specification.*

5.9.2 Frame Delay Time Listen→Poll

The Frame Delay Time Listen→Poll ( $FDT_{B,POLL}$ ) is the time between a Listen Frame and a Poll Frame, as shown in Figure 13. The minimum value  $FDT_{B,POLL,MIN}$  defines the time an NFC Forum Device in Poll Mode has to wait before sending a new Poll Frame after receipt of a Listen Frame. A maximum value  $FDT_{B,POLL,MAX}$  is not defined.  $FDT_{B,POLL,MIN}$  is defined:

$$FDT_{B,POLL,MIN} = TR2_{MIN}$$

$TR2_{MIN}$  is the minimum value of  $TR2$ , as shown in Figure 13. The value of  $TR2_{MIN}$ , supported by the NFC Forum Device in Listen Mode, is indicated in the Protocol\_Type of the SENS\_B\_RES Response as specified in Table 31 in Section 5.6.

**Requirements 67:  $FDT_{B,POLL,MIN}$** 

<b>Poll Mode</b>	<b>Listen Mode</b>
5.9.2.1 Following the end of a Listen Frame, the NFC Forum Device MUST wait at least for <b><math>FDT_{B,POLL,MIN}</math></b> before transmitting the SoS of a new Command.	5.9.2.2 The NFC Forum Device MUST be ready to receive the SoS of a new Command no later than <b><math>FDT_{B,POLL,MIN}</math></b> after the end of a Listen Frame.  <i>If the SoS of a new Command is received before <b><math>FDT_{B,POLL,MIN}</math></b>, then the NFC Forum Device MAY treat this as a Transmission Error.</i>

A specific value for  **$FDT_{B,POLL,MIN}$**  is defined after the ATTRIB Response. The SFGT is the minimum time the NFC Forum Device in Poll Mode, configured for reading Type 4B Tag platform, is waiting before the (NFC Forum Device in Listen Mode, configured for emulating) Type 4B Tag platform is ready to receive the next frame after it has sent the ATTRIB Response.

The SFGT is calculated by the following formula:

$$SFGT = (256 \times 16/f_c) \times 2^{SFGI}$$

where SFGI has a range from 1 to 14 and is returned by the NFC Forum Device in Listen Mode in the in the SENSB\_RES Response. Refer to Section 5.6.2 for the definition of the SENSB\_RES Response. If the NFC Forum Device in Listen Mode returns SFGI equal to zero or SFGI is not returned, then no SFGT is needed and  **$FDT_{B,POLL,MIN}$**  applies.

**Requirements 68: SFGT – NFC-B**

<b>Poll Mode</b>	<b>Listen Mode</b>
5.9.2.3 If the NFC Forum Device in Listen Mode returns an SFGI different from zero, the NFC Forum Device in Poll Mode MUST wait at least <b>SFGT+<math>\Delta</math>SFGT</b> before sending the next frame after the NFC Forum Device in Listen Mode has sent the ATTRIB Response. Refer to Appendix A.2 for the value of <b><math>\Delta</math>SFGT</b> .	5.9.2.4 If the NFC Forum Device returns an SFGI other than zero, then the NFC Forum Device MUST be ready to receive the start of a new Poll Frame no later than SFGT after the end of the ATTRIB Response frame.  <i>If the start of a new Poll Frame is received before SFGT, then the NFC Forum Device MAY treat this as a Transmission Error.</i>
5.9.2.5 If the NFC Forum Device in Listen Mode returns an SFGI equal to zero or SFGI is not returned, the NFC Forum Device in Poll Mode MUST wait at least <b>FDT<sub>B,POLL,MIN</sub></b> before sending the next frame after the NFC Forum Device in Listen Mode has sent the ATTRIB Response.	5.9.2.6 If the NFC Forum Device returns an SFGI equal to zero or does not return an SFGI, then the NFC Forum Device MUST be ready to receive the start of a new Poll Frame no later than <b>FDT<sub>B,POLL,MIN</sub></b> after the end of the ATTRIB Response frame.  <i>If the start of a new Poll Frame is received before <b>FDT<sub>B,POLL,MIN</sub></b>, then the NFC Forum Device MAY treat this as a Transmission Error.</i>

A specific value for **FDT<sub>B,POLL,MIN</sub>** is defined for the re-activation of the NFC Forum Device in Listen Mode following an S(DESELECT) Response (see Section 13.2.7).

**Requirements 69: FDT<sub>B,REACTIVATION</sub>**

<b>Poll Mode</b>	<b>Listen Mode</b>
5.9.2.7 Following the end of an S(DESELECT) Response, the NFC Forum Device MUST wait at least <b>FDT<sub>B,REACTIVATION</sub></b> before transmitting the start of a new Poll Frame.  Refer to Appendix A.2 for the value of <b>FDT<sub>B,REACTIVATION</sub></b> .	5.9.2.8 The NFC Forum Device MUST be ready to receive the start of a new Poll Frame no later than <b>FDT<sub>B,REACTIVATION</sub></b> after the end of an S(DESELECT) Response.  <i>If the start of a new Poll Frame is received before <b>FDT<sub>B,REACTIVATION</sub></b>, then the NFC Forum Device MAY treat this as a Transmission Error.</i>

**5.9.3 Guard Time**

This section specifies the guard time of an Unmodulated Carrier, after which the NFC Forum Device in Listen Mode must be ready to receive an ALLB\_REQ or SENSB\_REQ Command.

**Requirements 70: Guard Time – NFC-B**

---

**Listen Mode**

---

- 5.9.3.1 When an NFC Forum Device in Listen Mode is exposed to an Unmodulated Carrier (see [ANALOG]), it MUST be ready to receive an ALLB\_REQ or SENSB\_REQ Command within a guard time **GT<sub>B</sub>**. Refer to Appendix A.2 for the value of **GT<sub>B</sub>**.
-



## 6 NFC-F Technology

This section specifies NFC-F Technology-related features of the NFC Forum Device. Contents, when already defined elsewhere, have been included for readability reasons.

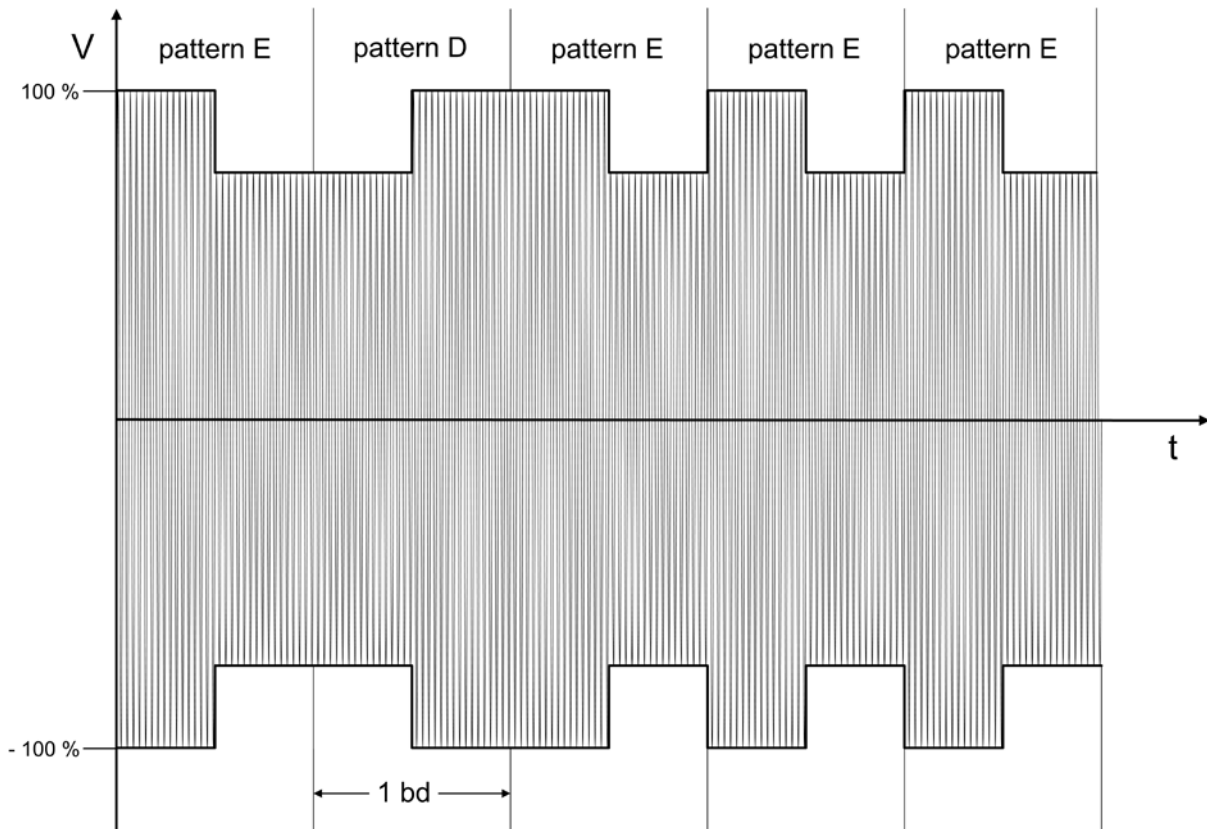
If not explicitly stated otherwise, [ISO/IEC\_18092] is normative reference for this section. Deviations from and restrictions of [ISO/IEC\_18092] are indicated within the text on the topics for which deviations and restrictions apply.

### 6.1 Sequence Format

This section describes the sequence format for NFC-F Technology.

#### 6.1.1 Modulation

In both transmission directions, the analog signal is modulated using Manchester coding with ASK modulation. For details about the Modulation Index refer to [ANALOG]. Figure 19 illustrates Manchester coding.



**Figure 19: Manchester Coding with ASK Modulation**

The modulation principle of ASK is used to modulate the carrier amplitude in a defined way between a  $V_3$  and a  $V_4$ . The carrier signal not modulated is denoted as  $V_3$ , the modulated carrier signal is denoted as  $V_4$ . Refer to [ANALOG] for the definition of  $V_3$  and  $V_4$ .

Manchester coding uses this principle to define two particular patterns: E and D.

---

### Requirements 71: Creation of Signal Patterns – NFC-F

---

#### Poll and Listen Mode

---

- 6.1.1.1 The NFC Forum Device MUST build signal patterns E and D:
- Pattern E: at the beginning of the bit duration, a  $V_3$  MUST occur for a time of half the bit duration (no carrier modulation). After a time of half the bit duration, a transition to  $V_4$  MUST occur and the  $V_4$  MUST last for a time of half the bit duration.
  - Pattern D: at the beginning of the bit duration, a  $V_4$  MUST occur for a time of half the bit duration. After a time of half the bit duration, a transition to  $V_3$  MUST occur and the  $V_3$  MUST last for a time of half the bit duration (no carrier modulation).
- 

### Requirements 72: Reading of Signal Patterns – NFC-F

---

#### Poll and Listen Mode

---

- 6.1.1.2 The NFC Forum Device MUST read signal patterns E and D as follows:
- If the NFC Forum Device detects a  $V_3$  at the beginning of the bit duration that lasts for half the bit duration, followed by a transition to  $V_4$  with the  $V_4$  lasting for half the bit duration, the NFC Forum Device MUST read this as pattern E.
  - If the NFC Forum Device detects a  $V_4$  at the beginning of the bit duration that lasts for half the bit duration, followed by a transition to  $V_3$  with the  $V_3$  lasting for half the bit duration, the NFC Forum Device MUST read this as pattern D.
  - All other patterns MUST be treated as invalid patterns.
- 

### Requirements 73: NFC-F Transition Boundaries

---

#### Poll and Listen Mode

---

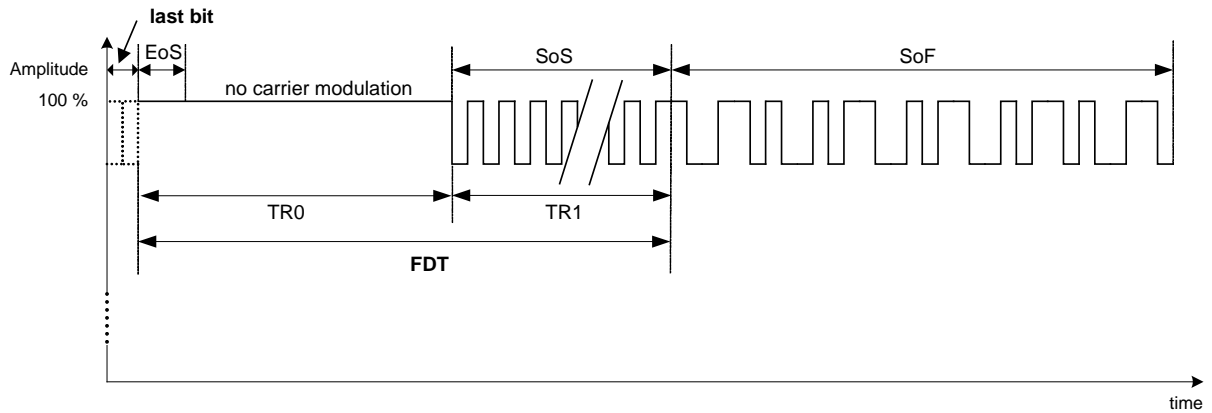
- 6.1.1.3 For both of Poll→Listen communication and Listen→Poll communication, the transition boundaries MUST occur between  $\pm 4/f_c$  in the Operating Field.
- 

NOTE Requirements in Section 6.1.1 overrule deviating definitions in [ISO/IEC\_18092].

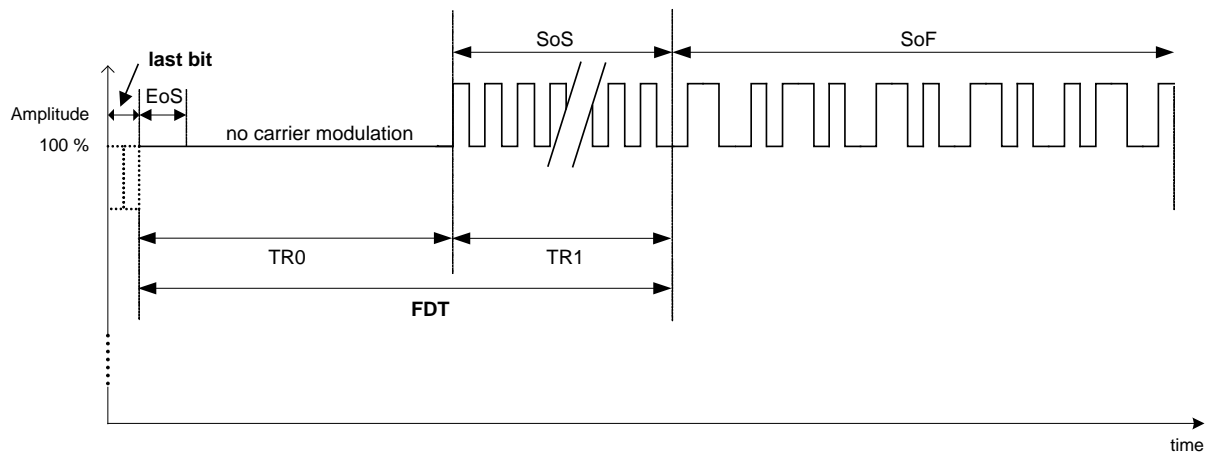
## 6.1.2 Synchronization

Figure 20 illustrates the different parameters used for NFC-F Technology signal synchronization and related signal timing parameters. Depending on the coupling between the NFC Forum Device in Poll Mode and the NFC Forum Device in Listen Mode, the waveform looks different. The coupling may inverse the waveform, called reverse polarity.

**a) Obverse Polarity**



**b) Reverse Polarity**



**Figure 20: Signal Synchronization and Timing Parameters**

For the demodulator, the beginning of a signal is indicated by the SoS. The duration of the SoS is TR1 and has a value of 48 bd.

**Requirements 74: Synchronization - NFC-F Sequence**

**Poll and Listen Mode**

6.1.2.1 The NFC Forum Device **MUST** code SoS as a series of 48 patterns D.

6.1.2.2 The NFC Forum Device **MUST** be synchronized no later than 48 patterns D or E.

### 6.1.3 De-synchronization

The end of a signal is indicated by no longer modulating the carrier.

---

#### Requirements 75: De-synchronization - NFC-F

---

##### Poll and Listen Mode

---

*The NFC Forum Device MAY indicate the EoS by not modulating the carrier for a time longer than one bit duration.*

---

- 6.1.3.1 For de-synchronization, the NFC Forum Device MUST use the SoD as specified in Section 6.4 or the EoS.
- If the SoD is used, the following applies:  
The NFC Forum Device MUST stop demodulating after receipt of the EoD, as indicated by the SoD containing the LEN byte (see Section 6.4).
  - If the EoS is used, the following applies:  
When directly following a pattern E or D, the NFC Forum Device MUST treat a carrier not modulated for a time longer than one bit duration as the EoS and stop demodulating.
- 

## 6.2 Bit Level Coding

The patterns E and D are used to code the digital alphabet Logic “0” and Logic “1”.

---

#### Requirements 76: Bit Level Coding – NFC-F

---

##### Poll and Listen Mode

---

- 6.2.1.1 The NFC Forum Device MUST code Logic “0” and Logic “1” as follows:
- Logic “1”: pattern E
  - Logic “0”: pattern D
- 

---

#### Requirements 77: Bit Level Decoding – NFC-F

---

##### Poll and Listen Mode

---

- 6.2.1.2 The NFC Forum Device MUST decode Logic “0” and Logic “1” as follows:
- If the NFC Forum Device detects pattern E, then it MUST decode this as Logic “1”.
  - If the NFC Forum Device detects pattern D, then it MUST decode this as Logic “0”.

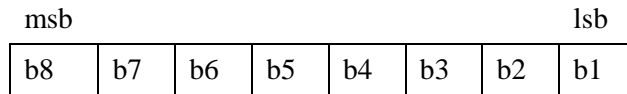
*Invalid patterns MAY be treated as Transmission Errors, or the device MAY use implementation dependent methods to decode an invalid pattern into a Logic “0” or Logic “1”.*

---

## 6.3 Frame Format

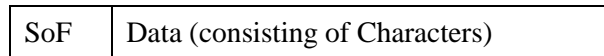
This section defines the frames and characters of the NFC Forum Device configured for NFC-F Technology.

A character consists of 8 data bits without start, stop, and parity bits, as shown in Figure 21. Characters are transmitted as a continuous string, with no separation in time between characters.



**Figure 21: NFC-F – Character Format**

A frame starts with the SoF followed by the data, as illustrated in Figure 22. Data consists of characters.



**Figure 22: NFC-F – Frame Format**

**Requirements 78: NFC-F – Frame Format**

<b>Poll and Listen Mode</b>	
6.3.1.1	A frame <b>MUST</b> begin with an SoF, followed by characters according to Figure 21.
6.3.1.2	The NFC Forum Device <b>MUST</b> code SoF as B24Dh.
6.3.1.3	After the synchronization, the NFC Forum Device <b>MUST</b> decode a series of Logic “0”s and Logic “1”s with a value B24Dh or 4DB2h as SoF. If the NFC Forum Device receives 4DB2h, then all subsequent bits <b>MUST</b> be inverted (i.e., a Logic “0” becomes Logic “1” and Logic “1” becomes Logic “0”). <i>The NFC Forum Device MAY decide the polarity to be used based on a subset of the SoF.</i>

## 6.4 Data and Payload Format

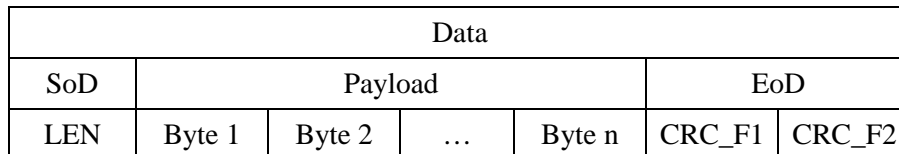
Data transmitted in an NFC-F frame has the following substructure. They consist of an SoD, the payload, and an EoD.

The SoD contains a length byte LEN, indicating the length of the payload + 1.

The payload consists of the Commands and Responses, as described in Section 6.5.

The EoD contains a two-byte checksum, referred to as CRC\_F. The CRC\_F is a function of k data bits. The number of bits k is a multiple of eight. Input for CRC\_F calculation is the SoD and the payload.

The NFC-F data and payload format is illustrated in Figure 23.



**Figure 23: Data and Payload Format – NFC-F**

## Requirements 79: Data and Payload Format – NFC-F

<b>Poll and Listen Mode</b>	
6.4.1.1	Data <b>MUST</b> be transmitted in frames using the frame format as defined in Section 6.3.
6.4.1.2	The SoD <b>MUST</b> contain a length byte LEN at the position shown in Figure 23 with a value equal to n+1, where n indicates the number of bytes that the payload consists of.
6.4.1.3	The SoD <b>MUST</b> have a value between 1 and 254. The NFC Forum Device <b>MUST</b> treat other values as a Transmission Error.
6.4.1.4	The payload <b>MUST</b> follow the SoD as indicated in Figure 23.
6.4.1.5	The payload <b>MUST</b> be followed by an EoD at the position, as indicated in Figure 23.
6.4.1.6	The EoD <b>MUST</b> contain a CRC_F.
6.4.1.7	The CRC_F <b>MUST</b> be calculated as defined in [ISO/IEC_18092]. CRC_F1 is the MSB and CRC_F2 is the LSB.
6.4.1.8	The NFC Forum Device <b>MUST</b> verify a CRC_F on receipt, as defined in [ISO/IEC_18092], and <b>MUST</b> treat failure of verification as a Transmission Error.

NOTE Requirement 6.4.1.3 overrules [ISO/IEC\_18092].

## 6.5 Command Set

Payload exchanged between NFC Forum Devices consists of Commands and Responses. Table 39 lists the Commands that are available to the NFC Forum Device configured for NFC-F Technology. For each Command, the corresponding Response is indicated.

**Table 39: NFC-F – Command Set**

<b>Poll Mode (Command)</b>	<b>Listen Mode (Response)</b>
SENSF_REQ	SENSF_RES

NOTE The Command and Response listed in Table 39 have different names in [ISO/IEC\_18092]: SENSF\_REQ refers to Polling Request and SENSF\_RES refers to Polling Response.

## 6.6 SENSF\_REQ

The SENSF\_REQ Command is used by an NFC Forum Device in Poll Mode to probe the Operating Field for NFC Forum Devices in Listen Mode.

### 6.6.1 SENSF\_REQ Command

Table 40 defines the format of the SENSF\_REQ Command.

**Table 40: SENSEF\_REQ Command Format**

Byte 1	Byte 2 - 3	Byte 4	Byte 5
00h	SC	RC	TSN

The components of this Command are defined as follows:

### SC

The System Code (SC) contains information regarding the NFC Forum Device to be polled for (e.g., the Technology Subset).

**Requirements 80: System Code (SC)**

Poll Mode		Listen Mode	
6.6.1.1	If configured to poll for any NFC Forum Device in Listen Mode, the NFC Forum Device MUST set SC to FFFFh.  <i>The NFC Forum Device MAY set SC to a different value if configured to poll for NFC Forum Devices in a specific configuration.</i>	6.6.1.2	The NFC Forum Device MUST be ready to receive a SENSEF_REQ Command with SC equal to FFFFh.  When configured for the Tag Type 3 platform, the NFC Forum Device MUST accept at least one different value for SC in accordance to the requirements given in [JIS_X_6319-4] for SC. The NFC Forum Device MUST not send the SENSEF_RES Response in all other cases.

### RC

The Request Code (RC) is used to retrieve additional information in the SENSEF\_RES Response and Table 41 specifies the RC code.

**Table 41: Coding of RC**

Value	Meaning
00h	No System Code information requested
01h	System Code information requested
02h	Advanced protocol features supported
Other values	RFU

**Requirements 81: Request Code (RC)**

<b>Poll Mode</b>		<b>Listen Mode</b>	
6.6.1.3	<p>If configured to poll for NFC Forum Devices that are configured for the NFC-DEP Protocol, the NFC Forum Device MUST set RC to 00h.</p> <p>If configured to poll for NFC Forum Devices that are configured for the Type 3 Tag Platform, the NFC Forum Device MUST set RC to 00h or 01h.</p>	6.6.1.4	<p>If configured for NFC-DEP Protocol, the NFC Forum Device MUST be ready to receive a SENSF_REQ Command with RC set to 00h or 02h.</p> <p>If configured for Type 3 Tag platform, the NFC Forum Device MUST be ready to receive a SENSF_REQ Command with RC set to 00h, 01h, or 02h.</p> <p>The NFC Forum Device MUST treat a SENSF_REQ Command with RC different from 00h, 01h, and 02h as RC equal to 00h.</p>

NOTE Advanced protocol features are not used by this version of the specification.

NOTE Support for Type 3 Tag platform Listen Mode is optional.

**TSN**

The Time Slot Number (TSN) is used for collision resolution and to reduce the probability of collisions. The anticollision scheme is based on the definition of time slots in which NFC Forum Devices in Listen Mode are invited to respond with minimum identification data.

The NFC Forum Device in Poll Mode sends a SENSF\_REQ Command with a TSN value indicating the number of time slots available. Each NFC Forum Device in Listen Mode presents within the range of the Operating Field, and then randomly selects a time slot in which it responds. The TSN byte set to 00h forces all NFC Forum Devices in Listen Mode to respond in the first time slot, and therefore, this TSN value is used if collision resolution is not used.

Figure 24 shows an example for the collision resolution mechanism. The NFC Forum Device in Poll Mode (Device 0) receives answers from four NFC Forum Devices in Listen Mode. Device 0 can distinguish Device 4 and Device 2 because they responded in different time slots. The Responses from Device 1 and Device 3 cannot be distinguished because a collision occurred in time slot 1. Time slots are numbered sequentially, starting from 0.



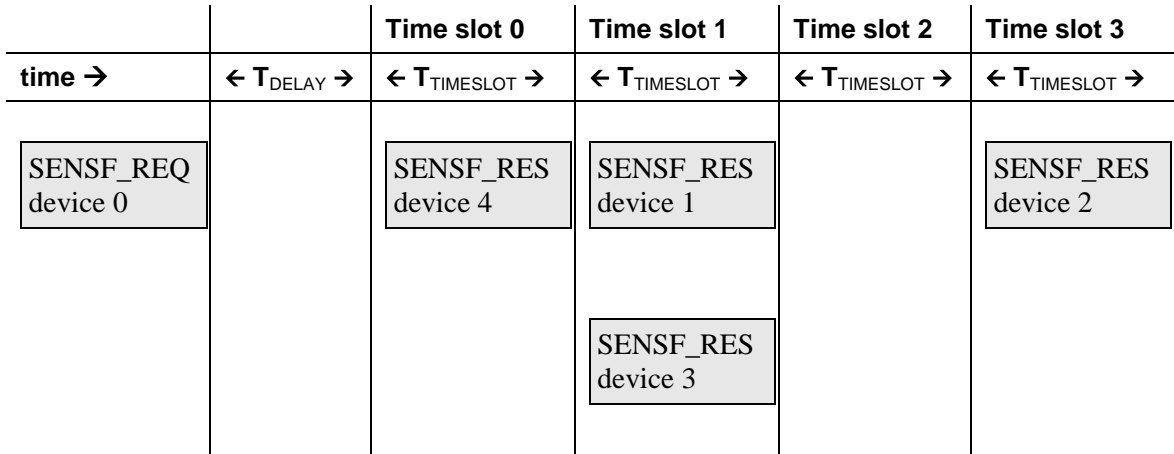


Figure 24: Collision Resolution (Example)

Table 42 specifies the coding of the TSN byte.

Table 42: Coding of TSN

TSN	00h	01h	03h	07h	0Fh	All other values
Number of Time Slots	1	2	4	8	16	RFU
Time Slots allowed for SENSF_RES Response	Time slot 0	Time slot 0 Time slot 1	Time slot 0 ... Time slot 3	Time slot 0 ... Time slot 7	Time slot 0 ... Time slot 15	

Requirements 82: Time Slot Number (TSN)

Poll Mode		Listen Mode	
6.6.1.5	The NFC Forum Device MUST set the TSN byte as specified in Table 42.	6.6.1.6	The NFC Forum Device MUST be ready to receive a SENSF_REQ Command with a TSN byte set to a value as specified in Table 42.
6.6.1.7	The NFC Forum Device MUST be ready to receive SENSF_RES Responses in each allowed time slot, according to Table 42.	6.6.1.8	The NFC Forum Device MUST randomly select a time slot allowed according to Table 42, and send its SENSF_RES Response in the selected time slot.

### 6.6.2 SENSF\_RES Response

Table 43 defines the SENSF\_RES format.

**Table 43: SENSEF\_RES Format**

Byte 1	Byte 2-9	Byte 10-11	Byte 12-14	Byte 15	Byte 16	Byte 17	[Byte 18-19]
01h	NFCID2	PAD0	PAD1	MRTI <sub>CHECK</sub>	MRTI <sub>UPDATE</sub>	PAD2	[RD]

**NFCID2**

NFCID2 is the NFC Forum Device identifier. Table 44 shows the NFCID2 format.

**Table 44: NFCID2 Format**

Byte 1	Byte 2	Description
01h	FEh	NFC-DEP Protocol supported. Bytes 3 to 8 are randomly generated.
02h	FEh	Tag Type 3 platform supported. Bytes 3 to 8 are randomly generated.
All other values		Tag Type 3 platform supported. Bytes 1 to 8 are proprietary.

**Requirements 83: NFCID2 in SENSEF\_RES****Listen Mode**

6.6.2.1 The NFCID2 MUST be coded as specified in Table 44.

**PAD0**

NFC Forum Devices do not use PAD0 for information exchange.

**Requirements 84: PAD0**

Poll Mode		Listen Mode	
6.6.2.2	The NFC Forum Device MUST be ready to receive a PAD0 field coded as any value.	6.6.2.3	The NFC Forum Device MUST set PAD0 to FFh FFh. <i>The NFC Forum Device MAY set PAD0 to a different value if configured for Type 3 Tag platform in a specific configuration.</i>

**PAD1**

The PAD1 format depends on the NFC-F Technology Subset for which the NFC Forum Device in Listen Mode is configured. NFC Forum Devices configured for the NFC-DEP Protocol do not use PAD1.

**Requirements 85: PAD1**

Poll Mode	Listen Mode
<p>6.6.2.4 If configured to poll for NFC Forum Devices that are only configured for the NFC-DEP Protocol, the NFC Forum Device MUST disregard any value returned in the PAD1 field.</p> <p>If configured to poll for NFC Forum Devices that are configured for the Type 3 Tag platform, the NFC Forum Device MUST be ready to receive a PAD1 field coded as any value.</p>	<p><i>The NFC Forum Device MAY set PAD1 to any value.</i></p>

NOTE Support for Type 3 Tag platform Listen Mode is optional.

**MRTI<sub>CHECK</sub>**

Coding of **MRTI<sub>CHECK</sub>** depends on the NFC-F Technology Subset for which the NFC Forum Device in Listen Mode is configured. NFC Forum Devices configured for the NFC-DEP Protocol do not use **MRTI<sub>CHECK</sub>**.

**Requirements 86: MRTI<sub>CHECK</sub>**

Poll Mode	Listen Mode
<p>6.6.2.5 If configured to poll for only NFC Forum Devices that are configured for the NFC-DEP Protocol, the NFC Forum Device MUST disregard any value returned in the <b>MRTI<sub>CHECK</sub></b> field.</p> <p>If configured to poll for NFC Forum Devices that are configured for the Type 3 Tag platform, the NFC Forum Device MUST be ready to receive an <b>MRTI<sub>CHECK</sub></b> value as specified in Section 10.6.</p>	<p>6.6.2.6 If configured for Type 3 Tag platform, the NFC Forum Device MUST set <b>MRTI<sub>CHECK</sub></b> as specified in Section 10.6.</p> <p><i>If configured for the NFC-DEP Protocol, the NFC Forum Device MAY set <b>MRTI<sub>CHECK</sub></b> to any value.</i></p>

**MRTI<sub>UPDATE</sub>**

The **MRTI<sub>UPDATE</sub>** format depends on the NFC-F Technology Subset for which the NFC Forum Device in Listen Mode is configured. NFC Forum Devices configured for the NFC-DEP Protocol do not use **MRTI<sub>UPDATE</sub>**.

**Requirements 87: MRTI<sub>UPDATE</sub>**

<b>Poll Mode</b>		<b>Listen Mode</b>	
6.6.2.7	<p>If configured to poll for the NFC-DEP Protocol, the NFC Forum Device MUST disregard any value returned in the <b>MRTI<sub>UPDATE</sub></b> field.</p> <p>If configured to poll for Type 3 Tag platform, the NFC Forum Device MUST be ready to receive an <b>MRTI<sub>UPDATE</sub></b> value as specified in Section 10.6.</p>	6.6.2.8	<p>If configured for the Type 3 Tag platform, the NFC Forum Device MUST set <b>MRTI<sub>UPDATE</sub></b> as specified in Section 10.6.</p> <p><i>If configured for the NFC-DEP Protocol, the NFC Forum Device MAY set <b>MRTI<sub>UPDATE</sub></b> to any value.</i></p>

**PAD2**

The PAD2 format depends on the NFC-F Technology Subset for which the NFC Forum Device in Listen Mode is configured. NFC Forum Devices configured for the NFC-DEP Protocol do not use PAD2.

**Requirements 88: PAD2**

<b>Poll Mode</b>		<b>Listen Mode</b>	
6.6.2.9	<p>If configured to poll for NFC-DEP Protocol, the NFC Forum Device MUST disregard any value returned in the PAD2 field.</p> <p>If configured to poll for Type 3 Tag platform, the NFC Forum Device MUST be ready to receive an PAD2 value as any value.</p>		<p><i>The NFC Forum Device MAY set PAD2 to any value.</i></p>

**RD**

Request Data (RD) is included in the SENSF\_RES Response if requested in the RC field of the SENSF\_REQ Command. The Request Data (RD) format depends on the NFC-F Technology Subset for which the NFC Forum Device in Listen Mode is configured.

**Requirements 89: Request Data (RD)**

<b>Poll Mode</b>		<b>Listen Mode</b>	
6.6.2.10	<p>If the preceding SENSF_REQ Command contained an RC byte set to 00h, the NFC Forum Device <b>MUST</b> treat the presence of the RD field in the SENSF_RES Response as Protocol Error.</p> <p>If configured to poll for Type 3 Tag platform and if the preceding SENSF_REQ Command contained an RC byte set to 01h, the NFC Forum Device <b>MUST</b> be ready to receive an RD value of 2 bytes length in the SENSF_RES Response.</p> <p>If the preceding SENSF_REQ Command contained an RC byte set to 02h, the NFC Forum Device <b>MUST</b> be ready to receive an RD value 2 bytes long in the SENSF_RES Response.</p>	6.6.2.11	<p>If the preceding SENSF_REQ Command contained an RC byte set to 00h, the NFC Forum Device <b>MUST</b> not send RD bytes within the SENSF_RES Response.</p> <p>If configured for the Type 3 Tag platform and if the preceding SENSF_REQ Command contained an RC byte set to 01h, the NFC Forum Device <b>MUST</b>:</p> <ul style="list-style-type: none"> <li>• Set the 2 RD bytes equal to the System Code.</li> <li>• Send out the RD bytes as part of the SENSF_RES Response MSB first.</li> </ul> <p>If the preceding SENSF_REQ Command contained an RC byte set to 02h, the NFC Forum Device <b>MAY</b> send RD bytes shown in Table 45 and Table 46 within the SENSF_RES Response.</p>
6.6.2.12	<p>If the preceding SENSF_REQ Command contained an RC byte set to 01h or 02h, the NFC Forum Device <b>MUST</b> accept a SENSF_RES without RD bytes as a valid response.</p>		

**Table 45: RD Format Advanced Protocol Features (Byte 18)**

<b>b8</b>	<b>b7</b>	<b>b6</b>	<b>b5</b>	<b>b4</b>	<b>b3</b>	<b>b2</b>	<b>b1</b>	<b>Meaning</b>
0	0	0	0	0	0	0	0	RFU

Table 46: RD Format Advanced Protocol Features (Byte 19)

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1								Automatically selectable bit rates
	0							RFU
		0						RFU
			0					RFU
				x				$D_{\text{POLL} \rightarrow \text{LISTEN}} = 16$ and $D_{\text{LISTEN} \rightarrow \text{POLL}} = 16$ supported, if bit is set to 1b.
					x			$D_{\text{POLL} \rightarrow \text{LISTEN}} = 8$ and $D_{\text{LISTEN} \rightarrow \text{POLL}} = 8$ supported, if bit is set to 1b.
						x		$D_{\text{POLL} \rightarrow \text{LISTEN}} = 4$ and $D_{\text{LISTEN} \rightarrow \text{POLL}} = 4$ supported, if bit is set to 1b.
							x	$D_{\text{POLL} \rightarrow \text{LISTEN}} = 2$ and $D_{\text{LISTEN} \rightarrow \text{POLL}} = 2$ supported, if bit is set to 1b.

## 6.7 Timing Requirements

This section specifies the requirements for the different Frame Delay Times for NFC-F Technology. The Frame Delay Time (FDT) is the time between two frames transmitted in opposite directions.

### 6.7.1 Frame Delay Time Poll→Listen

The Frame Delay Time Poll→Listen ( $FDT_{F,LISTEN}$ ) is the time between the end of the last bit of a Poll Frame and the start of the SoF of a Listen Frame as shown in Figure 20. The time between the minimum value  $FDT_{F,LISTEN,MIN}$  and the maximum value  $FDT_{F,LISTEN,MAX}$  defines the time interval in which a Listen Frame is allowed to be sent in response to a Poll Frame.

NOTE The term “Frame Response Time” as used in [ISO/IEC\_18092] is referred to as “Frame Delay Time” in this document.

#### $FDT_{F,LISTEN}$ (except SENSEF\_REQ)

$FDT_{F,LISTEN,MIN}$  is the minimum time an NFC Forum Device in Listen Mode has to wait before sending the Listen Frame after the end of a Poll Frame.

Except for the SENSEF\_REQ Command,  $FDT_{F,LISTEN,MIN}$  is defined:

$$FDT_{F,LISTEN,MIN} = TR0_{F,LISTEN,MIN} + TR1$$

where TR1 is the synchronization time (the bit duration of SoS: 48 bd) as illustrated in Figure 20. Refer to Appendix A.3 for the value of  $TR0_{F,LISTEN,MIN}$ .

**Requirements 90:  $FDT_{F,LISTEN,MIN}$  (except SENSF\_REQ)**

<b>Poll Mode</b>		<b>Listen Mode</b>	
6.7.1.1	Except for the SENSF_REQ Command, the NFC Forum Device MUST be ready to receive the SoF of a new Listen Frame no later than $FDT_{F,LISTEN,MIN}$ after the end of a Poll Frame.	6.7.1.2	Except for the SENSF_REQ Command following the end of a Poll Frame, the NFC Forum Device MUST wait at least $FDT_{F,LISTEN,MIN}$ before sending the SoF of its Listen Frame.
<p><i>The NFC Forum Device MAY ignore any response before <math>FDT_{F,LISTEN,MIN}</math>.</i></p>			

**$FDT_{F,LISTEN,SENSF\_REQ}$**

For the SENSF\_REQ Command,  $FDT_{F,LISTEN,SENSF\_REQ}$  depends on the time slot the NFC Forum Device in Listen Mode is responding in, and is defined:

$$FDT_{F,LISTEN,SENSF\_REQ} = TR0_{LISTEN} + TR1$$

where TR1 is the synchronization time (the bit duration of SoS: 48 bd) as illustrated in Figure 20.

$TR0_{LISTEN}$  is given in the following formula:

$$T_{DELAY} + n \times T_{TIMESLOT} \leq TR0_{LISTEN} \leq T_{DELAY} + (n + 1) \times T_{TIMESLOT} - 30 \times 8 \text{ bd}$$

where n is the time slot number randomly selected by the NFC Forum Device in Listen Mode ( $0 \leq n \leq TSN$ ), and  $30 \times 8 \text{ bd}$  is the bit duration of SENSF\_RES. Refer to Appendix A.3 for the values of  $T_{DELAY}$  and  $T_{TIMESLOT}$ .

Requirements 91:  $FDT_{F,LISTEN,SENSF\_REQ}$ 

Poll Mode	Listen Mode
<p>6.7.1.3 Following the end of a Poll Frame containing the SENSF_REQ Command, the NFC Forum Device MUST accept the SoF of the Listen Frame containing the SENSF_RES Response within the time slots defined by <math>FDT_{F,LISTEN,SENSF\_REQ}</math>.</p> <p>If the NFC Forum Device does not receive the SoF of the Listen Frame containing the SENSF_RES Response within <math>T_{DELAY} + (TSN+1) \times T_{TIMESLOT} - 30 \times 8 \text{ bd} + TR1 + \Delta T_{POLL}</math>, then the NFC Forum Device MUST treat this as a Timeout Error.</p> <p>Refer to Appendix A.3 for the value of <math>\Delta T_{POLL}</math>.</p> <p><i>Between <math>T_{DELAY} + (TSN+1) \times T_{TIMESLOT} - 30 \times 8 \text{ bd} + TR1</math> and <math>T_{DELAY} + (TSN+1) \times T_{TIMESLOT} - 30 \times 8 \text{ bd} + TR1 + \Delta T_{POLL}</math>, the NFC Forum Device MAY accept a SENSF_RES Response or MAY generate a Timeout Error.</i></p>	<p>6.7.1.4 Following the end of a SENSF_REQ Command, the NFC Forum Device MUST send the SoF of the Listen Frame containing the SENSF_RES Response within one of the time slots defined by <math>FDT_{F,LISTEN,SENSF\_REQ}</math>.</p>

### 6.7.2 Frame Delay Time Listen→Poll

The Frame Delay Time Listen→Poll ( $FDT_{F,POLL}$ ) is the time between the end of the last bit of a Listen Frame and the start of the SoF of a Poll Frame, as shown in Figure 20. In case the Listen frame is a SENSF\_RES, The Frame Delay Time Listen→Poll ( $FDT_{F,POLL}$ ) is the time between the end of the time slot specified in the SENSF\_REQ and the start of the SoF of a Poll Frame. The minimum value  $FDT_{F,POLL,MIN}$  defines the time an NFC Forum Device in Poll Mode has to wait before sending a new Poll Frame after receipt of a Listen Frame. A maximum value  $FDT_{F,POLL,MAX}$  is not defined.

$FDT_{F,POLL,MIN}$  is defined:

$$FDT_{F,POLL,MIN} = TR0_{F,POLL,MIN} + TR1$$

where TR1 is the synchronization time (the bit duration of SoS: 48 bd) as illustrated in Figure 20. Refer to Appendix A.3 for the value of  $TR0_{F,POLL,MIN}$ .



**Requirements 92:  $FDT_{F,POLL,MIN}$**

Poll Mode	Listen Mode
<p>6.7.2.1 Except for the SENSF_RES, following the end of a Listen Frame, the NFC Forum Device MUST wait at least <b><math>FDT_{F,POLL,MIN}</math></b> before transmitting the SoF of a new Poll Frame.</p> <p>For the SENSF_RES, following the end of the time slot specified in the SENSF_REQ, the NFC Forum Device MUST wait at least for a time <b><math>FDT_{F,POLL,MIN}</math></b> before transmitting the SoF of a new Poll Frame.</p>	<p>6.7.2.2 The NFC Forum Device MUST be ready to receive the SoF of a new Poll Frame no later than <b><math>FDT_{F,POLL,MIN}</math></b> after the Listen Frame.</p> <p><i>If the SoF of a Poll Frame is received before <b><math>FDT_{F,POLL,MIN}</math></b>, then the NFC Forum Device MAY treat this as a Transmission Error.</i></p>

A specific value for  **$FDT_{F,POLL,MIN}$**  is defined for the re-activation of the NFC Forum Device in Listen Mode following a DSL\_RES Response (see Section 14.9.2) or an RLS\_RES Response (see Section 14.10.2).

**Requirements 93:  $FDT_{F,REACTIVATION}$**

Poll Mode	Listen Mode
<p>6.7.2.3 Following the end of a DSL_RES Response or a RLS_RES Response, the NFC Forum Device MUST wait at least for <b><math>FDT_{F,REACTIVATION}</math></b> before transmitting the start of a new Poll Frame.</p> <p>Refer to Appendix A.3 for the value of <b><math>FDT_{F,REACTIVATION}</math></b>.</p>	<p>6.7.2.4 The NFC Forum Device MUST be ready to receive the start of a new Poll Frame no later than <b><math>FDT_{F,REACTIVATION}</math></b> after the end of a DSL_RES Response or a RLS_RES Response.</p> <p><i>If the start of a new Poll Frame is received before <b><math>FDT_{F,REACTIVATION}</math></b>, then the NFC Forum Device MAY treat this as a Transmission Error.</i></p>

NOTE Requirement 6.7.2.3 only applies for Initiator and not for Reader/Writer. In Reader/Writer mode, re-activation of a Type 3 Tag Platform is done by means of a Reset as defined in [ANALOG].

Requirement 6.7.2.4 applies for Target and not for Type 3 Tag (emulation).

### 6.7.3 Guard Time

This section specifies the guard time of an Unmodulated Carrier, after which the NFC Forum Device in Listen Mode must be ready to receive a SENSEF\_REQ Command.

#### Requirements 94: Guard Time – NFC-F

---

##### Listen Mode

---

- 6.7.3.1 When an NFC Forum Device in Listen Mode is exposed to an Unmodulated Carrier (see [ANALOG]), it MUST be ready to receive a SENSEF\_REQ Command within a guard time  $GT_F$ .  
Refer to Appendix A.3 for the value of  $GT_F$ .
-

## 7 Half-duplex Protocols

For the NFC Forum Devices, data is transmitted using half-duplex protocols. These protocols share common characteristics that are described in this section. Specific characteristics of each half-duplex protocol such as timeouts and exception handling are described in the following sections.

Half-duplex means that only one device sends data at a time. Specifically for NFC Forum Devices, there is a master-slave relationship between two devices. If the NFC Forum Device is in Poll Mode, it is the master; if the NFC Forum Device is in Listen Mode, it is the slave.

The NFC Forum Device in Poll Mode sends a single Poll Frame and then waits for a Listen Frame. The NFC Forum Device in Listen Mode waits for a Poll Frame before sending a single Listen Frame.

### Requirements 95: General Rules for Half-duplex Transmission Protocol

Poll Mode		Listen Mode	
7.1.1.1	After sending a Poll Frame, the NFC Forum Device in Poll Mode MUST switch to receive mode and wait for a Listen Frame before switching back to transmit mode.	7.1.1.2	After the activation of the NFC Forum Device in Listen Mode, the NFC Forum Device in Listen Mode MUST wait for a Poll Frame.
7.1.1.3	The NFC Forum Device in Poll Mode MUST not send another Poll Frame until it has received a Listen Frame or until a timeout occurred with no Listen Frame.	7.1.1.4	The NFC Forum Device in Listen Mode MUST send a Listen Frame when it has received a valid Poll Frame. After responding, the NFC Forum Device in Listen Mode MUST return to the receive mode.

## 8 Type 1 Tag Platform

### 8.1 Sequence Format

The Type 1 Tag platform uses the NFC-A synchronization mechanism.

#### Requirements 96: Sequence Format – Type 1 Tag Platform

Poll Mode	Listen Mode
8.1.1.1 Commands and Responses specified in Section 8 (Type 1 Tag platform) MUST be transmitted using NFC-A Technology sequences as defined in Section 4.1.	<i>Commands and Responses specified in Section 8 (Type 1 Tag platform) HAVE TO be transmitted using NFC-A Technology sequences as defined in Section 4.1.</i>

### 8.2 Bit Level Coding

The Type 1 Tag platform uses NFC-A bit level coding.

#### Requirements 97: Bit Level Coding – Type 1 Tag Platform

Poll Mode	Listen Mode
8.2.1.1 Commands and Responses specified in Section 8 (Type 1 Tag platform) MUST be transmitted using NFC-A Technology bit-level coding as defined in Section 4.2.	<i>Commands and Responses specified in Section 8 (Type 1 Tag platform) HAVE TO be transmitted using NFC-A Technology bit level coding as defined in Section 4.2.</i>

### 8.3 Frame Format

The Type 1 Tag platform uses NFC-A frame format for Listen Mode and a specific frame format for Poll Mode.

A Poll Frame consists of the following (see Figure 25):

- NFC-A SoF
- 8 data bits, transmitted lsb first
- EoF

No parity is added.

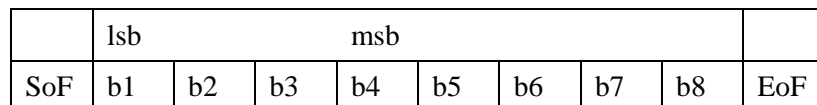


Figure 25: Frame Format – Type 1 Tag Platform in Poll Mode

**Requirements 98: Frame Format – Type 1 Tag Platform**

<b>Poll Mode</b>	<b>Listen Mode</b>
8.3.1.1 A Type 1 Tag Poll Frame MUST consist of eight data bits to be transmitted lsb first, MUST begin with an SoF, and MUST end with an EoF.  Refer to Section 4.3 for the definitions of SoF and EoF.	<i>A Listen Frame HAS TO have the NFC-A standard frame format as defined in Section 4.3.</i>

**8.4 Data and Payload Format**

Type 1 Tag data has the following substructure, and consists of the payload and the EoD.

The payload consists of the Commands and Responses as described in Section 8.5.

The Type 1 Tag data and payload format is illustrated in Figure 26.

Data					
Payload				EoD	
Byte 1	Byte 2	...	Byte n	CRC_B1	CRC_B2

**Figure 26: Data and Payload Format – Type 1 Tag**

The EoD contains a two-byte checksum referred to as CRC\_B. Input for CRC\_B calculation is the payload.

## Requirements 99: Data and Payload Format – Type 1 Tag

Poll Mode	Listen Mode
8.4.1.1 The payload MUST be followed by an EoD at the position, as indicated in Figure 26. The EoD MUST contain a CRC_B.	<i>The payload HAS TO be followed by an EoD at the position, as indicated in Figure 26. The EoD HAS TO contain a CRC_B.</i>
8.4.1.2 The CRC_B MUST be calculated as defined in [ISO/IEC_13239]. The initial register content MUST be all ones (FFFFh). CRC_B1 is the LSB and CRC_B2 is the MSB.	<i>The CRC_B HAS TO be calculated as defined in [ISO/IEC_13239]. The initial register content HAS TO be all ones (FFFFh). CRC_B1 is the LSB and CRC_B2 is the MSB.</i>
8.4.1.3 The NFC Forum Device MUST compare the received CRC_B with the CRC_B calculated according to Requirement 8.4.1.2. If different, then the NFC Forum Device MUST resort to exception processing with a Transmission Error.	<i>The NFC Forum Device HAS TO compare the received CRC_B with the CRC_B calculated according to Requirement 8.4.1.2. If different, then the NFC Forum Device HAS TO resort to exception processing with a Transmission Error.</i>

## 8.5 Command Set

Payload exchanged between NFC Forum Devices consists of Commands and Responses. Table 47 lists the Commands that are available to the NFC Forum Device in Poll Mode for communication with the (NFC Forum Device in Listen Mode configured for emulating the) Type 1 Tag platform. For each Command, the corresponding Response from the NFC Forum Device in Listen Mode is indicated.

Refer to [T1TOP] for Command coding (except for RID Command/Response) and NFC Type 1 Tag platform operation.

Command parameters and names are defined in [T1TOP].

**Table 47: Command Set**

<b>Poll Mode (Command)</b>	<b>Listen Mode (Response)</b>
RID Command	RID Response
RALL Command	RALL Response
READ Command	READ Response
WRITE-E Command	WRITE-E Response
WRITE-NE Command	WRITE-NE Response
RSEG Command	RSEG Response
READ8 Command	READ8 Response
WRITE-E8 Command	WRITE-E8 Response
WRITE-NE8 Command	WRITE-NE8 Response

## 8.6 Read Identifier (RID)

The RID Command is used to retrieve unique identifier and protocol support capabilities of an NFC Forum Device in Listen Mode.

### 8.6.1 RID Command

The format of the RID Command is shown in Table 48. There are no requirements for the ADD, DATA, and UID-echo fields other than that they have to be present.

**Table 48: RID Command Format**

<b>Byte 1</b>	<b>Byte 2</b>	<b>Byte 3</b>	<b>Byte 4 – 7</b>
78h	ADD	DATA	UID-echo

**Requirements 100: RID Command**

<b>Poll Mode</b>	<b>Listen Mode</b>
8.6.1.1 The NFC Forum Device <b>MUST</b> send each byte of the RID Command and the related CRC_B (i.e., the payload and the EoD) in a separate frame. The first frame <b>MUST</b> be a short frame as defined in Section 4.3, followed by Poll Frames according to Requirement 8.3.1.1.	<i>The NFC Forum Device <b>HAS TO</b> accept each byte of the RID Command and the related CRC_B (i.e., the payload and the EoD) in a separate frame, where the first frame is a short frame as defined in Section 4.3, followed by Poll Frames according to Requirement 8.3.1.1.</i>
<i>The NFC Forum Device does not use the ADD, DATA, and UID-echo fields, and <b>MAY</b> set these fields to Any Value.</i>	<i>The NFC Forum Device <b>HAS TO</b> disregard Any Value received in the ADD, DATA, and UID-echo fields of the RID Command.</i>

## 8.6.2 RID Response

The format of the RID Response is shown in Table 49.

**Table 49: RID Response Format**

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
HR0	HR1	UID0	UID1	UID2	UID3

The UID0, UID1, UID2, and UID3 bytes contain the unique identifier of the NFC Forum Device configured for the Type 1 Tag platform.

**Requirements 101: RID Response**

Poll Mode		Listen Mode
8.6.2.1	The NFC Forum Device MUST be ready to receive an RID Response with the most significant nibble of HR0 set to 0001b, indicating NDEF support.  The NFC Forum Device MUST treat an RID Response with the most significant nibble of HR0 set to a value different from 0001b as Protocol Error.	<i>The NFC Forum Device HAS TO indicate NDEF support by setting the most significant nibble of HR0 to 0001b.  Coding of HR1 and of the least significant nibble of HR0 is out of scope of this specification.</i>
8.6.2.2	The NFC Forum Device MUST be ready to receive the UID0, UID1, UID2, and UID3 bytes.	<i>The NFC Forum Device HAS TO return UID0, UID1, UID2, and UID3. The value of UID0, UID1, UID2, and UID3 is a fixed number or a random number.</i>

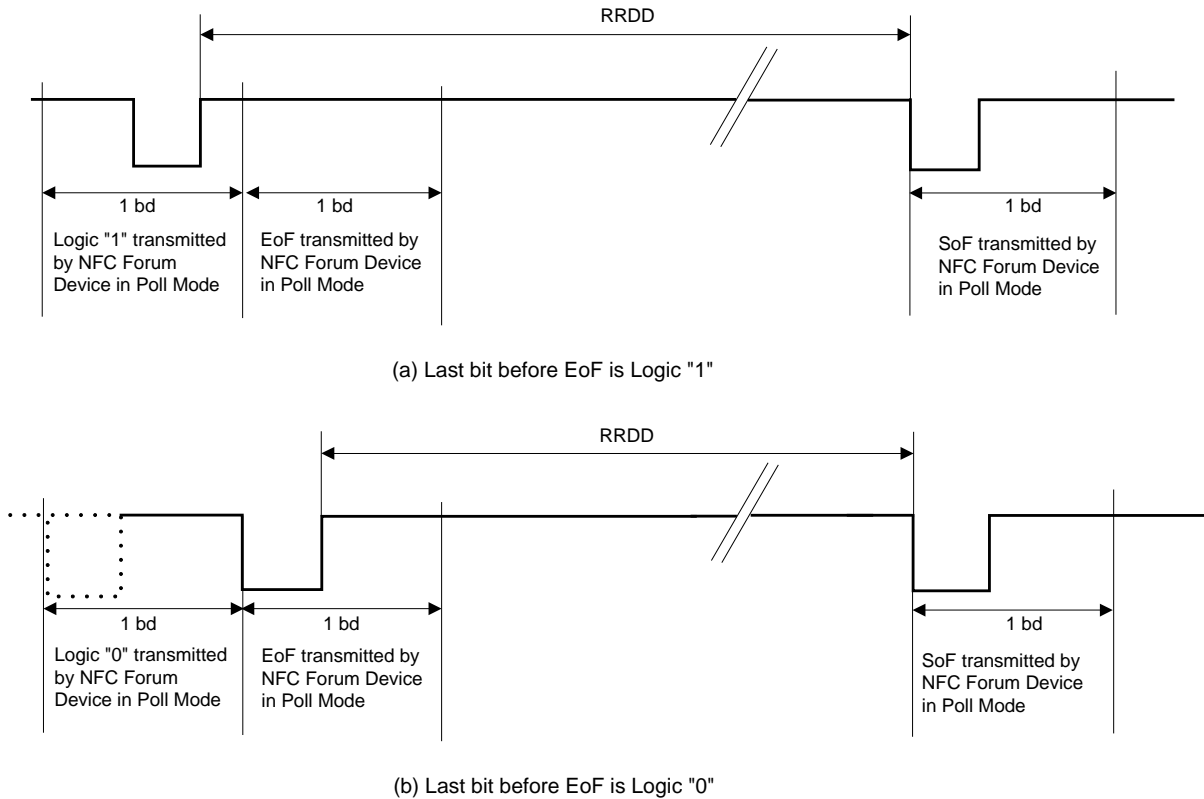
## 8.7 Timing Requirements

This section specifies the requirements for the different delay times used by the Type 1 Tag platform.

### 8.7.1 Reader-Reader Data Delay

The Reader-Reader Data Delay (RRDD) is the time between two successive Poll Frames and depends on the logic value of the last data bit of the preceding Poll Frame transmitted by the NFC Forum Device, as illustrated in Figure 27.





**Figure 27: Reader-Reader Data Delay (RRDD)**

**Requirements 102: Reader-Reader Data Delay (RRDD)**

**Poll Mode**

- 8.7.1.1 When transmitting a series of successive Poll Frames, the NFC Forum Device MUST separate the frames through the RRDD. The NFC Forum Device MUST wait for a time no less than
- $RRDD_{MIN,BIT0}$ , if the last transmitted data bit was Logic "0", or
  - $RRDD_{MIN,BIT1}$ , if the last transmitted data bit was Logic "1"
- before transmitting the next Poll Frame. RRDD MUST be measured from the rising edge of the last  $V_2$  in the preceding frame to the falling edge of the first  $V_2$  of the subsequent frame, as illustrated in Figure 27. Refer to Appendix A.4 for the values of  $RRDD_{MIN,BIT0}$  and  $RRDD_{MIN,BIT1}$ .

**8.7.2 Frame Delay Time**

NFC Forum Devices configured for the Type 1 Tag platform use NFC-A Frame Delay Times.

**Requirements 103: Frame Delay Times – Type 1 Tag Platform**

<b>Poll Mode</b>	<b>Listen Mode</b>
8.7.2.1    Commands and Responses specified in Section 8 (Type 1 Tag platform) <b>MUST</b> be transmitted according to NFC-A Technology Frame Delay Time restrictions, as defined in Section 4.10.1 and Section 4.10.2.	<i>Commands and Responses specified in Section 8 (Type 1 Tag platform) <b>HAS TO</b> be transmitted according to NFC-A Technology Frame Delay Time restrictions, as defined in Section 4.10.1 and Section 4.10.2.</i>

## 9 Type 2 Tag Platform

### 9.1 Sequence Format

The Type 2 Tag platform uses the NFC-A synchronization mechanism.

#### Requirements 104: Sequence Format – Type 2 Tag Platform

Poll Mode	Listen Mode
9.1.1.1	<p>Commands and Responses specified in this section (Type 2 Tag platform) MUST be transmitted using NFC-A Technology sequence format as defined in Section 4.1.</p>

### 9.2 Bit Level Coding

The Type 2 Tag platform uses NFC-A bit level coding.

#### Requirements 105: Bit Level Coding – Type 2 Tag Platform

Poll Mode	Listen Mode
9.2.1.1	<p>Commands and Responses specified in this section (Type 2 Tag platform) MUST be transmitted using NFC-A Technology bit level coding as defined in Section 4.2.</p>

### 9.3 Frame Format

The Type 2 Tag platform transmits Commands and Responses in NFC-A standard frames, except for the ACK and NACK Response.

A Listen Frame for the ACK and NACK Response consists of a short frame with 4 data bits (see Section 4.3.2).

#### Requirements 106: Frame Format – Type 2 Tag Platform

Poll Mode	Listen Mode
9.3.1.1	<p>Commands and Responses specified in this section (Type 2 Tag platform), except for the ACK and NACK Response, MUST be transmitted using NFC-A Technology standard frames as defined in Section 4.3.</p>

**Requirements 107: Frame Format – ACK and NACK Response**

<b>Poll Mode</b>	<b>Listen Mode</b>
9.3.1.2 A Listen Frame for the ACK and NACK Response <b>MUST</b> consist of four data bits to be transmitted lsb first, as shown in Figure 28, and <b>MUST</b> begin with an SoF, as defined in Section 4.3.	<i>A Listen Frame for the ACK and NACK Response <b>HAS TO</b> consist of four data bits to be transmitted lsb first, as shown in Figure 28, and <b>HAS TO</b> begin with an SoF, as defined in Section 4.3.</i>

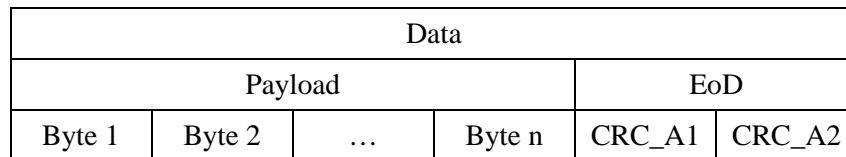
**9.4 Data and Payload Format**

Type 2 Tag data transmitted in an NFC-A standard frame, i.e., the bytes following the SoF, have the following substructure. They consist of the payload and, depending on the payload, of the EoD.

The payload consists of the Commands and Responses as described in Section 9.5.

If present, the EoD contains a 2-byte checksum referred to as CRC\_A. Input for CRC\_A calculation is the payload. If the payload consists of the ACK or NACK Response, then EoD is not present.

Type 2 Tag data and payload format is illustrated in Figure 28.



**Figure 28: Data and Payload Format – Type 2 Tag (except for ACK and NACK Response)**

**Requirements 108: Data and Payload Format – Type 2 Tag**

<b>Poll Mode</b>	<b>Listen Mode</b>
9.4.1.1 If the payload consists of a Command or Response different from the ACK or NACK Response, then the payload <b>MUST</b> be followed by an EoD at the position, as indicated in Figure 28, and the EoD <b>MUST</b> contain a CRC_A. Otherwise, the EoD <b>MUST</b> not be present.	<i>If the payload consists of a Command or Response different from the ACK or NACK Response, then the payload <b>HAS TO</b> be followed by an EoD at the position, as indicated in Figure 28, and the EoD <b>HAS TO</b> contain a CRC_A. Otherwise, the EoD does not <b>HAVE TO</b> be present.</i>
9.4.1.2 The CRC_A <b>MUST</b> be calculated and verified as specified in Section 4.4.	<i>The CRC_A <b>HAS TO</b> be calculated and verified as specified in Section 4.4.</i>

## 9.5 Command Set

Payload exchanged between NFC Forum Devices consists of Commands and Responses. Table 50 lists the Commands that are available to the NFC Forum Device in Poll Mode for communication with an NFC Forum Device in Listen Mode configured for the Type 2 Tag platform. For each Command, the corresponding Response from the NFC Forum Device in Listen Mode is indicated.

The READ Command and Response are specified in Section 9.6. Refer to [T2TOP] for the coding of the remaining Commands and for the Type 2 Tag platform operation. Command parameters and names are defined in [T2TOP].

**Table 50: Command Set**

Poll Mode (Command)	Listen Mode (Response)
READ Command	READ Response, NACK Response
WRITE Command	ACK Response, NACK Response
SECTOR SELECT Command	ACK Response, NACK Response, Passive ACK Response

## 9.6 READ

[T2TOP] is the normative reference for this section.

### 9.6.1 Command

The format of the READ Command is shown in Table 51. Refer to [T2TOP] for further details regarding syntax and semantic of the READ Command.

**Table 51: READ Command Format**

Byte 1	Byte 2
30h	BNo

### 9.6.2 Response

In response to the READ Command, the NFC Forum Device returns a READ Response or a NACK response.

**Requirements 109: READ Response**

Poll Mode		Listen Mode	
9.6.2.1	The NFC Forum Device <b>MUST</b> be ready to receive a READ Response with a payload composed of 16 bytes or a NACK response.	9.6.2.2	If the NFC Forum Device receives a READ Command, then it <b>HAS TO</b> return a READ Response with a payload composed of 16 bytes or a NACK response.

The format of the READ Response is shown in Table 52.

**Table 52: READ Response Format**

<b>Byte 1 - 16</b>
payload

Refer to [T2TOP] for further details regarding the payload coding.

The format of the NACK Response is shown in Table 53.

**Table 53: NACK Response Format**

<b>b4</b>	<b>b3</b>	<b>b2</b>	<b>b1</b>	<b>Meaning</b>
0	x	0	x	The NACK Response has a value of 0h, 1h, 4h, or 5h.

**Requirements 110: NACK Response**

<b>Poll Mode</b>	
9.6.2.3	When a NACK Response is received by the NFC Forum Device, this <b>MUST</b> be treated as Protocol Error.

## 9.7 WRITE

[T2TOP] is the normative reference for this section.

### 9.7.1 Command

The format of the WRITE Command is shown in Table 54. Refer to [T2TOP] for further details regarding syntax and semantic of the WRITE Command.

**Table 54: WRITE Command Format**

<b>Byte 1</b>	<b>Byte 2</b>	<b>Byte 3 – 6</b>
A2h	BNo	Data

### 9.7.2 Response

In response to the WRITE Command, the NFC Forum Device returns an ACK or NACK response.

**Requirements 111: WRITE Response**

<b>Poll Mode</b>		<b>Listen Mode</b>	
9.7.2.1	When an ACK Response is received, the WRITE Command <b>MUST</b> be treated as being successful. A NACK Response <b>MUST</b> be treated as Protocol Error.	9.7.2.2	If the NFC Forum Device receives a WRITE Command, it <b>HAS TO</b> return an ACK Response or a NACK Response.

The format of the ACK Response is shown in Table 55. The format of the NACK Response is shown in Table 53.

**Table 55: ACK Response Format**

<b>b4</b>	<b>b3</b>	<b>b2</b>	<b>b1</b>	<b>Meaning</b>
1	0	1	0	The ACK Response has a value of Ah.

## 9.8 SECTOR SELECT

[T2TOP] is the normative reference for this section.

The SECTOR SELECT Command is composed of two command packets, called SECTOR SELECT Command Packet 1 and SECTOR SELECT Command Packet 2.

### 9.8.1 SECTOR SELECT Command Packet 1

The format of the SECTOR SELECT Command Packet 1 is shown in Table 56. Refer to [T2TOP] for further details regarding syntax and semantic of the SECTOR SELECT Command Packet 1.

**Table 56: SECTOR SELECT Command Packet 1 Format**

<b>Byte 1</b>	<b>Byte 2</b>
C2h	“FF”

### 9.8.2 SECTOR SELECT Command Packet 2

The format of the SECTOR SELECT Command Packet 2 is shown in Table 57. Refer to [T2TOP] for further details regarding syntax and semantic of the SECTOR SELECT Command Packet 2.

**Table 57: SECTOR SELECT Command Packet 2 Format**

<b>Byte 1</b>	<b>Byte 2 - 4</b>
SNo	RFU

### 9.8.3 Response

In response to the SECTOR SELECT Command Packet 1, the NFC Forum Device returns an ACK or NACK response. The format of the ACK Response is shown in Table 55. The format of the NACK Response is shown in Table 53. If the Poll Frame contains an SECTOR SELECT Command Packet 2, then  $FDT_{A,LISTEN,MAX}$  is equal to  $PAT_{T2T,SL,MAX}$ .

### Requirements 112: SECTOR SELECT Command Packet 1 Response

Poll Mode	Listen Mode
9.8.3.1	9.8.3.2
<p>When a SECTOR SELECT Command Packet 1 has been sent by the NFC Forum Device and an ACK Response is received, the SECTOR SELECT Command Packet 1 MUST be treated as being successful.</p> <p>A NACK Response MUST be treated as Protocol Error.</p>	<p>If the NFC Forum Device receives a SECTOR SELECT Command Packet 1, then it HAS TO return an ACK or NACK Response.</p>

In response to the SECTOR SELECT Command Packet 2, the NFC Forum Device returns a Passive ACK Response.

### Requirements 113: Passive ACK Response

Poll Mode	Listen Mode
9.8.3.3	9.8.3.4
<p>When a SECTOR SELECT Command Packet 2 has been sent by the NFC Forum Device, and no response is received within <math>PAT_{T2T,SL,MAX}</math> after the SECTOR SELECT Command Packet 2, then the SECTOR SELECT Command Packet 2 MUST be treated as being successful (i.e., <math>FDT_{A,LISTEN,MAX}</math> equals <math>PAT_{T2T,SL,MAX}</math>).</p> <p>Refer to Appendix A.5 for the value of <math>PAT_{T2T,SL,MAX}</math>.</p> <p>When a SECTOR SELECT Command Packet 2 has been sent by the NFC Forum Device, and a response is received within <math>PAT_{T2T,SL,MAX}</math> after the SECTOR SELECT Command Packet 2, then the SECTOR SELECT Command Packet 2 MUST be treated as a Protocol Error.</p>	<p>If the NFC Forum Device receives a SECTOR SELECT Command Packet 2, it is free to return a NACK Response or no Response within <math>PAT_{T2T,SL,MAX}</math> of the reception of the SECTOR SELECT Command Packet 2 (i.e., <math>FDT_{A,LISTEN,MAX}</math> equals <math>PAT_{T2T,SL,MAX}</math>). It does not HAVE TO respond after <math>PAT_{T2T,SL,MAX}</math>.</p>



## 9.9 Timing Requirements

The Type 2 Tag platform uses NFC-A Frame Delay Times.

### Requirements 114: Frame Delay Times – Type 2 Tag Platform

Poll Mode	Listen Mode
<p>9.9.1.1 Commands and Responses specified in this section (Type 2 Tag platform) <b>MUST</b> be transmitted according to NFC-A Technology Frame Delay Time restrictions, as defined in Section 4.10.1 and Section 4.10.2.</p> <p>Regarding the maximum Frame Delay Time Poll→Listen, Requirement 9.9.1.3 <b>MUST</b> be applied.</p>	<p>9.9.1.2 Commands and Responses specified in this section (Type 2 Tag platform) <b>HAVE TO</b> be transmitted according to NFC-A Technology Frame Delay Time restrictions, as defined in Section 4.10.1 and Section 4.10.2.</p> <p>Regarding the maximum Frame Delay Time Poll→Listen, Requirement 9.9.1.3 <b>HAS TO</b> be applied.</p>
<p>9.9.1.3 The NFC Forum Device <b>MUST</b> wait for a Response at least for time:</p> <ul style="list-style-type: none"> <li>• <b>FDT</b><sub>T2T,READ,MAX</sub> for the READ Command</li> <li>• <b>FDT</b><sub>T2T,WRITE,MAX</sub> for the WRITE Command</li> <li>• <b>FDT</b><sub>T2T,SL,MAX</sub> for the SECTOR SELECT Command Packet 1</li> </ul> <p>If the NFC Forum Device does not receive a Response within <b>FDT</b><sub>T2T,READ/WRITE/SL,MAX</sub> + <math>\Delta T_{POLL}</math>, then the NFC Forum Device <b>MUST</b> treat this as a Timeout Error (i.e., <b>FDT</b><sub>A,LISTEN,MAX</sub> equals <b>FDT</b><sub>T2T,READ/WRITE/SL,MAX</sub> + <math>\Delta T_{POLL}</math>).</p> <p>Refer to Appendix A.5 for the values of <b>FDT</b><sub>T2T,READ,MAX</sub>, <b>FDT</b><sub>T2T,WRITE,MAX</sub>, <b>FDT</b><sub>T2T,SL,MAX</sub>, and <math>\Delta T_{POLL}</math>.</p> <p><i>Between <b>FDT</b><sub>T2T,READ/WRITE/SL,MAX</sub> and <b>FDT</b><sub>T2T,READ/WRITE/SL,MAX</sub> + <math>\Delta T_{POLL}</math>, the NFC Forum Device <b>MAY</b> accept the Response or <b>MAY</b> generate a Timeout Error.</i></p> <p><i>Note: See Section 9.8.3.3 for the definition of the passive ACK response and timing for the SECTOR SELECT Command Packet 2.</i></p>	<p><i>The NFC Forum Device <b>HAS TO</b> respond no later than:</i></p> <ul style="list-style-type: none"> <li>• <b>FDT</b><sub>T2T,READ,MAX</sub> for the READ Command</li> <li>• <b>FDT</b><sub>T2T,WRITE,MAX</sub> for the WRITE Command</li> <li>• <b>FDT</b><sub>T2T,SL,MAX</sub> for each packet of the SECTOR SELECT Command</li> </ul> <p><i>(i.e., <b>FDT</b><sub>A,LISTEN,MAX</sub> equals <b>FDT</b><sub>T2T,READ/WRITE/SL,MAX</sub>)</i></p>

## 10 Type 3 Tag Platform

Implementation of Type 3 Tag platform Listen Mode is optional. However, if implemented, then it has to be implemented as specified here (see Requirements 1.10.1.1).

### 10.1 Sequence Format

The Type 3 Tag platform uses the NFC-F synchronization mechanism.

#### Requirements 115: Sequence Format – Type 3 Tag Platform

---

##### Poll and Listen Mode

---

- 10.1.1.1 Commands and Responses specified in Section 10 (Type 3 Tag platform) MUST be transmitted using NFC-F Technology sequence format as defined in Section 6.1.
- 

### 10.2 Bit Level Coding

The Type 3 Tag platform uses NFC-F bit level coding.

#### Requirements 116: Bit Level Coding – Type 3 Tag Platform

---

##### Poll and Listen Mode

---

- 10.2.1.1 Commands and Responses specified in Section 10 (Type 3 Tag platform) MUST be transmitted using NFC-F Technology bit level coding as defined in Section 6.1.
- 

### 10.3 Frame Format

The Type 3 Tag platform transmits Commands and Responses in NFC-F frames.

#### Requirements 117: Frame Format – Type 3 Tag Platform

---

##### Poll and Listen Mode

---

- 10.3.1.1 Commands and Responses specified in Section 10 (Type 3 Tag platform) MUST be transmitted using NFC-F Technology frames as defined in Section 6.2.
- 

### 10.4 Data and Payload Format

Type 3 Tag data follow the data and payload format as specified in Section 6.4 for NFC-F Technology.

#### Requirements 118: Data and Payload Format – Type 3 Tag

---

##### Poll and Listen Mode

---

- 10.4.1.1 Commands and Responses specified in Section 10 (Type 3 Tag platform) MUST be transmitted using NFC-F Technology data and payload format as defined in Requirements 6.4.1.2 to 6.4.1.5.
-

## 10.5 Command Set

Payloads exchanged between NFC Forum Devices consist of Commands and Responses. Table 58 lists the Commands that are available to the NFC Forum Device in Poll Mode for communication with an NFC Forum Device in Listen Mode configured for the Type 3 Tag platform, after device activation. For each Command, the corresponding Response from the NFC Forum Device in Listen Mode is indicated. The CHECK Command is used to retrieve information regarding the NDEF capabilities of the responding NFC Forum Device.

Refer to [T3TOP] for the format of the POLLING, CHECK, and UPDATE Commands and Responses.

**Table 58: Command Set**

Poll Mode (Command)	Listen Mode (Response)
POLLING Command	POLLING Response
CHECK Command	CHECK Response
UPDATE Command	UPDATE Response

## 10.6 Timing Requirements

The Type 3 Tag platform uses NFC-F Frame Delay Times. Contents, when already defined elsewhere, have been included in this section for readability reasons. Timing requirements on SENSF\_REQ and SENSF\_RES are described in Section 6.7.

The Type 3 Tag platform uses the Maximum Response Time (MRT) for the timing requirements.

The MRTI for the CHECK Command (**MRTI<sub>CHECK</sub>**) and for the UPDATE Command (**MRTI<sub>UPDATE</sub>**) defines the MRT within which the (NFC Forum Device in Listen Mode configured for emulating the) Type 3 Tag platform has to send the SoD of its Response after the EoD of a Poll Frame containing the CHECK Command or UPDATE Command, respectively. MRT is calculated by the following formula, applicable for **MRT<sub>CHECK</sub>** and **MRT<sub>UPDATE</sub>**:

$$\text{MRT} = \mathbf{T} \times ((\mathbf{A}+1) + n (\mathbf{B}+1)) \times 4^{\mathbf{E}}$$

where:

- The parameter  $n$  denotes the size of the Block field (i.e., the number of blocks) in the CHECK Command or in the UPDATE Command, as defined in [T3TOP].
- The parameters  $A$ ,  $B$ , and  $E$  for the CHECK Command are transmitted to the NFC Forum Device in Poll Mode in the **MRTI<sub>CHECK</sub>** field of the SENSF\_RES Response, as defined in Table 59 and Section 6.6.2.
- The parameters  $A$ ,  $B$ , and  $E$  for the UPDATE Command are transmitted to the NFC Forum Device in Poll Mode in the **MRTI<sub>UPDATE</sub>** field of the SENSF\_RES Response, as defined in Table 59 and Section 6.6.2.

The value of  $\mathbf{T}$  is given in Appendix A.6.

The **FDT<sub>F,LISTEN,MAX</sub>** defined in Section 6.7 equals  $\text{MRT} - 16 \text{ bd}$  (where 16 bd is the bit duration of the SoF).

Table 59: Format of  $MRT_{CHECK}$  and  $MRT_{UPDATE}$ 

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
					x	x	x	Parameter A
		x	x	x				Parameter B
x	x							Parameter E

Requirements 119:  $MRT_{CHECK}$ 

Poll Mode	Listen Mode
<p>10.6.1.1 Following the CHECK Command, the NFC Forum Device <b>MUST</b> wait at least <math>MRT_{CHECK} + \Delta RWT</math> for the SoD of the Response.</p> <p>If the NFC Forum Device does not receive the SoD of the Response within <math>MRT_{CHECK} + \Delta RWT + \Delta T_{POLL}</math>, then the NFC Forum Device <b>MUST</b> treat this as a Timeout Error.</p> <p>Refer to Appendix A.6 for the values of <math>\Delta RWT</math> and <math>\Delta T_{POLL}</math>.</p> <p><i>Between <math>MRT_{CHECK} + \Delta RWT</math> and <math>MRT_{CHECK} + \Delta RWT + \Delta T_{POLL}</math>, the NFC Forum Device <b>MAY</b> accept the Response or <b>MAY</b> generate a Timeout Error.</i></p>	<p>10.6.1.2 Following the EoD of the CHECK Command, the NFC Forum Device <b>MUST</b> wait no longer than <math>MRT_{CHECK}</math> before sending the SoD of the Response.</p>

Requirements 120:  $MRT_{UPDATE}$ 

Poll Mode	Listen Mode
<p>10.6.1.3 Following the UPDATE Command, the NFC Forum Device <b>MUST</b> wait at least <math>MRT_{UPDATE} + \Delta RWT</math> for the SoD of the Response.</p> <p>If the NFC Forum Device does not receive the SoD of the Response within <math>MRT_{UPDATE} + \Delta RWT + \Delta T_{POLL}</math>, then the NFC Forum Device <b>MUST</b> treat this as a Timeout Error.</p> <p>Refer to Appendix A.6 for the values of <math>\Delta RWT</math> and <math>\Delta T_{POLL}</math>.</p> <p><i>Between <math>MRT_{UPDATE} + \Delta RWT</math> and <math>MRT_{UPDATE} + \Delta RWT + \Delta T_{POLL}</math>, the NFC Forum Device <b>MAY</b> accept the Response or <b>MAY</b> generate a Timeout Error.</i></p>	<p>10.6.1.4 Following the EoD of the UPDATE Command, the NFC Forum Device <b>MUST</b> wait no longer than <math>MRT_{UPDATE}</math> before sending the SoD of the Response.</p>

## 11 Type 4A Tag Platform

Implementation of Type 4A Tag platform Listen Mode is optional. However, if implemented, then it has to be implemented as specified here (see Requirements 1.10.1.1).

### 11.1 Sequence Format

The Type 4A Tag platform uses the NFC-A synchronization mechanism.

#### Requirements 121: Sequence Format – Type 4A Tag Platform

---

##### Poll and Listen Mode

---

- 11.1.1.1 Commands and Responses specified in Section 11 (Type 4A Tag platform) MUST be transmitted using NFC-A Technology sequence format as defined in Section 4.1.
- 

### 11.2 Bit Level Coding

The Type 4A Tag platform uses NFC-A bit level coding.

#### Requirements 122: Bit Level Coding – Type 4A Tag Platform

---

##### Poll and Listen Mode

---

- 11.2.1.1 Commands and Responses specified in Section 11 (Type 4A Tag platform) MUST be transmitted using NFC-A Technology bit level coding as defined in Section 4.2.
- 

### 11.3 Frame Format

The Type 4A Tag platform transmits Commands and Responses in NFC-A standard frames.

#### Requirements 123: Frame Format – Type 4A Tag Platform

---

##### Poll and Listen Mode

---

- 11.3.1.1 Commands and Responses specified in this section (Type 4A Tag platform) MUST be transmitted using NFC-A Technology standard frames as defined in Section 4.3.
- 

### 11.4 Data and Payload Format

During the Device Activation Activity, Type 4A Tag data has the following substructure, which consists of the payload and of an EoD.

The payload consists of the Commands and Responses as described in Section 11.5.

The EoD contains a two-byte checksum referred to as CRC\_A. Input for CRC\_A calculation is the payload.

Figure 29 illustrates the Type 4A Tag data and payload format.

Data					
Payload				EoD	
Byte 1	Byte 2	...	Byte n	CRC_A1	CRC_A2

**Figure 29: Data and Payload Format – Type 4A Tag**

**Requirements 124: Data and Payload Format – Type 4A Tag**

---

**Poll and Listen Mode**

---

11.4.1.1 The payload MUST be followed by an EoD at the position, as indicated in Figure 29. The EoD MUST contain a CRC\_A.

11.4.1.2 The CRC\_A MUST be calculated and verified as specified in Section 4.4.

---

## 11.5 Command Set

Payloads exchanged between NFC Forum Devices consist of Commands and Responses. Table 60 lists the Commands that are available to the NFC Forum Device in Poll Mode for communication with the (NFC Forum Device in Listen Mode configured for emulating the) Type 4A Tag platform during the Device Activation Activity. For each Command, the corresponding Response from the NFC Forum Device in Listen Mode is indicated.

**Table 60: Command Set**

Poll Mode (Command)	Listen Mode (Response)
RATS	ATS

Refer to [T4TOP] for the definition of the high level command set.

## 11.6 Request for Answer to Select (RATS)

The RATS Command is used by the NFC Forum Device in Poll Mode during the Device Activation Activity to negotiate the maximum frame size and the bit rate divisors (D) with the NFC Forum Device in Listen Mode.

### 11.6.1 RATS Command

Table 61 specifies the code for the RATS Command.

**Table 61: Format of RATS Command**

Byte 1	Byte 2
E0h	PARAM

PARAM, the parameter byte, consists of two parts as shown in Table 62.

**Table 62: Format of RATS Parameter Byte (PARAM)**

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
x	x	x	x					FSDI
				x	x	x	x	DID

The most significant nibble b8 to b5 is called FSDI (Frame Size for proximity coupling Device Integer) and codes FSD (Frame Size for proximity coupling Device). The FSD defines the maximum size of a frame that the NFC Forum Device in Poll Mode is able to receive. FSD is expressed in number of data bytes included in the frame.

**Requirements 125: Frame Size for NFC Forum Device in Poll Mode (FSD)**

Poll Mode		Listen Mode	
11.6.1.1	The NFC Forum Device MUST accept frames with a number of data bytes less than or equal to FSD. The NFC Forum Device MUST resort to exception processing (Protocol Error) if it receives a frame with more than FSD data bytes.	11.6.1.2	The NFC Forum Device MUST send frames with a number of data bytes less than or equal to FSD.
11.6.1.3	The FSD supported by the NFC Forum Device MUST be <b>FSD<sub>MIN</sub></b> . Refer to Appendix A.7 for the value of <b>FSD<sub>MIN</sub></b> .	11.6.1.4	The NFC Forum Device MUST support an FSD equal to <b>FSD<sub>MIN</sub></b> . <i>The NFC Forum Device MAY support an FSD less than <b>FSD<sub>MIN</sub></b>.</i>

Table 63 provides the coding of FSD in terms of FSDI.

**Table 63: FSDI to FSD Conversion**

FSDI	0h	1h	2h	3h	4h	5h	6h	7h	8h	9h-Eh	Fh
FSD (bytes)	16	24	32	40	48	64	96	128	256	RFU	256

**Requirements 126: FSDI<sub>MIN</sub>**

Poll Mode	
11.6.1.5	The NFC Forum Device MUST set FSDI equal to 8h or Fh. FSDI set to Fh indicates support of advanced protocol parameters.

### Requirements 127: Listen Mode Handling of RFU Values of FSDI

---

#### Listen Mode

---

11.6.1.6 A received value of FSDI = 9h–Eh MUST be treated by the NFC Forum Device as FSDI = Fh.

---

NOTE Requirements 11.6.1.5 and 11.6.1.6 and Table 63 overrule [ISO/IEC\_14443].

The least significant nibble b4 to b1 is named DID, and it defines the logical number of the addressed NFC Forum Device in Listen Mode in the range from 0 to 14 (DID = 15 is not allowed).

NOTE The term CID as used in [ISO/IEC\_14443] is referred to as DID in this document.

### Requirements 128: Support of DID

---

#### Poll Mode

---



---

#### Listen Mode

---

11.6.1.7 The NFC Forum Device MUST set the DID field to a value between 0 and 14.

11.6.1.8 The NFC Forum Device MUST support a DID field with a value between 0 and 14.

---

### Requirements 129: Listen Mode Handling of RFU Value of DID

---

#### Listen Mode

---

11.6.1.9 The NFC Forum Device MUST treat a DID=15 as Protocol Error.

---

## 11.6.2 RATS Response (Answer To Select)

The Answer To Select (ATS) is transmitted by the NFC Forum Device in Listen Mode in response to the RATS Command. This section defines the ATS with all its available fields, as shown in Table 64.

**Table 64: Structure of the ATS**

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6 – 6+k-1
TL	T0	TA(1)	TB(1)	TC(1)	T1 ... Tk

---

The length byte TL is followed by a variable number of bytes in the following order:

- Format byte T0,
- Interface bytes TA(1), TB(1), TC(1), and
- Historical bytes T1 to Tk.



The parameter  $k$  denotes the number of historical bytes.

- Length byte

The length byte TL is mandatory and specifies the length of the transmitted ATS, including the TL byte itself.

**Requirements 130: Length Byte TL of the ATS**

Poll Mode		Listen Mode	
		11.6.2.1	The first byte of the ATS (TL) MUST specify the length of the ATS including TL itself.
11.6.2.2	The NFC Forum Device MUST be ready to receive an ATS with TL specifying a length less than or equal to 20 bytes.	11.6.2.3	TL MUST not indicate a length greater than 20 bytes.
<i>The NFC Forum Device MAY accept an ATS with TL indicating a length greater than 20 bytes.</i>			

- Format Byte T0

The format byte T0 is coded as specified in Table 65.

**Table 65: Coding of Format Byte T0**

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0								RFU
	x							TC(1) is transmitted, if bit is set to 1
		x						TB(1) is transmitted, if bit is set to 1
			x					TA(1) is transmitted, if bit is set to 1
				x	x	x	x	FSCI

The least significant nibble b4 to b1 is called FSCI (Frame Size for proximity Card Integer) and codes the maximum frame size (FSC). The FSC defines the maximum size of a frame accepted by the NFC Forum Device in Listen Mode. FSC is expressed in number of data bytes included in the frame.

**Requirements 131: FSC**

<b>Poll Mode</b>		<b>Listen Mode</b>	
11.6.2.4	The NFC Forum Device MUST send frames with a number of data bytes less than or equal to FSC.	11.6.2.5	The NFC Forum Device MUST accept frames with a number of data bytes less than or equal to FSC.  The NFC Forum Device MUST resort to exception processing (Protocol Error) if it receives a frame with more than FSC data bytes.
11.6.2.6	The NFC Forum Device MUST be capable of sending frames in accordance with an FSC greater than or equal to <b>FSC<sub>MIN</sub></b> .	11.6.2.7	The FSC supported by the NFC Forum Device MUST be at least <b>FSC<sub>MIN</sub></b> . Refer to Appendix A.7 for the value of <b>FSC<sub>MIN</sub></b> .

Table 66 specifies the coding of FSC in terms of FSCI. The default value of FSCI is 2 and leads to a FSC of 32 bytes.

**Table 66: FSCI to FSC Conversion**

FSCI	0h	1h	2h	3h	4h	5h	6h	7h	8h	9h-Fh
FSC (bytes)	16	24	32	40	48	64	96	128	256	RFU

**Requirements 132: FSCI**

<b>Listen Mode</b>	
11.6.2.8	The NFC Forum Device MUST set FSCI greater than or equal to the value corresponding to <b>FSC<sub>MIN</sub></b> with a maximum value of 8h. Refer to Appendix A.7 for the value of <b>FSC<sub>MIN</sub></b> . Refer to Table 66 for FSCI to FSC conversion.

**Requirements 133: Poll Mode Handling of RFU Values of FSCI**

<b>Poll Mode</b>	
11.6.2.9	A received value of FSCI = 9h–Fh MUST be treated by the NFC Forum Device as FSCI = 8h.

## Requirements 134: Format Byte T0 of the ATS

Poll Mode	Listen Mode
11.6.2.10 The NFC Forum Device MUST be ready to receive an ATS including T0, TA(1), TB(1), and TC(1). If one or more of the fields T0, TA(1), TB(1), and TC(1) are missing, then the NFC Forum Device MUST use the default values as specified in this section.	11.6.2.11 TA(1), TB(1) and TC(1) MUST be present in the ATS and the presence MUST be indicated in T0.

- Interface Byte TA(1)

The interface byte TA(1) conveys information to define the bit rate capabilities of the NFC Forum Device in Listen Mode. The interface byte TA(1) is coded as specified in Table 67. The bits b7 to b5 code the bit rate capability of the NFC Forum Device in Listen Mode for the direction from NFC Forum Device in Listen Mode to NFC Forum Device in Poll Mode ( $D_{\text{LISTEN} \rightarrow \text{POLL}}$ ). The default value for the bits b7 to b5 is 000b ( $D_{\text{LISTEN} \rightarrow \text{POLL}} = 1$ ). The bits b3 to b1 code the bit rate capability of the NFC Forum Device in Listen Mode for the direction from NFC Forum Device in Poll Mode to NFC Forum Device in Listen Mode ( $D_{\text{POLL} \rightarrow \text{LISTEN}}$ ). The default value for the bits b3 to b1 is 000b ( $D_{\text{POLL} \rightarrow \text{LISTEN}} = 1$ ).

Table 67: Coding of Interface Byte TA(1)

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
x								If b8 = 1b, then only the same bit rate divisor for both directions is supported ( $D_{\text{LISTEN} \rightarrow \text{POLL}} = D_{\text{POLL} \rightarrow \text{LISTEN}}$ ). If b8 = 0b, then a different bit rate divisor for each direction is supported.
	x							$D_{\text{LISTEN} \rightarrow \text{POLL}} = 8$ supported, if bit is set to 1b.
		x						$D_{\text{LISTEN} \rightarrow \text{POLL}} = 4$ supported, if bit is set to 1b.
			x					$D_{\text{LISTEN} \rightarrow \text{POLL}} = 2$ supported, if bit is set to 1b.
				0				RFU
					x			$D_{\text{POLL} \rightarrow \text{LISTEN}} = 8$ supported, if bit is set to 1b.
						x		$D_{\text{POLL} \rightarrow \text{LISTEN}} = 4$ supported, if bit is set to 1b.
							x	$D_{\text{POLL} \rightarrow \text{LISTEN}} = 2$ supported, if bit is set to 1b.

**Requirements 135: Format Byte TA(1) of the ATS**

<b>Poll Mode</b>	<b>Listen Mode</b>
11.6.2.12 The NFC Forum Device <b>MUST</b> support a bit rate of 106 kbits/s in both directions.  <i>The NFC Forum Device MAY support higher bit rates.</i>	11.6.2.13 In response to a RATS Command with FSDI of PARAM (byte 2) set to 8h, the NFC Forum Device <b>MUST</b> set the bits b7 to b5 and b3 to b1 of TA(1) equal to 0b, indicating that it supports only a bit rate of 106 kbits/s in both directions.  <i>The NFC Forum Device MAY indicate support for higher bit rates in response to a RATS Command with FSDI of PARAM (byte 2) set to a value of Fh.</i>

**Requirements 136: Poll Mode Handling of RFU Value of b4 of Interface Byte TA(1)**

<b>Listen Mode</b>
11.6.2.14 A received RFU value of b4 = 1b <b>MUST</b> be interpreted by the NFC Forum Device in Poll Mode as if b7 to b1 = 0000000b (only 106 kbits/s in both directions).

NOTE Higher bit rates will be specified in next versions of this specification.

- Interface Byte TB(1)

The interface byte TB(1) conveys information to define the Frame Waiting Time (FWT) and the Start-up Frame Guard Time (SFGT). The interface byte TB(1) is coded as specified in Table 68.

**Table 68: Coding of Interface Byte TB(1)**

<b>b8</b>	<b>b7</b>	<b>b6</b>	<b>b5</b>	<b>b4</b>	<b>b3</b>	<b>b2</b>	<b>b1</b>	<b>Meaning</b>
x	x	x	x					FWI
				x	x	x	x	SFGI

- The most significant nibble b8 to b5 is called FWI (Frame Waiting time Integer) and codes the Frame Waiting Time (FWT) for the subsequent Commands. Refer to Section 11.7 for the definition of the FWT. FWI has a value in the range from 1 to 14. The value 15 is RFU. The default value of FWI is 4.

**Requirements 137: Interface Byte TB(1) of the ATS**

<b>Poll Mode</b>		<b>Listen Mode</b>	
11.6.2.15	The NFC Forum Device in Poll Mode MUST support an ATS with TB(1) indicating an FWI less than or equal to <b>FWI<sub>MAX</sub></b> .	11.6.2.16	The NFC Forum Device MUST set FWI less than or equal to <b>FWI<sub>MAX</sub></b> . Refer to Appendix A.7 for the value of <b>FWI<sub>MAX</sub></b> .

**Requirements 138: Poll Mode Handling of RFU Value of FWI**

<b>Poll Mode</b>	
11.6.2.17	A received value of FWI = 15 MUST be treated by the NFC Forum Device as FWI = 4.

- The least significant nibble b4 to b1 codes SFGI (Start-up Frame Guard time Integer) and is used by the NFC Forum Device in Listen Mode to code a multiplier value used to define the SFGT. Refer to Section 11.7 for the definition of SFGT. SFGI has a value in the range from 1 to 14. The value 15 is RFU. The default value of SFGI is 0.

**Requirements 139: Interface Byte TB(1) of the ATS**

<b>Poll Mode</b>		<b>Listen Mode</b>	
11.6.2.18	The NFC Forum Device MUST be ready to receive an ATS with TB(1) indicating an SFGI less than or equal to <b>SFGI<sub>MAX</sub></b> .	11.6.2.19	The NFC Forum Device MUST set SFGI less than or equal to <b>SFGI<sub>MAX</sub></b> . Refer to Appendix A.7 for the value of <b>SFGI<sub>MAX</sub></b> .

**Requirements 140: Poll Mode Handling of RFU value of SFGI**

<b>Poll Mode</b>	
11.6.2.20	A received value of SFGI = 15 MUST be treated by the NFC Forum Device as SFGI = 0.

- Interface Byte TC(1)

The interface byte TC(1) indicates whether advanced protocol features NAD and DID are supported by the NFC Forum Device in Listen Mode. The interface byte TC(1) is coded as specified in Table 69.

**Table 69: Coding of Interface Byte TC(1)**

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	0	0						RFU
			x					0b: Advanced protocol features not supported 1b: Advanced protocol features supported
				0	0			RFU
						x		DID supported, if bit is set to 1b
							x	NAD supported, if bit is set to 1b

The bits b2 and b1 are used by the NFC Forum Device in Listen Mode to indicate which optional fields in the SoD it supports. Bit b1 set to 1b indicates NAD supported; b2 set to 1b indicates DID supported. Refer to Section 13.1 for the specification of the SoD.

Bit b5 is used to indicate whether the NFC Forum Device in Listen Mode supports advanced protocol features.

**Requirements 141: NAD**

Poll Mode		Listen Mode	
11.6.2.21	The NFC Forum Device <b>MUST</b> not use NAD and <b>MUST</b> disregard any value returned by the NFC Forum Device in b1.	11.6.2.22	The NFC Forum Device <b>MUST</b> not support NAD and set b1 equal to 0b.

- Historical Bytes

The historical bytes T1 to Tk are optional and are used by the NFC Forum Device in Listen Mode to designate general information. The maximum length of the ATS gives the maximum possible number of historical bytes.

**Requirements 142: Historical Bytes of the ATS**

Poll Mode		Listen Mode	
11.6.2.23	The NFC Forum Device <b>MUST</b> be ready to receive an ATS with up to 15 historical bytes.	11.6.2.24	The NFC Forum Device <b>MUST</b> send no more than 15 historical bytes.
<i>The NFC Forum Device MAY accept an ATS with more than 15 historical bytes.</i>			

## 11.7 Timing Requirements

### 11.7.1 FWT

The Frame Waiting Time (FWT) defines the time within which an NFC Forum Device in Listen Mode configured for the Type 4A Tag platform has to start its Response after the end of a Poll Frame and is calculated by the following formula:

$$FWT = (256 \times 16/f_c) \times 2^{FWI}$$

where the value of FWI has the range from 0 to 14 and is provided by the RATS Response as specified in Section 11.6.2.

The RATS Command includes a definition for a specific FWT. For this Command, the NFC Forum Device in Listen Mode starts sending its Response frame within **FWT<sub>ACTIVATION</sub>** (activation frame waiting time). Refer to Section 11.6.1 for the definition of the RATS Command.

**Requirements 143: Frame Timing – Type 4A Tag Platform**

Poll Mode	Listen Mode
<p>11.7.1.1 Following the RATS Command, the NFC Forum Device MUST wait at least <b>FWT<sub>ACTIVATION</sub></b> for the ATS. If the NFC Forum Device does not receive the ATS within <b>FWT<sub>ACTIVATION</sub> + ΔT<sub>POLL</sub></b>, then the NFC Forum Device MUST treat this as a Timeout Error (i.e., <b>FDT<sub>A,LISTEN,MAX</sub></b> equals <b>FWT<sub>ACTIVATION</sub> + ΔT<sub>POLL</sub></b>). Refer to Appendix A.7 for the value of <b>ΔT<sub>POLL</sub></b>.</p> <p><i>Between <b>FWT<sub>ACTIVATION</sub></b> and <b>FWT<sub>ACTIVATION</sub> + ΔT<sub>POLL</sub></b>, the NFC Forum Device MAY accept the ATS or MAY generate a Timeout Error.</i></p>	<p>11.7.1.2 The NFC Forum Device MUST start the ATS Response after the end of the RATS Command within <b>FWT<sub>ACTIVATION</sub></b> (i.e., <b>FDT<sub>A,LISTEN,MAX</sub></b> equals <b>FWT<sub>ACTIVATION</sub></b>). Refer to Appendix A.7 for the value of <b>FWT<sub>ACTIVATION</sub></b>.</p>
<p>11.7.1.3 Except for the RATS Command, the NFC Forum Device MUST wait at least <b>FWT+ΔFWT</b> for a Response. Except for the RATS Command, if the NFC Forum Device does not receive a Response within <b>FWT+ΔFWT + ΔT<sub>POLL</sub></b>, then the NFC Forum Device MUST treat this as a Timeout Error (i.e., <b>FDT<sub>A,LISTEN,MAX</sub></b> equals <b>FWT+ΔFWT + ΔT<sub>POLL</sub></b>). Refer to Appendix A.7 for the values of <b>ΔFWT</b> and <b>ΔT<sub>POLL</sub></b>.</p> <p><i>Between <b>FWT + ΔFWT</b> and <b>FWT + ΔFWT + ΔT<sub>POLL</sub></b>, the NFC Forum Device MAY accept the Response or MAY generate a Timeout Error.</i></p>	<p>11.7.1.4 Except for the RATS Command, the NFC Forum Device MUST start its Response after the end of a Command within the FWT (i.e., <b>FDT<sub>A,LISTEN,MAX</sub></b> equals FWT).</p>

### 11.7.2 SFGT

The SFGT is the minimum time the NFC Forum Device in Poll Mode is waiting before the NFC Forum Device in Listen Mode is ready to receive the next frame after it has sent the RATS Response. Refer to Section 11.6.2 for the definition of the RATS Response.

The SFGT is calculated:

$$\text{SFGT} = (256 \times 16/f_c) \times 2^{\text{SFGI}}$$

where SFGI has the range from 1 to 14 and is returned by the NFC Forum Device in Listen Mode in the interface byte TB(1) of the RATS Response. If the NFC Forum Device in Listen Mode returns SFGI equal to zero or SFGI is not returned, then no SFGT is needed and **FDT<sub>A,POLL,MIN</sub>** applies.

#### Requirements 144: SFGT – Type 4A Tag Platform

Poll Mode	Listen Mode
<p>11.7.2.1 If the NFC Forum Device in Listen Mode returns an SFGI different from zero, the NFC Forum Device in Poll Mode <b>MUST</b> wait at least <b>SFGT+ΔSFGT</b> before sending the next frame after the NFC Forum Device in Listen Mode has sent the ATS. Refer to Appendix A.7 for the value of <b>ΔSFGT</b>.</p>	<p>11.7.2.2 If the NFC Forum Device in Listen Mode returns an SFGI different from zero, then the NFC Forum Device <b>MUST</b> be ready to receive the start of a new Poll Frame no later than SFGT after the end of the ATS frame.</p> <p><i>If the start of a new Poll Frame is received before SFGT, then the NFC Forum Device MAY treat this as a Transmission Error.</i></p>
<p>11.7.2.3 If the NFC Forum Device in Listen Mode returns an SFGI equal to zero or SFGI is not returned, the NFC Forum Device in Poll Mode <b>MUST</b> wait at least <b>FDT<sub>A,POLL,MIN</sub></b> before sending the next frame after the NFC Forum Device in Listen Mode has sent the ATS.</p>	<p>11.7.2.4 If the NFC Forum Device in Listen Mode returns an SFGI equal to zero or does not return an SFGI, then the NFC Forum Device <b>MUST</b> be ready to receive the start of a new Poll Frame no later than <b>FDT<sub>A,POLL,MIN</sub></b> after the end of the ATS frame.</p> <p><i>If the start of a new Poll Frame is received before <b>FDT<sub>A,POLL,MIN</sub></b>, then the NFC Forum Device MAY treat this as a Transmission Error.</i></p>



## 12 Type 4B Tag Platform

Implementation of Type 4B Tag platform Listen Mode is optional. However, if implemented, then it has to be implemented as specified here (see Requirements 1.10.1.1).

### 12.1 Sequence Format

The Type 4B Tag platform uses the NFC-B synchronization mechanism.

#### Requirements 145: Sequence Format – Type 4B Tag Platform

---

##### Poll and Listen Mode

---

- 12.1.1.1 Commands and Responses specified in Section 12 (Type 4B Tag platform) MUST be transmitted using NFC-B Technology sequence format as defined in Section 5.1.
- 

### 12.2 Bit Level Coding

The Type 4B Tag platform uses NFC-B bit level coding.

#### Requirements 146: Bit Level Coding – Type 4B Tag Platform

---

##### Poll and Listen Mode

---

- 12.2.1.1 Commands and Responses specified in Section 12 (Type 4B Tag platform) MUST be transmitted using NFC-B Technology bit level coding as defined in Section 5.2.
- 

### 12.3 Frame Format

The Type 4B Tag platform transmits Commands and Responses in NFC-B frames.

#### Requirements 147: Frame Format – Type 4B Tag Platform

---

##### Poll and Listen Mode

---

- 12.3.1.1 Commands and Responses specified in Section 12 (Type 4B Tag platform) MUST be transmitted using NFC-B Technology frames as defined in Section 5.3.
- 

### 12.4 Data and Payload Format

The payload consists of the Commands and Responses as described in Section 12.5.

#### Requirements 148: Data and Payload Format – Type 4B Tag Platform

---

##### Poll and Listen Mode

---

- 12.4.1.1 Commands and Responses specified in Section 12 (Type 4B Tag platform) MUST be transmitted using the NFC-B Technology data and payload format as defined in Section 5.4.
-

## 12.5 Command Set

Payload exchanged between NFC Forum Devices consists of Commands and Responses. Table 70 lists the Commands that are available to the NFC Forum Device in Poll Mode for communication with an NFC Forum Device in Listen Mode configured for the Type 4B Tag platform during the Device Activation Activity. For each Command, the corresponding Response from the NFC Forum Device in Listen Mode is indicated.

**Table 70: Command Set**

Poll Mode (Command)	Listen Mode (Response)
ATTRIB Command	ATTRIB Response

Refer to [T4TOP] for the definition of the high level command set.

## 12.6 ATTRIB

The ATTRIB Command is sent by the NFC Forum Device in Poll Mode during the Device Activation Activity in order to negotiate a set of communication parameters with the NFC Forum Device in Listen Mode.

### 12.6.1 ATTRIB Command

Table 71 defines the format of the ATTRIB Command.

**Table 71: ATTRIB Command Format**

Byte 1	Byte 2 – 5	Byte 6	Byte 7	Byte 8	Byte 9	Byte 10 – 10+k-1
1Dh	NFCID0	Param 1	Param 2	Param 3	Param 4	Higher layer - INF

### NFCID0

Byte 2 up to byte 5 include the NFCID0 sent by the NFC Forum Device in Listen Mode in the SENSB\_RES.

**Requirements 149: NFCID0 in ATTRIB Command**

Poll Mode	Listen Mode
12.6.1.1 The NFC Forum Device in Poll Mode MUST send the ATTRIB Command using the NFCID0 received in the Valid SENSB_RES Response from the NFC Forum Device in Listen Mode.	12.6.1.2 The NFC Forum Device MUST recognize its own NFCID0 and respond only to a Valid ATTRIB Command in which its NFCID0 is included.

**Param 1**

The NFC Forum Device in Poll Mode codes Param 1 with the values of minimum TR0 and TR1, and whether SoS and EoS should be used. The format of Param 1 is specified in Table 72.

**Table 72: Format of Param 1 of the ATTRIB Command**

<b>b8</b>	<b>b7</b>	<b>b6</b>	<b>b5</b>	<b>b4</b>	<b>b3</b>	<b>b2</b>	<b>b1</b>	<b>Meaning</b>
x	x							Minimum TR0 00b: Default value 01b: $48 \times 16/f_C$ 10b: $16 \times 16/f_C$ 11b: RFU
		x	x					Minimum TR1 00b: Default value 01b: $64 \times 16/f_C$ 10b: $16 \times 16/f_C$ 11b: RFU
				x				Suppression of EoS 0b: EoS required 1b: EoS not required
					x			Suppression of SoS 0b: SoS required 1b: SoS not required
						0	0	RFU

## Requirements 150: Format of Param 1 of the ATTRIB Command

Poll Mode	Listen Mode
<p>12.6.1.3 If bit b4 of the ADC in the SENSB_RES of the NFC Forum Device in Listen Mode is set to 0b, then the NFC Forum Device in Poll Mode MUST set b8 and b7 equal to 00b, indicating that the default minimum value of TR0 (<b>TR0<sub>MIN</sub></b>) is to be used.</p> <p>Refer to Appendix A.2 for the value of <b>TR0<sub>MIN</sub></b>.</p> <p><i>If bit b4 of the ADC in the SENSB_RES of the NFC Forum Device in Listen Mode is set to 1b, then the NFC Forum Device in Poll Mode MAY set b8 and b7 to values different from 00b, as indicated in Table 72.</i></p>	<p><i>The NFC Forum Device MAY implement values different from 00b for Minimum TR0, as indicated in Table 72.</i></p>
<p>12.6.1.4 If bit b4 of the ADC in the SENSB_RES of the NFC Forum Device in Listen Mode is set to 0b, then the NFC Forum Device in Poll Mode MUST set b6 and b5 equal to 00b, indicating that the default minimum value of TR1 (<b>TR1<sub>MIN</sub></b>) is to be used.</p> <p>Refer to Appendix A.2 for the value of <b>TR1<sub>MIN</sub></b>.</p> <p><i>If bit b4 of the ADC in the SENSB_RES of the NFC Forum Device in Listen Mode is set to 1b, then the NFC Forum Device in Poll Mode MAY set b6 and b7 to values different from 00b, as indicated in Table 72.</i></p>	<p><i>The NFC Forum Device MAY implement values different from 00b for Minimum TR1, as indicated in Table 72.</i></p>
<p>12.6.1.5 If bit b4 of the ADC in the SENSB_RES of the NFC Forum Device in Listen Mode is set to 0b, then the NFC Forum Device in Poll Mode MUST set b4 and b3 equal to 00b, indicating that it does not support suppression of SoS/EoS.</p> <p><i>If bit b4 of the ADC in the SENSB_RES of the NFC Forum Device in Listen Mode is set to 1b, then the NFC Forum Device in Poll Mode MAY set b4 and b3 to values different from 00b, as indicated in Table 72.</i></p>	<p>12.6.1.6 The NFC Forum Device in Listen Mode MUST not implement suppression of SoS/EoS for bit rates higher than <math>f_c/128</math>.</p> <p><i>The NFC Forum Device MAY implement suppression of SoS/EoS for communication at <math>f_c/128</math>.</i></p>

**Requirements 151: Listen Mode Handling of RFU Values of Minimum TR0 and TR1**

<b>Listen Mode</b>	
12.6.1.7	A received RFU value of 11b for Minimum TR0 and Minimum TR1 MUST be interpreted as 00b, the default value.

**Param 2**

The NFC Forum Device in Poll Mode codes Param 2 as specified in Table 73.

**Table 73: Format of Param 2 of the ATTRIB Command**

<b>b8</b>	<b>b7</b>	<b>b6</b>	<b>b5</b>	<b>b4</b>	<b>b3</b>	<b>b2</b>	<b>b1</b>	<b>Meaning</b>
x	x							Bit rate Listen→Poll
		x	x					Bit rate Poll→Listen
				x	x	x	x	FSDI

The least significant nibble (b4 to b1) of Param 2 is used by the NFC Forum Device in Poll Mode to code the maximum frame size (FSD) that it can receive. The FSD defines the maximum size of a frame the NFC Forum Device in Poll Mode is able to receive. FSD is expressed in number of data bytes included in the frame.

**Requirements 152: Frame Size for Poll Frame (FSD)**

<b>Poll Mode</b>		<b>Listen Mode</b>	
12.6.1.8	The NFC Forum Device MUST accept frames with a number of data bytes less than or equal to FSD.  The NFC Forum Device MUST resort to exception processing (Protocol Error) if it receives a frame with more than FSD data bytes.	12.6.1.9	The NFC Forum Device MUST send frames with a number of data bytes less than or equal to FSD.
12.6.1.10	The FSD supported by the NFC Forum Device MUST be <b>FSD<sub>MIN</sub></b> . Refer to Appendix A.8 for the value of <b>FSD<sub>MIN</sub></b> .	12.6.1.11	The NFC Forum Device MUST support an FSD equal to <b>FSD<sub>MIN</sub></b> . <i>The NFC Forum Device MAY accept an FSD less than <b>FSD<sub>MIN</sub></b>.</i>

The FSD, in terms of FSDI, is indicated in Table 74.

**Table 74: FSDI to FSD Conversion**

<b>FSDI</b>	<b>0h</b>	<b>1h</b>	<b>2h</b>	<b>3h</b>	<b>4h</b>	<b>5h</b>	<b>6h</b>	<b>7h</b>	<b>8h</b>	<b>9h-Fh</b>
FSD (bytes)	16	24	32	40	48	64	96	128	256	RFU

### Requirements 153: FSDI

#### Poll Mode

- 12.6.1.12 The NFC Forum Device MUST set FSDI greater than or equal to the value corresponding to  $FSD_{MIN}$ , with a maximum value of 8h. Refer to Appendix A.8 for the value of  $FSD_{MIN}$ . Refer to Table 74 for FSDI to FSD conversion.

### Requirements 154: Listen Mode Handling of RFU Values of FSDI

#### Listen Mode

- 12.6.1.13 A received value of FSDI = 9h–Fh MUST be treated by the NFC Forum Device as FSDI = 8h.

The most significant nibble (b8 to b5) is used by the NFC Forum Device in Poll Mode for bit rate selection, as shown in Table 75 and Table 76.

**Table 75: Coding of b8 and b7 of Param 2**

b8	b7	Meaning
0	0	$D_{LISTEN \rightarrow POLL} = 1$
0	1	$D_{LISTEN \rightarrow POLL} = 2$
1	0	$D_{LISTEN \rightarrow POLL} = 4$
1	1	$D_{LISTEN \rightarrow POLL} = 8$

**Table 76: Coding of b6 and b5 of Param 2**

b6	b5	Meaning
0	0	$D_{POLL \rightarrow LISTEN} = 1$
0	1	$D_{POLL \rightarrow LISTEN} = 2$
1	0	$D_{POLL \rightarrow LISTEN} = 4$
1	1	$D_{POLL \rightarrow LISTEN} = 8$

**Requirements 155: Setting the Bit Rate**

<b>Poll Mode</b>		<b>Listen Mode</b>	
12.6.1.14	The NFC Forum Device MUST select bit rates compliant with the bit rates proposed by the NFC Forum Device in Listen Mode in its SENSB_RES Response.	12.6.1.15	The NFC Forum Device in Listen Mode MUST accept the bit rates selected by the NFC Forum Device in Poll Mode in the ATTRIB Command, provided they comply with those proposed by the NFC Forum Device in Listen Mode in its SENSB_RES Response.

**Param 3**

Param 3 is used for confirmation of the protocol type and is coded as specified in Table 77.

**Table 77: Format of Param 3 of the ATTRIB Command**

<b>b8</b>	<b>b7</b>	<b>b6</b>	<b>b5</b>	<b>b4</b>	<b>b3</b>	<b>b2</b>	<b>b1</b>	<b>Meaning</b>
0	0	0	0	0				RFU
					x	x		Minimum TR2
							x	0b: NFC Forum Device in Listen Mode not compliant with [ISO/IEC_14443] 1b: NFC Forum Device in Listen Mode compliant with [ISO/IEC_14443]

**Requirements 156: Format of Param 3 of the ATTRIB Command**

<b>Poll Mode</b>	
12.6.1.16	The NFC Forum Device MUST set b1 to 1b, indicating compliance with [ISO/IEC_14443].

**Requirements 157: Listen Mode Handling of RFU Values of b8 to b4 of Param 3**

<b>Poll Mode</b>	
12.6.1.17	The NFC Forum Device in Listen Mode MUST ignore and not answer an ATTRIB Command with b8 to b4 of Param 3 different from 00000b.

**Param 4**

The Param 4 byte codes the DID as shown in Table 78.

**Table 78: Format of Param 4 of the ATTRIB Command**

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	0	0	0					RFU
				x	x	x	x	DID

**Requirements 158: Format of Param 4 of the ATTRIB Command**

Poll Mode		Listen Mode	
12.6.1.18	The NFC Forum Device MUST set the DID field to a value between 0 and 14.	12.6.1.19	The NFC Forum Device MUST be ready to receive a DID field with a value between 0 and 14.

**Requirements 159: Listen Mode Handling of RFU Value of DID**

Listen Mode	
12.6.1.20	The NFC Device in Listen Mode MUST ignore and not answer the ATTRIB command when received value of DID = 15.

**Higher layer – INF**

The Higher layer – INF field may include any higher layer Command transferable as INF field in the Half-Duplex Block Transmission protocol, as defined in Section 13.

**Requirements 160: Higher Layer – INF**

Poll Mode		Listen Mode	
<i>An NFC Forum Device in Poll Mode MAY include a higher layer Command in the Higher layer – INF field.</i>		12.6.1.21	The NFC Forum Device MUST be ready to receive an ATTRIB Command with or without Higher layer – INF field.

**12.6.2 ATTRIB Response**

An NFC Forum Device in Listen Mode answers to an ATTRIB Command with the format described in Table 79. A Valid Response to an ATTRIB Command is the means for an NFC Forum Device in Poll Mode to verify that activation of the NFC Forum Device in Listen Mode has been successful.

**Table 79: ATTRIB Response Format**

Byte 1		Byte 2 – 2+n-1
MBLI	DID	Higher layer – Response



The least significant nibble (b4 to b1) of Byte 1 contains the returned DID.

**Requirements 161: DID in ATTRIB Response**

Poll Mode		Listen Mode	
12.6.2.1	The NFC Forum Device <b>MUST</b> be ready to receive a DID field with 0 or the same value as sent before in the DID field of the ATTRIB Command.  <i>The NFC Forum Device MAY treat a DID field with a different value as Protocol Error.</i>	12.6.2.2	The NFC Forum Device <b>MUST</b> set the DID in the ATTRIB Response to the same value as received in the DID field of the preceding ATTRIB Command.  If the DID is not supported by the NFC Forum Device, it <b>MUST</b> set the DID to 0.

**NOTE** A valid answer (same DID and valid CRC\_B) to an ATTRIB command is the way that an NFC Forum Device in Poll Mode verifies that the NFC Forum Device in Listen Mode has been selected successfully.

- The most significant nibble (b8 to b5) of Byte 1 codes the Maximum Buffer Length Index (MBLI). It is used by the NFC Forum Device in Listen Mode to inform the NFC Forum Device in Poll Mode about its Maximum Buffer Length (MBL) to receive chained frames. The coding of MBLI is:
  - MBLI = 0 means that the NFC Forum Device in Listen Mode provides no information about its internal input buffer size.
  - MBLI > 0 is used to calculate the actual internal maximum buffer length (MBL) calculated with the following formula:
 
$$MBL = FSC \times 2^{MBLI-1}$$
- The Higher layer – Response field includes the answer to the higher layer Command included in the Higher layer – INF field of the ATTRIB Command.

**Requirements 162: Higher Layer – Response**

Poll Mode		Listen Mode	
12.6.2.3	The NFC Forum Device <b>MUST</b> be ready to receive an ATTRIB Response with or without a Higher – layer INF field in response to an ATTRIB Command with Higher – layer INF field.	12.6.2.4	The NFC Forum Device <b>MUST</b> respond with an empty Higher layer – Response to an ATTRIB Command without Higher layer – INF field.  <i>The NFC Forum Device MAY respond with a Higher layer – Response to an ATTRIB Command with Higher layer – INF field.</i>  <i>The NFC Forum Device MAY also respond with an empty Higher layer – Response to an ATTRIB Command with Higher layer – INF field, indicating that the Higher layer Command is not supported.</i>

## 12.7 Timing Requirements

The Type 4B Tag platform uses NFC-B frame timing rules.

### Requirements 163: Frame Timing – Type 4B Tag Platform

---

#### Poll and Listen Mode

---

- 12.7.1.1 Commands and Responses specified in Section 12 (Type 4B Tag platform) **MUST** follow the NFC-B Technology frame timing definitions as defined in Section 5.9.
-

## 13 ISO-DEP Protocol

When the NFC Forum Device, configured for Type 4A or Type 4B Tag platform has been activated, then during the Data Exchange Activity until device deactivation all Commands and Responses (i.e., the payload) are transmitted according to the Half-Duplex Block Transmission Protocol, as specified in this section.

[ISO/IEC\_14443] is the normative reference for this section.

### 13.1 Block Format

Data bytes transmitted in a frame are organized as blocks. A block complies with the data protocol layer as used in this document.

#### 13.1.1 Block

Blocks consist of the SoD, the payload, and the EoD. The SoD contains the Protocol Control Byte, referred to as PCB and the optional DID.

The EoD contains a two-byte checksum referred to as CRC. Input for CRC calculation is the SoD and the payload.

NOTE For Type 4A Tag platform, the CRC is referred to as CRC\_A; for Type 4B Tag platform, the CRC is referred to as CRC\_B.

The block format is illustrated in Figure 30.

Block (Data)				
SoD		Payload		EoD
PCB	[DID]	[INF]		CRC_1 CRC_2

Figure 30: Block Format

NOTE The terms Prologue Field, Information Field, Epilogue Field, and EDC as used in [ISO/IEC\_14443] are referred to as SoD, payload, EoD, and CRC, respectively.

#### Requirements 164: Block Length

Poll Mode		Listen Mode	
13.1.1.1	The total length of a block sent by the Reader/Writer MUST be less than or equal to FSC (FSC as specified by the Card Emulator during the protocol installation).	13.1.1.2	The total length of a block sent by the Card Emulator MUST be less than or equal to FSD (FSD as specified by the Reader/Writer during the protocol installation).
13.1.1.3	The Reader/Writer MUST be ready to receive blocks with a length less than or equal to FSD bytes. The Reader/Writer MUST treat as Protocol Errors blocks containing more than FSD bytes.	13.1.1.4	The Card Emulator MUST be ready to receive blocks with a length less than or equal to FSC bytes. The Card Emulator MUST treat as Protocol Errors blocks containing more than FSC bytes.

### 13.1.2 SoD

The SoD contains the mandatory PCB and, optionally, the DID.

- PCB

The Protocol Control Byte (PCB) is used to convey the information required to control the data transmission. The protocol defines three fundamental types of blocks:

- I-block used to convey information for use by the application layer.
- R-block used to convey positive or negative acknowledgements. An R-block never contains an INF field. The acknowledgement relates to the last received block.
- S-block used to exchange control information between the Reader/Writer and the Card Emulator. Two different types of S-blocks are defined:
  - Waiting Time eXtension (WTX) containing a 1 byte long INF field
  - DESELECT containing no INF field.

The format of the PCB depends on its type. The format of I-blocks, R-blocks, and S-blocks is shown in Table 80, Table 81, and Table 82.

**Table 80: Format of I-block PCB**

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	0							I-block
		0						RFU
			x					Chaining, if bit is set to 1b
				x				DID following, if bit is set to 1b
					0			NAD following. MUST be set to 0b
						1		MUST be set to 1b
							x	Block number

**Table 81: Format of R-block PCB**

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1	0							R-block
		1						MUST be set to 1b
			x					ACK, if bit is set to 0b NAK, if bit is set to 1b
				x				DID following, if bit is set to 1b
					0			MUST be set to 0b
						1		RFU
							x	Block number

**Table 82: Format of S-block PCB**

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1	1							S-block
		x	x					DESELECT, if set to 00b WTX, if set to 11b
				x				DID following, if bit is set to 1b
					0			MUST be set to 0b
						1		RFU
							0	RFU

- DID Field

The DID is used to identify a specific NFC Forum Device in Listen Mode. The two most significant bits—b8 and b7—code the power level indicator. The format of the DID field is shown in Table 83. See also Section 11.6.1 and Section 12.6.1.

**Table 83: Format of DID Field**

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
x	x							Power Level Indicator
		0	0					RFU
				x	x	x	x	DID

**Requirements 165: DID**

<b>Poll Mode</b>		<b>Listen Mode</b>	
13.1.2.1	The NFC Forum Device in Poll Mode <b>MUST</b> not include a DID, if the NFC Forum Device in Listen Mode does not support DID.	13.1.2.2	An NFC Forum Device in Listen Mode that does not support DID <b>MUST</b> ignore blocks including a DID.
13.1.2.3	The NFC Forum Device in Poll Mode <b>MUST</b> use the DID negotiated during the Device Activation Activity, if the NFC Forum Device in Listen Mode supports DID.	13.1.2.4	An NFC Forum Device in Listen Mode that supports DID: <ul style="list-style-type: none"> <li>• <b>MUST</b> respond to blocks containing the DID that was negotiated during the Device Activation Activity by using this DID</li> <li>• <b>MUST</b> ignore a block with a DID different from the DID negotiated during the Device Activation Activity</li> <li>• <b>MUST</b>, in case the DID negotiated during the Device Activation Activity is 0, respond to a block containing no DID by using no DID</li> </ul>

**Requirements 166: Power Level Indicator**

<b>Poll Mode</b>		<b>Listen Mode</b>	
<i>The Reader/Writer MAY support a power level indicator different from 00b.</i>		13.1.2.5	The power level indicator <b>MUST</b> be set to 00b.

**Requirements 167: Handling of RFU Values of DID Field**

<b>Listen and Poll Mode</b>	
13.1.2.6	A received value for b6-b5 that is different from 00b <b>MUST</b> be treated as Protocol Error.

**13.1.3 Payload**

The payload consists of the optional INF field. When present, the INF field conveys either application data in I-blocks or non-application data and status information in S-blocks. The length of the payload is calculated by counting the number of bytes of the whole block minus the length of the SoD and the EoD.

**13.1.4 EoD**

The EoD contains the CRC.

## Requirements 168: CRC

---

### Poll and Listen Mode

---

- 13.1.4.1 A block **MUST** contain an EoD at the position, as indicated in Figure 30.  
 For Type 4A Tag platform, the EoD **MUST** contain a CRC\_A as defined in Section 4.4.  
 For Type 4B Tag platform, the EoD **MUST** contain a CRC\_B as defined in Section 5.4.
- 

## 13.2 Protocol Operation

### 13.2.1 General Rules

General rules for the Type 4A Tag and Type 4B Tag transmission protocol are described in Section 7.

### 13.2.2 Frame Waiting Time Extension

When the Card Emulator needs more time than the defined FWT to process the received block, it uses an S(WTX) request for a waiting time extension. An S(WTX) request contains a 1-byte INF field as specified in Table 84.

**Table 84: Format of INF Field of an S(WTX) Request**

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
x	x							Power level indication
		x	x	x	x	x	x	WTXM

- Power Level Indicator

The two most significant bits—b8 and b7—code the power level indication.

### Requirements 169: Power Level Indication

Poll Mode	Listen Mode
<i>The Reader/Writer MAY support a power level indication different from 00b.</i>	13.2.2.1 The power level indication is not used and the bits <b>MUST</b> be set to 00b.

- WTXM

The bits b6 to b1 code WTXM. The WTXM is coded in the range from 1 to 59.

The Reader/Writer acknowledges the request by sending an S(WTX) Response containing also a 1-byte INF field. The INF field consists of two parts that contain the same WTXM as received in the request (see Table 85).

**Table 85: Format of INF Field of an S(WTX) Response**

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	0							RFU
		x	x	x	x	x	x	WTXM

**Requirements 170: Handling of RFU Values of INF Field of an S(WTX) Response****Listen Mode**

13.2.2.2 A received value for b8 to b7 that is different from 00b MUST be treated as Protocol Error.

The corresponding temporary value of FWT is calculated by the following formula:

$$\mathbf{FWT}_{\text{TEMP}} = \mathbf{FWT} \times \mathbf{WTXM}$$

The time  $\mathbf{FWT}_{\text{TEMP}}$  requested by the Card Emulator starts after the Reader/Writer has sent the S(WTX) Response.



## Requirements 171: Frame Waiting Time Extension

Poll Mode		Listen Mode	
13.2.2.3	The Reader/Writer MUST be ready to receive an S(WTX) having a WTXM with a value in the range from 1 to 59. The Reader/Writer MUST resort to exception processing (Protocol Error) on reception of an S(WTX) with WTXM set to 0 and 60 to 63.	13.2.2.4	The Card Emulator MUST code WTXM in the range from 1 to 59.
13.2.2.5	The Reader/Writer MUST support a frame waiting time extension less than or equal to the maximum supported FWT: $\mathbf{FWT}_{TEMP} \leq (256 \times 16/f_C) \times 2^{\mathbf{FWI}_{MAX}}$	13.2.2.6	The Card Emulator MUST code WTXM such that $\mathbf{FWT}_{TEMP}$ is less than or equal to the maximum value of FWT: $\mathbf{FWT}_{TEMP} \leq (256 \times 16/f_C) \times 2^{\mathbf{FWI}_{MAX}}$
13.2.2.7	The Reader/Writer MUST code WTXM in the S(WTX) Response with the same value as the value of WTXM in the S(WTX) request.	13.2.2.8	The Card Emulator MUST treat a WTXM in the S(WTX) Response with a value different from the value of WTXM in the S(WTX) request as a Protocol Error.
13.2.2.9	After sending the S(WTX) Response block in response to an S(WTX) request block from the Card Emulator, the Reader/Writer MUST wait at least for $\mathbf{FWT}_{TEMP} + \Delta\mathbf{FWT}$ for a block of the Card Emulator (i.e., $\mathbf{FDT}_{A,LISTEN,MAX}$ equals $\mathbf{FDT}_{TEMP} + \Delta\mathbf{FWT}$ in case of NFC-A or $\mathbf{FDT}_{B,LISTEN,MAX}$ equals $\mathbf{FDT}_{TEMP} + \Delta\mathbf{FWT}$ in case of NFC-B).  If the Reader/Writer does not receive a block from the Card Emulator within $\mathbf{FWT}_{TEMP} + \Delta\mathbf{FWT} + \Delta\mathbf{T}_{POLL}$ , then the Reader/Writer MUST treat this as a Timeout Error.	13.2.2.10	After receiving the S(WTX) Response block of the Reader/Writer, the Card Emulator MUST start sending the next block within $\mathbf{FWT}_{TEMP}$ (i.e., $\mathbf{FDT}_{A,LISTEN,MAX}$ equals $\mathbf{FDT}_{TEMP}$ in case of NFC-A or $\mathbf{FDT}_{B,LISTEN,MAX}$ equals $\mathbf{FDT}_{TEMP}$ in case of NFC-B).

*Between  $\mathbf{FWT}_{TEMP} + \Delta\mathbf{FWT}$  and  $\mathbf{FWT}_{TEMP} + \Delta\mathbf{FWT} + \Delta\mathbf{T}_{POLL}$ , the Reader/Writer MAY accept the Response or MAY generate a Timeout Error.*

Poll Mode	Listen Mode
13.2.2.11 The Reader/Writer MUST apply <b>FWT<sub>TEMP</sub> + ΔFWT</b> only until the next block has been received from the Card Emulator or until the Reader/Writer resorts to exception processing.	13.2.2.12 The Card Emulator MUST apply <b>FWT<sub>TEMP</sub></b> only until the next block has been sent by the Card Emulator.

### 13.2.3 Chaining

The chaining feature allows the Reader/Writer or the Card Emulator to transmit information that does not fit in a single block as defined by FSC or FSD, respectively, by dividing the information into several blocks.

#### Requirements 172: Chaining Rules

Poll and Listen Mode	
13.2.3.1	The chaining bit in the PCB of an I-block controls the chaining of blocks. When an I-block indicating chaining is received, the block MUST be acknowledged by an R(ACK) block.

#### Requirements 173: Block Sizes during Chaining

Poll Mode	
13.2.3.2	When the Reader/Writer sends a chain of I-blocks, each block that indicates chaining (i.e., all blocks of the chain except the last one) MUST have a length equal to FSC.

#### Requirements 174: Last Block

Poll Mode	Listen Mode
13.2.3.3	<div style="display: flex;"> <div style="flex: 1;"> <p>The Reader/Writer MUST not send an empty I-block (i.e., without an INF field) not indicating chaining after an I-block indicating chaining (i.e., the last block in a chain of blocks is not allowed to be empty).</p> </div> <div style="flex: 1; padding-left: 20px;"> <p><i>The Card Emulator MAY treat receipt of an empty I-block not indicating chaining after receipt of an I-block indicating chaining as a Protocol Error.</i></p> </div> </div>

### 13.2.4 Block Numbering Rules

This section specifies the rules for block numbering in a block of the Reader/Writer and Card Emulator.

### Requirements 175: Block Numbering Rules

Poll Mode		Listen Mode	
13.2.4.1	The block number of the Reader/Writer MUST be initialized to 0 for the current activated Card Emulator.	13.2.4.2	The block number of the Card Emulator MUST be initialized to 1 at activation.
13.2.4.3	When a Valid I-block or a Valid R(ACK) block with a block number equal to the current block number is received, the Reader/Writer MUST toggle the current block number for the current Card Emulator before optionally sending a block.	13.2.4.4	When an I-block is received, the Card Emulator MUST toggle its block number before sending a block.  <i>The Card Emulator MAY check if the received block number is not in compliance with the rules of the Reader/Writer to decide not to toggle its internal block number nor to send a Response block.</i>
		13.2.4.5	When an R(ACK) block with a block number not equal to the current Card Emulator's block number is received, the Card Emulator MUST toggle its block number before sending a block.  <i>The Card Emulator MAY check if the received block number is not in compliance with rules of the Reader/Writer to decide not to toggle its internal block number nor to send a Response block.</i>

### 13.2.5 Block Handling Rules

This section specifies the block handling rules for the Reader/Writer and the Card Emulator.

#### Requirements 176: General Block Handling Rules

Poll and Listen Mode	
13.2.5.1	The first block MUST be sent by the Reader/Writer.
13.2.5.2	S-blocks are only used in pairs. An S(...) Request block MUST always be followed by an S(...) Response block except for the case where the Reader/Writer does not want to accept another S(WTX) Request block anymore. The Reader/Writer MUST respond to the first S(WTX) Request that it receives after sending an I-block or R-block to the Card Emulator.  <i>The Reader/Writer MAY stop accepting subsequent S(WTX) Requests preventing the Reader/Writer to be blocked by arbitrary long Card Emulator processing times. Nevertheless the Reader/Writer SHOULD accept a number of S(WTX) Requests to allow the normal operation to finish on the Card Emulator side.</i>

---

**Requirements 177: Block Handling Rules for the Reader/Writer**


---

**Poll Mode**


---

13.2.5.3 When an R(ACK) block is received with a block number not equal to the Reader/Writer's current block number, then the Reader/Writer MUST re-transmit the last I-block if this R(ACK) block is received in response to an R(NAK) block sent by the Reader/Writer to notify a timeout.

*In all other cases, when an R(ACK) block with a block number not equal to the Reader/Writer's current block number is received, then the Reader/Writer MAY raise the Protocol Error protocol exception or the Reader/Writer MAY re-transmit the last I-block.*

---

13.2.5.4 When the Reader/Writer has re-transmitted an I-block two times (i.e., the same I-block was sent three times), it MUST be treated as a Protocol Error if an R(ACK) block with a block number not equal to the Reader/Writer's current block number is received..

---

13.2.5.5 When an R(ACK) block is received, chaining MUST be continued if its block number is equal to the Reader/Writer's current block number and the last I-block sent by the Reader/Writer indicates chaining. If the last I-block sent by the Reader/Writer did not indicate chaining, then the Reader/Writer MUST treat the R(ACK) block as a Protocol Error.

---

13.2.5.6 When an R(NAK) block is received by the Reader/Writer, it MUST be treated as a Protocol Error.

---

---

**Requirements 178: Block Handling Rules for the Card Emulator**


---

**Listen Mode**

13.2.5.7 The Card Emulator is permitted to send an S(WTX) block instead of an I-block or an R(ACK) block (except in the case of a retransmitted I-block or a retransmitted R(ACK) block).

---

13.2.5.8 When an I-block not indicating chaining is received, the block **MUST** be acknowledged by an I-block.

*If the received I-block not indicating chaining is empty (i.e., without an INF field), then the mandatory I-block sent MAY either be empty or contain any applicative information (e.g., error code).*

---

13.2.5.9 When an R(ACK) or an R(NAK) block is received with a block number equal to the Card Emulator's current block number, then:

- If the last block was sent by the Card Emulator (i.e., the last block from the Card Emulator has not been acknowledged by the Reader/Writer), then the last block **MUST** be retransmitted.
- If the last block was sent by the Reader/Writer (i.e., the last block from the Card Emulator has been acknowledged by the Reader/Writer), then the next block **MUST** be sent.

---

13.2.5.10 When an R(NAK) block is received, an R(ACK) block **MUST** be sent if its block number is not equal to the Card Emulator's current block number.

---

13.2.5.11 When an R(ACK) block is received, chaining **MUST** be continued if its block number is not equal to the Card Emulator's current block number and the last I-block sent by the Card Emulator indicates chaining.

*If the last I-block sent by the Card Emulator did not indicate chaining, then the Card Emulator MAY treat the R(ACK) block as a Protocol Error.*

---

**13.2.6 Exception Processing**

When errors are detected the following error handling is attempted.

---

**Requirements 179: Exception Processing – Card Emulator**


---

**Listen Mode**

13.2.6.1 The Card Emulator **MUST** detect Transmission Errors (frame error or EDC error) and Protocol Errors (infringement of the protocol rules).

---

13.2.6.2 The Card Emulator **MUST NOT** attempt error recovery. The Card Emulator **MUST** always stay in receive mode when a Transmission Error or a Protocol Error occurs.

*Note that an R(NAK) block is never sent by the Card Emulator.*

---

---

**Requirements 180: Exception Processing – Reader/Writer**


---

**Poll Mode**

13.2.6.3 If a block with a Transmission Error is received after receipt of a block not indicating chaining (except in the case of S(DESELECT)), then the Reader/Writer MUST send an R(NAK) block if all of the following are true:

- The CRC is incorrect or the frame has a parity error.
- The block is encapsulated in a frame of at least 4 characters long.

The Reader/Writer MUST ignore all other Transmission Errors and MUST be ready to process a Valid Block within a time  $t_{RECOVERY}$  after the end of the block with the Transmission Error.

The Reader/Writer MUST send the R(NAK) block within a time  $t_{RETRANSMISSION}$  measured from the start of the block with the Transmission Error.

If a Transmission Error is received after sending an R(NAK) block the Reader/Writer MUST send another R(NAK) block until  $n_{RETRY,NAK}$  consecutive R(NAK) blocks have been sent. If no Valid Response to all R(NAK) block is received, then the Reader/Writer MUST RAISE THE Transmission Error PROTOCOL EXCEPTION.

Refer to Appendix A.9 for the values of  $n_{RETRY,NAK}$ ,  $t_{RECOVERY}$  and  $t_{RETRANSMISSION}$ .

---

13.2.6.4 If a block with a Protocol Error is received after receipt of a block not indicating chaining (except in the case of S(DESELECT)), then the Reader/Writer MUST RAISE THE Protocol Error PROTOCOL EXCEPTION.

---

13.2.6.5 If a Timeout Error occurs after receipt of a block not indicating chaining (except in the case of S(DESELECT)), then the Reader/Writer MUST send an R(NAK) block. The Reader/Writer MUST send the R(NAK) block after  $FWT+\Delta FWT$  and before  $FWT+\Delta FWT+t_{RETRANSMISSION}$ . If a Timeout Error occurs after sending an R(NAK) block the Reader/Writer MUST send another R(NAK) block until  $n_{RETRY,NAK}$  consecutive R(NAK) blocks have been sent. If no Valid Response to all R(NAK) blocks is received or if the Reader/Writer has detected a time-out following  $n_{RETRY,WTX} + 1$  consecutive S(WTX) Response blocks (i.e., the Reader/Writer detects three times a Timeout Error following an S(WTX) Response without receiving a Valid I-block or R-block from the Card Emulator), then the Reader/Writer MUST RAISE THE Timeout Error PROTOCOL EXCEPTION.

Refer to Appendix A.9 for the values of  $n_{RETRY,WTX}$ .

---

---

**Poll Mode**


---

- 13.2.6.6 If a block with a Transmission Error is received after receipt of a block indicating chaining, then the last R(ACK) block sent by the Reader/Writer MUST be retransmitted if all of the following are true:
- The CRC is incorrect or the frame has a parity error.
  - The block is encapsulated in a frame of at least 4 characters long.
- The Reader/Writer MUST ignore all other Transmission Errors and MUST be ready to process a Correct Frame within  $t_{\text{RECOVERY}}$  after the end of the block with the Transmission Error.
- The Reader/Writer MUST send the R(ACK) block within  $t_{\text{RETRANSMISSION}}$  measured from the start of the block with the Transmission Error.
- If a Transmission Error is received after sending an R(ACK) block the Reader/Writer MUST send another R(ACK) the Reader/Writer until  $n_{\text{RETRY,ACK}} + 1$  consecutive R(ACK) blocks have been sent. If no Valid Response to all R(ACK) blocks is received, then the Reader/Writer MUST RAISE THE Transmission Error PROTOCOL EXCEPTION.
- Refer to Appendix A.9 for the values of  $n_{\text{RETRY,ACK}}$ .
- 
- 13.2.6.7 If a block with a Protocol Error is received after receipt of a block indicating chaining, then the Reader/Writer MUST RAISE THE Protocol Error PROTOCOL EXCEPTION.
- 
- 13.2.6.8 If a Timeout Error occurs after receipt of a block indicating chaining, the last R(ACK) block sent by the Reader/Writer MUST be retransmitted to ask for retransmission until  $n_{\text{RETRY,ACK}} + 1$  consecutive R(ACK) blocks have been sent. The Reader/Writer MUST send the R(ACK) block after  $\text{FWT} + \Delta\text{FWT}$  and before  $\text{FWT} + \Delta\text{FWT} + t_{\text{RETRANSMISSION}}$ . If no Valid Response to all R(ACK) blocks is received or if the Reader/Writer has detected a time-out following  $n_{\text{RETRY,WTX}} + 1$  consecutive S(WTX) Response blocks (i.e., the Reader/Writer detects three times a Timeout Error following an S(WTX) Response without receiving a Valid I-block or R-block from the Card Emulator), then the Reader/Writer MUST RAISE THE Timeout Error PROTOCOL EXCEPTION.
- 

### 13.2.7 De-activation Rules

This section lists the requirements related to the deactivation of the NFC Forum Device in Listen Mode. The NFC Forum Device in Poll Mode deactivates the NFC Forum Device in Listen Mode by sending an S(DESELECT) request block. The NFC Forum Device in Listen Mode acknowledges the deactivation by returning an S(DESELECT) response block.

### Requirements 181: Deactivation Rules

Poll Mode		Listen Mode	
13.2.7.1	The NFC Forum Device in Poll Mode MUST deactivate the NFC Forum Device in Listen Mode by sending an S(DESELECT) request block. The S(DESELECT) request block MUST be coded as an S-block as specified in Table 82.	13.2.7.2	The NFC Forum Device in Listen Mode MUST acknowledge the deactivation request by sending an S(DESELECT) response block. The S(DESELECT) response block MUST be coded as an S-block as specified in Table 82.
13.2.7.3	Following the sending of the S(DESELECT) request block to the NFC Forum Device in Listen Mode, the NFC Forum Device in Poll Mode MUST wait at least <b>FWT<sub>DEACTIVATION</sub></b> for the S(DESELECT) response block from the NFC Forum Device in Listen Mode.	13.2.7.4	The NFC Forum Device in Listen Mode MUST send an S(DESELECT) response block to the NFC Forum Device in Poll Mode within <b>FWT<sub>DEACTIVATION</sub></b> (refer to <b>FWT<sub>DEACTIVATION</sub></b> in Appendix A.9).

### Requirements 182: Error Handling for Deactivation

Poll Mode	
13.2.7.5	If a Protocol Error is detected in the S(DESELECT) response block, then the NFC Forum Device in Poll Mode MUST RAISE THE Protocol Error PROTOCOL EXCEPTION.
13.2.7.6	When a Transmission Error is detected in the S(DESELECT) response block, then the NFC Forum Device in Poll Mode MUST resend the S(DESELECT) request block until <b>n<sub>RETRY,DESELECT</sub></b> consecutive S(DESELECT) request blocks have been sent. The NFC Forum Device in Poll Mode MUST resend the S(DESELECT) block within a delay of <b>t<sub>RETRANSMISSION</sub></b> measured from the start of the S(DESELECT) response block containing the Transmission Error. If no valid S(DESELECT) response block to all S(DESELECT) request blocks is received, then the NFC Forum Device in Poll Mode MUST RAISE THE Timeout Error PROTOCOL EXCEPTION. Refer to Appendix A.9 for the values of <b>n<sub>RETRY,DESELECT</sub></b> .
13.2.7.7	When no S(DESELECT) response block is received within the delay <b>FWT<sub>DEACTIVATION</sub></b> , then the NFC Forum Device in Poll Mode MUST resend the S(DESELECT) request block until <b>n<sub>RETRY,DESELECT</sub></b> consecutive S(DESELECT) request blocks have been sent. The NFC Forum Device MUST resend the S(DESELECT) request block after <b>FWT<sub>DEACTIVATION</sub></b> and before <b>FWT<sub>DEACTIVATION</sub> + t<sub>RETRANSMISSION</sub></b> . If no valid S(DESELECT) response block to all S(DESELECT) request blocks is received, then the NFC Forum Device MUST RAISE THE Timeout Error PROTOCOL EXCEPTION.



### 13.3 Timing Requirements

The Type 4A and Type 4B Tag Half-duplex Protocol uses Type 4A Tag Platform Timing or Type 4B Tag Platform Timing depending if NFC-A or if NFC-B technology is used.

---

#### Requirements 183: Timing – Type 4A and Type 4B Tag Half-duplex Protocol

---

##### Poll and Listen Mode

---

- |          |  |
|----------|--|
| 13.3.1.1 | Commands and Responses specified in Section 13 MUST follow the Timing Requirements of Type 4A Tag Platform as defined in Section 11.7 if NFC-A technology is used. |
| <hr/>    |  |
| 13.3.1.2 | Commands and Responses specified in Section 13 MUST follow the Timing Requirements of Type 4B Tag Platform as defined in Section 12.7 if NFC-B technology is used. |
-

## 14 NFC-DEP Protocol

This section specifies the NFC-DEP Protocol of the NFC Forum Device. Information, when already defined elsewhere, have been included for readability reasons.

[ISO/IEC\_18092] is the normative reference for this section.

### 14.1 Sequence Format

NFC Forum Devices use NFC-A sequence format or NFC-F sequence format for the NFC-DEP Protocol, depending on the configuration.

---

#### Requirements 184: Sequence Format – NFC-DEP Protocol

---

##### Initiator and Target

---

- 14.1.1.1 Commands and Responses specified in Section 14 (NFC-DEP Protocol) MUST be transmitted using:
- NFC-A Technology sequence format, as defined in Section 4.1, if the NFC Forum Device is using NFC-A Technology, or
  - NFC-F Technology sequence format, as defined in Section 6.1, if the NFC Forum Device is using NFC-F Technology.
- 

### 14.2 Bit Level Coding

NFC Forum Devices use NFC-A bit level coding or NFC-F bit level coding for the NFC-DEP Protocol, depending on the configuration.

---

#### Requirements 185: Bit Level Coding – NFC-DEP Protocol

---

##### Initiator and Target

---

- 14.2.1.1 Commands and Responses specified in Section 14 (NFC-DEP Protocol) MUST be transmitted using:
- NFC-A Technology bit level coding, as defined in Section 4.2, if the NFC Forum Device is using NFC-A Technology, or
  - NFC-F Technology bit level coding, as defined in Section 6.2, if the NFC Forum Device is using NFC-F Technology.
-

### 14.3 Frame Format

NFC Forum Devices use NFC-A frame format or NFC-F frame format for the NFC-DEP Protocol, depending on the configuration.

**Requirements 186: Frame Format – NFC-DEP Protocol**

<b>Initiator and Target</b>	
14.3.1.1	<p>Commands and Responses specified in Section 14 (NFC-DEP Protocol) MUST be transmitted within:</p> <ul style="list-style-type: none"> <li>• NFC-A Technology standard frames as defined in Section 4.3, if the NFC Forum Device is using NFC-A Technology, or</li> <li>• NFC-F Technology frames as defined in Section 6.3, if the NFC Forum Device is using NFC-F Technology.</li> </ul>

### 14.4 Data and Payload Format

Data transmitted in a frame has the following substructure and consists of an SoD, the payload, and an EoD.

The SoD contains a start byte SB (NFC-A only), followed by a length byte LEN, indicating the number of payload bytes.

The payload consists of the Commands and Responses as described in Section 14.5.

The EoD contains a two-byte checksum, referred to as CRC. Input for the CRC calculation is the SoD and the payload.

The NFC-DEP Protocol data and payload format is illustrated in Figure 31.

Data							
SoD		Payload				EoD	
SB (NFC-A only)	LEN	Byte 1	Byte 2	...	Byte n	CRC_1	CRC_2

**Figure 31: Data and Payload Format – NFC-DEP Protocol**

### Requirements 187: Data and Payload Format – NFC-DEP Protocol

Initiator and Target	
14.4.1.1	If configured for NFC-A Technology, the first byte in the SoD MUST be a start byte (SB) set to the value F0h.
14.4.1.2	The SoD MUST contain a length byte LEN at the position shown in Figure 31 with a value equal to n+1, where n indicates the number of bytes the payload consists of.
14.4.1.3	The SoD MUST have a value between 3 and 255. The NFC Forum Device MUST treat other values as a Transmission Error.
14.4.1.4	The payload MUST follow the SoD as indicated in Figure 31.
14.4.1.5	<p>The payload, i.e., Commands and Responses specified in Section 14 (NFC-DEP Protocol), MUST be followed by an EoD at the position, as indicated in Figure 31.</p> <ul style="list-style-type: none"> <li>• The EoD MUST contain a CRC_A as defined in Section 4.4, if the NFC Forum Device is using NFC-A Technology. CRC_1 is the LSB and CRC_2 is the MSB.</li> <li>• The EoD MUST contain a CRC_F as defined in Section 6.4, if the NFC Forum Device is using NFC-F Technology. CRC_1 is the MSB and CRC_2 is the LSB.</li> </ul>
14.4.1.6	<p>The NFC Forum Device MUST verify a checksum contained in the EoD as follows:</p> <ul style="list-style-type: none"> <li>• If the NFC Forum Device is using NFC-A Technology, then the NFC Forum Device MUST verify a CRC_A on receipt, as defined in Section 4.4, and MUST treat failure of verification as a Transmission Error.</li> <li>• If the NFC Forum Device is using NFC-F Technology, then the NFC Forum Device MUST verify a CRC_F on receipt, as defined in Section 6.4, and MUST treat failure of verification as a Transmission Error.</li> </ul>

## 14.5 Command Set

Payloads exchanged between NFC Forum Devices consist of Commands and Responses. Table 86 lists the Commands that are available to the Initiator. For each Command, the corresponding Response from the Target is indicated.

**Table 86: NFC-DEP Protocol - Command Set**

Initiator (Command)	Target (Response)
ATR_REQ	ATR_RES
PSL_REQ	PSL_RES
DEP_REQ	DEP_RES
DSL_REQ	DSL_RES
RLS_REQ	RLS_RES

This section details the format of these Commands and their Responses.

## 14.6 Attribute Request (ATR\_REQ)

The Attribute Request Command (ATR\_REQ) is used by the Initiator to activate a Target. The Target responds to the ATR\_REQ with the Attribute Request Response (ATR\_RES).

### 14.6.1 ATR\_REQ Length

**Requirements 188: Attribute Request Length**

Initiator		Target	
14.6.1.1	The number of bytes of the ATR_REQ Command MUST be less than or equal to 64.	14.6.1.2	The Target MUST treat an ATR_REQ Command with a number of bytes greater than 64 as a Protocol Error.
14.6.1.3	The Initiator MUST treat an ATR_RES Response with a number of bytes greater than 64 as a Protocol Error.	14.6.1.4	The number of bytes of the ATR_RES Response MUST be less than or equal to 64.

### 14.6.2 ATR\_REQ Command

Table 87 defines the ATR\_REQ format.

**Table 87: ATR\_REQ Format**

Byte 1	Byte 2	Byte 3–12	Byte 13	Byte 14	Byte 15	Byte 16	Byte 17–17+n
D4h	00h	NFCID <sub>3i</sub>	DID <sub>i</sub>	BS <sub>i</sub>	BR <sub>i</sub>	PP <sub>i</sub>	[G <sub>i0</sub> ... G <sub>in</sub> ]

#### NFCID<sub>3i</sub>

**NFCID<sub>3i</sub>** is the NFC Forum Device identifier of the Initiator for the NFC-DEP Protocol.

**Requirements 189: NFCID<sub>3i</sub> Format**

Initiator		Target	
14.6.2.1	If the Initiator is using NFC-F Technology, the Initiator MUST fill Byte 3 to Byte 10 of ATR_REQ with NFCID <sub>2</sub> of the Target it wants to address.  <i>Byte 11 to Byte 12 of ATR_REQ MAY be filled with Any Value.</i>	14.6.2.2	If the Target is using NFC-F Technology, the Target MUST ignore an ATR_REQ with Byte 3 to Byte 10 different from its NFCID <sub>2</sub> .

#### DID<sub>i</sub>

The Initiator Device Identification Number (DID<sub>i</sub>) is used to identify different Targets that are activated at one time. If multiple target activation is not used, the DID<sub>i</sub> field is set to zero.

**Requirements 190: Initiator Device Identification Number (DID<sub>i</sub>)**

Initiator		Target	
14.6.2.3	The Initiator <b>MUST</b> set the DID <sub>i</sub> field to a value between 0 and 14.	14.6.2.4	The Target <b>MUST</b> be ready to receive a value transmitted by the Initiator in the <b>DID<sub>i</sub></b> field between 0 and 14.  <i>The Target <b>MAY</b> treat a <b>DID<sub>i</sub></b> field with a different value as a Protocol Error.</i>

**BS<sub>i</sub> and BR<sub>i</sub>**

**BS<sub>i</sub>** and **BR<sub>i</sub>** indicate the bit rates in Active Communication mode supported by the Initiator in both transmission directions. The coding of **BS<sub>i</sub>** and **BR<sub>i</sub>** is specified in Table 88 and Table 89.

**Table 88: Bit Rates Supported by Initiator in Sending Direction (BS<sub>i</sub>)**

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	0	0	0					RFU
				x				<b>D<sub>POLL→LISTEN</sub></b> =64 supported, if bit is set to 1b.
					x			<b>D<sub>POLL→LISTEN</sub></b> =32 supported, if bit is set to 1b.
						x		<b>D<sub>POLL→LISTEN</sub></b> =16 supported, if bit is set to 1b.
							x	<b>D<sub>POLL→LISTEN</sub></b> =8 supported, if bit is set to 1b.

**Table 89: Bit Rates Supported by Initiator in Receiving Direction (BR<sub>i</sub>)**

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	0	0	0					RFU
				x				<b>D<sub>LISTEN→POLL</sub></b> =64 supported, if bit is set to 1b.
					x			<b>D<sub>LISTEN→POLL</sub></b> =32 supported, if bit is set to 1b.
						x		<b>D<sub>LISTEN→POLL</sub></b> =16 supported, if bit is set to 1b.
							x	<b>D<sub>LISTEN→POLL</sub></b> =8 supported, if bit is set to 1b.

**Requirements 191: BS<sub>i</sub> and BR<sub>i</sub> Coding**

Initiator		Target	
14.6.2.5	The Initiator <b>MUST</b> set b1-b4 of the <b>BS<sub>i</sub></b> and <b>BR<sub>i</sub></b> field to 0000b.	14.6.2.6	The Target <b>MUST</b> ignore the values.

**NOTE** In this version of the document, the Active Communication mode is not supported, and therefore, b1-b4 of **BS<sub>i</sub>** and **BR<sub>i</sub>** are set to 0000b.

**Presence of Optional Parameters (PP<sub>i</sub>)**

The **PP<sub>i</sub>** field indicates the Length Reduction field (**LR<sub>i</sub>**) and the presence of optional parameters. The format of the **PP<sub>i</sub>** byte is specified in Table 90.

**Table 90: PP<sub>i</sub> Format**

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	0							RFU
		x	x					<b>LR<sub>i</sub></b>
				0	0			RFU
						x		General ( <b>G<sub>i</sub></b> ) bytes available, if set to 1b.
							x	NAD used, if set to 1b.

The Length Reduction (**LR<sub>i</sub>**) bits restrict the size of the Responses (payload) of the Target. Table 91 defines the **LR<sub>i</sub>** coding.

**Table 91: LR<sub>i</sub> Coding**

b6	b5	Meaning
0	0	Maximum payload size is 64 bytes.
0	1	Maximum payload size is 128 bytes.
1	0	Maximum payload size is 192 bytes.
1	1	Maximum payload size is 254 bytes.

NOTE The maximum payload size of 254 bytes for b6 b5 = 11b overrules [ISO/IEC\_18092].

**Requirements 192: PP Format**

Initiator	
14.6.2.7	The Initiator <b>MUST</b> be ready to receive Responses containing a number of payload bytes less than or equal to the value as specified by <b>LR<sub>i</sub></b> .

The NAD bit indicates whether the Initiator uses NAD or not.

**Requirements 193: NAD**

Initiator		Target	
14.6.2.8	The Initiator <b>MUST</b> not use NAD and set b1 of <b>PP<sub>i</sub></b> to 0b.	14.6.2.9	The Target <b>MUST</b> disregard b1 of <b>PP<sub>i</sub></b> .

## General Bytes (G<sub>i</sub>)

The General Bytes (G<sub>i</sub>) are optional and used to provide general information. The maximum number of General Bytes is calculated by the maximum number of data bytes in the ATR\_REQ Command, subtracted by the number of mandatory data bytes in the ATR\_REQ Command.

### 14.6.3 ATR\_RES Response

Table 92 defines the ATR\_RES format.

Table 92: ATR\_RES Format

Byte 1	Byte 2	Byte 3–12	Byte 13	Byte 14	Byte 15	Byte 16	Byte 17	Byte 18–18+n
D5h	01h	NFCID3 <sub>T</sub>	DID <sub>T</sub>	BS <sub>T</sub>	BR <sub>T</sub>	TO	PP <sub>T</sub>	[G <sub>T0</sub> ... G <sub>Tn</sub> ]

#### NFCID3<sub>T</sub>

NFCID3<sub>T</sub> is the NFC Forum Device identifier of the Target for the NFC-DEP Protocol.

Requirements 194: NFCID3<sub>T</sub> Format

Initiator	Target
<i>The Initiator MAY treat an NFCID3<sub>T</sub> format that differs from Requirement 14.6.3.1 as a Protocol Error.</i>	14.6.3.1 If the Target is using NFC-F Technology, the Target MUST fill Byte 3 to Byte 10 of ATR_RES with its NFCID2.  <i>Byte 11 to Byte 12 of ATR_RES MAY be filled with Any Value.</i>

#### DID<sub>T</sub>

In the DID<sub>T</sub> field, the Target returns the same value as received in the DID<sub>I</sub> field of the preceding ATR\_REQ Command.

Requirements 195: Target Device Identification Number (DID<sub>T</sub>)

Initiator	Target
14.6.3.2 The Initiator MUST be ready to receive an ATR_RES Response containing a DID <sub>T</sub> byte with the same value as sent in the DID <sub>I</sub> field of the preceding ATR_REQ Command.  <i>The Initiator MAY treat an ATR_RES Response containing a different DID<sub>T</sub> byte as a Protocol Error.</i>	14.6.3.3 In the DID <sub>T</sub> field, the Target MUST return the same value as received in the DID <sub>I</sub> field of the preceding ATR_REQ Command.

#### BS<sub>T</sub> and BR<sub>T</sub>

BS<sub>T</sub> and BR<sub>T</sub> indicate the bit rates in Active Communication mode supported by the Target in both transmission directions. The format of BS<sub>T</sub> and BR<sub>T</sub> is specified in Table 93 and Table 94.



**Table 93: Bit Rates Supported by Target in Sending Direction ( $BS_T$ )**

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	0	0	0					RFU
				x				$D_{LISTEN \rightarrow POLL} = 64$ supported, if bit is set to 1b.
					x			$D_{LISTEN \rightarrow POLL} = 32$ supported, if bit is set to 1b.
						x		$D_{LISTEN \rightarrow POLL} = 16$ supported, if bit is set to 1b.
							x	$D_{LISTEN \rightarrow POLL} = 8$ supported, if bit is set to 1b.

**Table 94: Bit Rates Supported by Target in Receiving Direction ( $BR_T$ )**

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	0	0	0					RFU
				x				$D_{POLL \rightarrow LISTEN} = 64$ supported, if bit is set to 1b.
					x			$D_{POLL \rightarrow LISTEN} = 32$ supported, if bit is set to 1b.
						x		$D_{POLL \rightarrow LISTEN} = 16$ supported, if bit is set to 1b.
							x	$D_{POLL \rightarrow LISTEN} = 8$ supported, if bit is set to 1b.

**Requirements 196:  $BS_T$  and  $BR_T$  Format**

Initiator		Target	
14.6.3.4	The Initiator MUST ignore the values.	14.6.3.5	The Target MUST set b1-b4 of the $BS_T$ and $BR_T$ field to 0000b.

NOTE b1-b4 of  $BS_T$  and  $BR_T$  are set to 0000b because the Active Communication mode is not supported in this version of the document.

**TO**

The least significant nibble b4 to b1 of the TO field is the Waiting Time (WT) and is used by the Target to code the Response Waiting Time (RWT). The format of the TO field is specified in Table 95. Refer to Section 14.11 for the definition of the RWT. WT has a value in the range from 0 to 14. The value 15 is RFU.

**Table 95: TO Format**

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	0	0	0					RFU
				x	x	x	x	WT

**Requirements 197: TO Format**

<b>Initiator</b>		<b>Target</b>	
14.6.3.6	The Initiator <b>MUST</b> support a TO indicating a WT less than or equal to <b>WT<sub>MAX</sub></b> . Refer to Appendix A.10 for the value of <b>WT<sub>MAX</sub></b> .	14.6.3.7	The Target <b>MUST</b> set WT less than or equal to <b>WT<sub>MAX</sub></b> . Refer to Appendix A.10 for the value of <b>WT<sub>MAX</sub></b> .

**Requirements 198: Initiator Handling of RFU Value of WT**

<b>Initiator</b>	
14.6.3.8	A received value of WT = 15 <b>MUST</b> be treated by the Initiator as WT = 14.

**Presence of Optional Parameters (PP<sub>T</sub>)**

The **PP<sub>T</sub>** field indicates the Length Reduction field (**LR<sub>T</sub>**) and the presence of optional parameters. The format of the **PP<sub>T</sub>** byte is specified in Table 96.

**Table 96: PP<sub>T</sub> Format**

<b>b8</b>	<b>b7</b>	<b>b6</b>	<b>b5</b>	<b>b4</b>	<b>b3</b>	<b>b2</b>	<b>b1</b>	<b>Meaning</b>
0	0							RFU
		x	x					<b>LR<sub>T</sub></b>
				0	0			RFU
						x		General bytes available, if set to 1b.
							x	NAD used, if set to 1b.

The Length Reduction (**LR<sub>T</sub>**) bits restrict the size of the Commands (payload) of the Initiator. Table 97 defines the **LR<sub>T</sub>** coding.

**Table 97: LR<sub>T</sub> Coding**

<b>b6</b>	<b>b5</b>	<b>Meaning</b>
0	0	Maximum payload size is 64 bytes.
0	1	Maximum payload size is 128 bytes.
1	0	Maximum payload size is 192 bytes.
1	1	Maximum payload size is 254 bytes.

**NOTE** The maximum payload size of 254 bytes for b6 b5 = 11b overrules [ISO/IEC\_18092].

### Requirements 199: PP Format

Target	
14.6.3.9	The Target MUST be ready to receive Commands containing a number of payload bytes less than or equal to the value as specified by $LR_T$ .

The NAD bit indicates whether the Target uses NAD or not.

### Requirements 200: NAD

Initiator	Target
14.6.3.10	14.6.3.11
The Initiator MUST ignore b1 of $PP_T$ .	The Target MUST not use NAD and set b1 of $PP_T$ to 0b.

### General Bytes

General Bytes ( $G_T$ ) are optional and are used to provide general information. The maximum number of General Bytes is calculated by the maximum number of data bytes in the ATR\_RES Response, subtracted by the number of mandatory data bytes in the ATR\_RES Response.

## 14.7 Parameter Selection Request (PSL\_REQ)

The PSL\_REQ Command is used to switch communication parameters for the subsequent data exchange through the NFC-DEP Protocol.

### 14.7.1 PSL\_REQ Command

The format of the PSL\_REQ Command is specified in Table 98.

Table 98: Format of PSL\_REQ Command

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
D4h	04h	DID	BRS	FSL

### Requirements 201: DID – PSL\_REQ Command

Initiator	Target
14.7.1.1	14.7.1.2
The Initiator MUST send the same DID as in the preceding ATR_REQ.	The Target MUST accept a PSL_REQ Command with the same DID as received in the preceding ATR_REQ Command. The Target MUST ignore a PSL_REQ Command with a DID that is different from the DID received in the preceding ATR_REQ.

The BRS byte specifies the selected bit rates for Initiator and Target, and is formatted as shown in Table 99.

**Table 99: Format of BRS**

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	0							RFU
		x	x	x				DSI
					x	x	x	DRI

The DSI codes the bit rate in communication direction from Initiator to Target. The DRI codes the bit rate in communication direction from Target to Initiator, as specified in Table 100.

**Table 100: Coding of DSI and DRI**

b6	b5	b4 (DSI)	Divisor D
b3	b2	b1 (DRI)	
0	0	0	1
0	0	1	2
0	1	0	4
0	1	1	8
1	0	0	16
1	0	1	32
1	1	0	64
1	1	1	RFU

The FSL byte defines the maximum length of Commands and Responses (number of payload bytes), as specified in Table 101. If no PSL\_REQ Command is sent, then the value exchanged in ATR\_REQ/RES is used.

**Table 101: Format of FSL**

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	0	0	0	0	0			RFU
						x	x	Length Reduction value: 00b: Maximum payload size is 64 bytes 01b: Maximum payload size is 128 bytes 10b: Maximum payload size is 192 bytes 11b: Maximum payload size is 254 bytes

NOTE The value 254 for b2 b1 = 11b overrules [ISO/IEC\_18092].

## Requirements 202: FSL

Initiator	Target
14.7.1.3	<p>The Initiator MUST set FSL to a value that defines a maximum length of Commands and Responses that is less than or equal to the minimum of the lengths defined by <math>LR_I</math> and <math>LR_T</math>.</p> <p><i>The Initiator MAY treat Responses containing a number of payload bytes greater than the value as specified by FSL as a Protocol Error.</i></p>
14.7.1.4	14.7.1.5
<p>The Initiator MUST be ready to receive Responses containing a number of payload bytes less than or equal to the value as specified by FSL.</p> <p><i>The Initiator MAY treat Responses containing a number of payload bytes greater than the value as specified by FSL as a Protocol Error.</i></p>	<p>The Target MUST be ready to receive Commands containing a number of payload bytes less than or equal to the value as specified by FSL.</p> <p><i>The Target MAY treat Commands containing a number of payload bytes greater than the value as specified by FSL as a Protocol Error.</i></p>

## 14.7.2 PSL\_RES Response

Table 102 defines the format of the PSL\_RES Response.

Table 102: Format of the PSL\_RES Response

Byte 1	Byte 2	Byte 3
D5h	05h	DID

## Requirements 203: DID – PSL\_RES Response

Initiator	Target
14.7.2.1	14.7.2.2
<p>The Initiator MUST accept a PSL_RES Response containing the same DID as sent in the PSL_REQ Command.</p> <p>The Initiator MUST treat a PSL_RES Response containing a different DID as Protocol Error.</p>	<p>The Target MUST return the same DID as received in the PSL_REQ Command.</p>

## 14.8 Data Exchange Protocol Request (DEP\_REQ)

The Data Exchange Protocol Request Command (DEP\_REQ) is used by the Initiator to exchange data with a Target that is configured for the NFC-DEP Protocol. The Target responds to the DEP\_REQ with the Data Exchange Protocol Request Response (DEP\_RES).

## 14.8.1 DEP\_REQ Command

Table 103 defines the format of the DEP\_REQ Command.

**Table 103: Format of the DEP\_REQ Command**

Data Exchange Protocol Header					Transport Data Bytes	
Byte 1	Byte 2	Byte 3	Byte 4 (optional)	Byte 5 (optional)	Byte 6 – 6+k-1 (optional)	
D4h	06h	PFB	[DID]	[NAD]	[Data Byte 1]	... [Data Byte k]

Parameter k denotes the variable number of transport data bytes. If optional bytes are not present in the DEP\_REQ Command, then the position of the subsequent bytes is adapted accordingly (e.g., if the DID field and NAD field are not present in the DEP\_REQ Command, then the first transport data byte is Byte 4 instead of Byte 6).

The format of the PFB is specified in Section 14.8.3.

**Requirements 204: DID – DEP\_REQ Command**

Initiator	Target
<p>14.8.1.1 If the DID sent in the preceding ATR_REQ is different from 0, the Initiator <b>MUST</b> include the same DID in the DEP_REQ Command.</p> <p>If the DID sent in the preceding ATR_REQ is equal to 0, the Initiator <b>MUST</b> include no DID in the DEP_REQ Command.</p>	<p>14.8.1.2 If the DID received in the preceding ATR_REQ is different from 0, then:</p> <ul style="list-style-type: none"> <li>• The Target <b>MUST</b> accept a DEP_REQ Command with the same DID.</li> <li>• The Target <b>MUST</b> ignore a DEP_REQ Command with a different DID.</li> </ul> <p>If the DID received in the preceding ATR_REQ is 0, then:</p> <ul style="list-style-type: none"> <li>• The Target <b>MUST</b> accept a DEP_REQ Command without DID.</li> <li>• The Target <b>MUST</b> ignore a DEP_REQ Command with DID.</li> </ul>

## 14.8.2 DEP\_RES Response

Table 104 defines the format of the DEP\_RES Response.

**Table 104: Format of the DEP\_RES Response**

Data Exchange Protocol Header					Transport Data Bytes	
Byte 1	Byte 2	Byte 3	Byte 4 (optional)	Byte 5 (optional)	Byte 6 – 6+k-1 (optional)	
D5h	07h	PFB	[DID]	[NAD]	[Data Byte 1]	... [Data Byte k]

Parameter k denotes the variable number of transport data bytes. If optional bytes are not present in the DEP\_RES Response, then the position of the subsequent bytes is adapted accordingly (e.g., if the DID field and NAD field are not present in the DEP\_RES Response, then the first transport data byte is Byte 4 instead of Byte 6).

The format of the PFB is specified in Section 14.8.3.

**Requirements 205: DID – DEP\_RES Response**

Initiator	Target
<p>14.8.2.1 If the Initiator sent a DID in the DEP_REQ, the Initiator MUST accept a DEP_RES Response with the same DID. The Initiator MUST treat a DEP_RES Response containing a different DID as a Protocol Error. If the Initiator did not send a DID in the DEP_REQ, a DEP_RES Response containing a DID MUST be treated as a Protocol Error.</p>	<p>14.8.2.2 If the DEP_REQ contained a DID, the Target MUST return the same DID in the DEP_RES Response. Otherwise, the Target MUST not return a DID in the DEP_RES Response.</p>

**14.8.3 Protocol Format Byte (PFB)**

The mandatory PFB is used to convey the information required to control data transmission. The protocol defines three fundamental types of blocks, called the Protocol Data Units (PDU):

- Information PDU is used to convey Application Layer Data in the transport data bytes. Application Layer Data is information for use by the adjacent upper layer.
- ACK/NACK PDU is used to convey positive or negative acknowledgements. An ACK/NACK PDU never contains transport data bytes. The acknowledgement relates to the last received PDU.
- Supervisory PDU is used to exchange control information between the Initiator and the Target. Two different types of Supervisory PDUs are defined:
  - Timeout extensions containing one transport data byte and
  - ATN PDU containing no transport data bytes.

The format of the PFB depends on its type. The format of Information PDU, ACK/NACK PDU, and Supervisory PDU is shown in Table 105, Table 106, and Table 107, respectively.

**Table 105: PFB Coding of Information PDU**

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	0	0						Indicates Information PDU
			x					MI: Chaining, if bit is set to 1b
				x				NAD following, if bit is set to 1b
					x			DID following, if bit is set to 1b
						x	x	PNI

The More Information (MI) bit indicates chaining, i.e., whether the current PDU contains only a part of a larger block of Application Layer Data that is split up into several PDUs. Refer to Section 14.12.2 for more details regarding chaining.

The Packet Number Information (PNI) bit counts the number of packets sent by the Initiator to the Target, and vice versa, starting with 0. These bits are used for error detection during protocol handling.

**Table 106: PFB Coding of ACK/NACK PDU**

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	1	0						Indicates ACK/NACK PDU
			x					0b: ACK (Acknowledged) 1b: NACK (Not acknowledged)
				x				NAD following, if bit is set to 1b
					x			DID following, if bit is set to 1b
						x	x	PNI

**Table 107: PFB Coding of Supervisory PDU**

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1	0	0						Indicates Supervisory PDU
			x					0b: ATN (Attention) 1b: Timeout
				x				NAD following, if bit is set to 1b
					x			DID following, if bit is set to 1b
						0	0	RFU

#### 14.8.4 Response Timeout Extension

When the Target needs more time than the defined RWT to process the received PDU, it sends a Response Timeout Extension (RTOX) Request within a Supervisory PDU to the Initiator. An RTOX Request contains one transport data byte, as specified in Table 108.



**Table 108: Coding of Transport Data Byte of an RTOX Request**

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	0							RFU
		x	x	x	x	x	x	RTOX

The bits b6 to b1 code the RTOX, which is coded in the range from 1 to 59.

The Initiator acknowledges the receipt of an RTOX Request by sending an RTOX Response containing one transport data byte. The transport data byte contains the same RTOX as received in the RTOX Request. Refer to Table 109 for the format of the RTOX Response.

**Table 109: Format of Transport Data Byte of an RTOX Response**

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	0							RFU
		x	x	x	x	x	x	RTOX

The corresponding intermediate value of RWT is calculated by the following formula:

$$\mathbf{RWT}_{\text{INT}} = \text{RWT} \times \text{RTOX}$$

The time  $\mathbf{RWT}_{\text{INT}}$  requested by the Target starts after the Initiator has sent the RTOX Response.

## Requirements 206: Response Timeout Extension

Initiator	Target
14.8.4.1	14.8.4.2
<p>The Initiator MUST be ready to receive an RTOX Request with an RTOX containing a value in the range from 1 to 59.</p> <p><i>The Initiator MAY resort to exception processing (Protocol Error) on reception of an RTOX Request having an RTOX set to 0 and 60 to 63.</i></p>	<p>The Target MUST code RTOX in the range from 1 to 59.</p>
14.8.4.3	14.8.4.4
<p>The Initiator MUST support a response waiting time extension less than or equal to the maximum supported RWT:</p> $RWT_{INT} \leq (256 \times 16/f_C) \times 2^{WT_{MAX}}$ <p>Refer to Appendix A.10 for the value of <math>WT_{MAX}</math>.</p>	<p>The Target MUST code RTOX such that <math>RWT_{INT}</math> is less than or equal to the maximum value of RWT:</p> $RWT_{INT} \leq (256 \times 16/f_C) \times 2^{WT_{MAX}}$ <p>Refer to Appendix A.10 for the value of <math>WT_{MAX}</math>.</p>
14.8.4.5	14.8.4.6
<p>The Initiator MUST code RTOX in the RTOX Response with the same value as the value of RTOX in the RTOX Request.</p>	<p>The Target MUST treat an RTOX in the RTOX Response with a value different from the value of RTOX in the RTOX Request as a Protocol Error.</p>
14.8.4.7	14.8.4.8
<p>After sending the RTOX Response PDU in response to an RTOX Request PDU from the Target, the Initiator MUST wait for <math>RWT_{INT} + \Delta RWT</math> for the SoD of a PDU of the Target.</p> <p>If the Initiator does not receive the SoD of a PDU from the Target within <math>RWT_{INT} + \Delta RWT + \Delta T_{INITIATOR}</math>, then the Initiator MUST treat this as a Timeout Error.</p> <p>Refer to Appendix A.10 for the values of <math>\Delta RWT</math> and <math>\Delta T_{INITIATOR}</math>.</p> <p><i>Between <math>RWT_{INT} + \Delta RWT</math> and <math>RWT_{INT} + \Delta RWT + \Delta T_{INITIATOR}</math>, the Initiator MAY accept the PDU or MAY generate a Timeout Error.</i></p>	<p>After receiving the RTOX Response PDU of the Initiator, the Target MUST send the SoD of the next PDU within <math>RWT_{INT}</math>.</p>

## 14.9 Deselect Request (DSL\_REQ)

The Deselect Request Command (DSL\_REQ) is used by the Initiator to deactivate a Target. The Target responds to the DSL\_REQ with the Deselect Response (DSL\_RES).

### 14.9.1 DSL\_REQ Command

Table 110 defines the format of the DSL\_REQ Command.

**Table 110: Format of the DSL\_REQ Command**

Byte 1	Byte 2	Byte 3 (optional)
D4h	08h	[DID]

**Requirements 207: DID – DSL\_REQ Command**

Initiator	Target
<p>14.9.1.1 If the DID sent in the preceding ATR_REQ is different from 0, the Initiator <b>MUST</b> include the same DID in the DSL_REQ Command.</p> <p>If the DID sent in the preceding ATR_REQ is equal to 0, the Initiator <b>MUST</b> not include a DID in the DSL_REQ Command.</p>	<p>14.9.1.2 If the DID received in the preceding ATR_REQ is different from 0, then:</p> <ul style="list-style-type: none"> <li>• The Target <b>MUST</b> accept a DSL_REQ Command with the same DID.</li> <li>• The Target <b>MUST</b> ignore a DSL_REQ Command with a different DID.</li> </ul> <p>If the DID received in the preceding ATR_REQ is 0, then:</p> <ul style="list-style-type: none"> <li>• The Target <b>MUST</b> accept a DSL_REQ Command without DID.</li> <li>• The Target <b>MUST</b> ignore a DSL_REQ Command with DID.</li> </ul>

### 14.9.2 DSL\_RES Response

Table 111 defines the format of the DSL\_RES Response.

**Table 111: Format of the DSL\_RES Response**

Byte 1	Byte 2	Byte 3 (optional)
D5h	09h	[DID]

**Requirements 208: DID – DSL\_RES Response**

<b>Initiator</b>	<b>Target</b>
<p>14.9.2.1 If the Initiator sent a DID in the DSL_REQ, the Initiator MUST accept a DSL_RES Response containing the same DID.</p> <p>The Initiator MUST treat a DSL_RES Response containing a different DID as a Protocol Error.</p> <p>If the Initiator did not send a DID in the DSL_REQ, a DSL_RES Response containing a DID MUST be treated as Protocol Error.</p>	<p>14.9.2.2 If the DSL_REQ contained a DID, the Target MUST return the same DID in the DSL_RES Response.</p> <p>Otherwise, the Target MUST not return a DID in the DSL_RES Response.</p>

**14.10 Release Request (RLS\_REQ)**

The Initiator uses the Release Request Command (RLS\_REQ) to release a Target, which responds to the RLS\_REQ with the Release Response (RLS\_RES).

**14.10.1 RLS\_REQ Command**

Table 112 defines the format of the RLS\_REQ Command.

**Table 112: Format of the RLS\_REQ Command**

<b>Byte 1</b>	<b>Byte 2</b>	<b>Byte 3 (optional)</b>
D4h	0Ah	[DID]

**Requirements 209: RLS\_REQ Command**

<b>Initiator</b>	<b>Target</b>
<p>14.10.1.1 If the DID sent in the preceding ATR_REQ is different from 0, the Initiator <b>MUST</b> include the same DID in the RLS_REQ Command.</p> <p>If the DID sent in the preceding ATR_REQ is equal to 0, the Initiator <b>MUST</b> not include a DID in the RLS_REQ Command.</p>	<p>14.10.1.2 If the DID received in the preceding ATR_REQ is different from 0, then:</p> <ul style="list-style-type: none"> <li>• The Target <b>MUST</b> accept a RLS_REQ Command with the same DID.</li> <li>• The Target <b>MUST</b> ignore a RLS_REQ Command with a different DID.</li> </ul> <p>If the DID received in the preceding ATR_REQ is 0, then:</p> <ul style="list-style-type: none"> <li>• The Target <b>MUST</b> accept a RLS_REQ Command without DID.</li> <li>• The Target <b>MUST</b> ignore a RLS_REQ Command with DID.</li> </ul>

**14.10.2 RLS\_RES Response**

Table 113 defines the format of the RLS\_RES Response.

**Table 113: Format of the RLS\_RES Response**

<b>Byte 1</b>	<b>Byte 2</b>	<b>Byte 3 (optional)</b>
D5h	0Bh	[DID]

**Requirements 210: RLS\_RES Response**

<b>Initiator</b>	<b>Target</b>
<p>14.10.2.1 If the Initiator sent a DID in the RLS_REQ, the Initiator <b>MUST</b> accept a RLS_RES Response containing the same DID.</p> <p>The Initiator <b>MUST</b> treat an RLS_RES Response containing a different DID byte as a Protocol Error.</p> <p>If the Initiator did not send a DID in the RLS_REQ, an RLS_RES Response containing a DID <b>MUST</b> be treated as a Protocol Error.</p>	<p>14.10.2.2 If the RLS_REQ contained a DID, the Target <b>MUST</b> return the same DID in the RLS_RES.</p> <p>Otherwise, the Target <b>MUST</b> not return a DID in the RLS_RES Response.</p>

## 14.11 Timing Requirements

NFC Forum Devices use NFC-A frame timing or NFC-F frame timing for the NFC-DEP Protocol, depending on the configuration.

The NFC-DEP protocol uses the Response Waiting Time (RWT) for the timing requirements.

### Requirements 211: Frame Timing – NFC-DEP Protocol

Initiator and Target	
14.11.1.1	<p>The minimum Frame Delay Time Poll→Listen for Commands and Responses specified in Section 14 (NFC-DEP Protocol) MUST be set to:</p> <ul style="list-style-type: none"> <li>• The minimum Frame Delay Time Poll→Listen, as defined in Section 4.10.1, if the NFC Forum Device is using NFC-A Technology, or</li> <li>• The minimum Frame Delay Time Poll→Listen, as defined in Section 6.7.1, if the NFC Forum Device is using NFC-F Technology.</li> </ul>
14.11.1.2	<p>The minimum Frame Delay Time Listen→Poll for Commands and Responses specified in Section 14 (NFC-DEP Protocol) MUST be set to:</p> <ul style="list-style-type: none"> <li>• The minimum Frame Delay Time Listen→Poll, as defined in Section 4.10.2, if the NFC Forum Device is using NFC-A Technology, or</li> <li>• The minimum Frame Delay Time Listen→Poll, as defined in Section 6.7.2, if the NFC Forum Device is using NFC-F Technology.</li> </ul>

The Response Waiting Time (RWT) defines the time within which a Target has to send the SoD of its Response after the end of a Poll Frame and is calculated by the following formula:

$$RWT = (256 \times 16/f_c) \times 2^{WT}$$

where the value of WT has the range from 0 to 14. WT is included in the ATR\_RES Response as specified in Section 14.6.3.

For the Device Activation Activity and the Device Deactivation Activity, the following specific RWTs are defined:

**RWT<sub>ACTIVATION</sub>**

A specific RWT is defined for the ATR\_REQ Command. For this Command, the Target starts sending its Response frame within **RWT<sub>ACTIVATION</sub>** (response waiting time). Refer to Section 14.6.2 for the definition of the ATR\_REQ Command.

NOTE The response waiting time **RWT<sub>ACTIVATION</sub>** is introduced and defined in this document.

**Requirements 212: Response Waiting Time**

Initiator	Target
<p>14.11.1.3 Following the ATR_REQ Command, the Initiator MUST wait at least <b>RWT<sub>ACTIVATION</sub>+ΔRWT</b> for the SoD of the ATR_RES Response. If the Initiator does not receive the SoD of the ATR_RES Response within <b>RWT<sub>ACTIVATION</sub> + ΔRWT + ΔT<sub>INITIATOR</sub></b>, then the Initiator MUST treat this as a Timeout Error. Refer to Appendix A.10 for the value of <b>ΔRWT</b> and <b>ΔT<sub>INITIATOR</sub></b>.</p> <p><i>Between <b>RWT<sub>ACTIVATION</sub>+ΔRWT</b> and <b>RWT<sub>ACTIVATION</sub>+ΔRWT+ΔT<sub>INITIATOR</sub></b>, the Initiator MAY accept the ATR_RES or MAY generate a Timeout Error.</i></p>	<p>14.11.1.4 Following the EoD of the ATR_REQ Command, the Target MUST wait no longer than <b>RWT<sub>ACTIVATION</sub></b> before sending the SoD of the ATR_RES. Refer to Appendix A.10 for the value of <b>RWT<sub>ACTIVATION</sub></b>.</p>
<p>14.11.1.5 Except for the ATR_REQ Command, the Initiator MUST wait at least <b>RWT+ΔRWT</b> for the SoD of the Response from the Target. If the Initiator does not receive the SoD of the Response from the Target within <b>RWT+ΔRWT+ΔT<sub>INITIATOR</sub></b>, then the Initiator MUST treat this as a Timeout Error.</p> <p><i>Between <b>RWT+ΔRWT</b> and <b>RWT+ΔRWT+ΔT<sub>INITIATOR</sub></b>, the Initiator MAY accept the Response or MAY generate a Timeout Error.</i></p>	<p>14.11.1.6 Except for the ATR_REQ Command following the EoD of a Command, the Target MUST wait no longer than RWT before sending the SoD of the Response.</p>

For NFC-F frame timing, the **FDT<sub>F,LISTEN,MAX</sub>** defined in Section 6.7 equals **RWT** - 16 bd (where 16 bd is the bit duration of the SoF). Similar for the ATR\_REQ, the **FDT<sub>F,LISTEN,MAX,ATR</sub>** equals **RWT<sub>ACTIVATION</sub>** - 16 bd.

## 14.12 NFC-DEP Protocol Operation

### 14.12.1 General Rules

General rules for the NFC-DEP Protocol are described in Section 7.

### 14.12.2 Chaining

The chaining feature allows the Initiator and the Target to transmit information that does not fit in a single PDU by dividing the information into several PDUs.

#### Requirements 213: Chaining Rules

Initiator	Target
14.12.2.1 The MI bit in the PFB of an Information PDU controls the chaining of PDUs. When an Information PDU indicating chaining is received, the PDU MUST be acknowledged by an ACK PDU.	14.12.2.2 When an Information PDU indicating chaining is received, the PDU MUST be acknowledged by an ACK PDU or an RTOX Request.

#### Requirements 214: Block Sizes during Chaining

Initiator
14.12.2.3 When the Initiator sends a chain of Information PDUs, the number of payload bytes of each PDU MUST be less than or equal to the maximum payload size defined by either the FSL byte in PSL_REQ or, if PSL_REQ has not been used, by the LR fields in ATR_REQ/RES.



### 14.12.3 PDU Numbering Rules

This section specifies the rules for PDU numbering in a PDU of the Initiator and the Target.

#### Requirements 215: PDU Numbering Rules

Initiator	Target
14.12.3.1 The PNI of the Initiator MUST be initialized to 00b for the current activated Target.	14.12.3.2 The PNI of the Target MUST be initialized to 00b.
14.12.3.3 When a Valid Information PDU or a Valid ACK PDU with a PNI equal to the current PNI is received, the Initiator MUST increment the current PNI for the current Target before optionally sending a new PDU.	14.12.3.4 When a Valid Information PDU or a Valid ACK PDU is received with a PNI equal to the current PNI of the Target, the Target MUST send its Response with this PNI and increment the current PNI after. When, having responded to a Valid ATN PDU directly before, a Valid Information PDU or a Valid ACK PDU is received with a PNI equal to the current PNI of the Target minus one, the Target MUST send its Response with the current PNI minus one and MUST leave the current PNI unchanged afterwards.

### 14.12.4 PDU Handling Rules

This section specifies the PDU handling rules for the Initiator and the Target.

#### Requirements 216: General PDU Handling Rules

Initiator and Target	
14.12.4.1	The first PDU MUST be sent by the Initiator.
14.12.4.2	Supervisory PDUs are only used in pairs. A Supervisory request MUST always be followed by a Supervisory Response.

#### Requirements 217: PDU Handling Rules for the Initiator

Initiator	
14.12.4.3	When an ACK PDU is received, if its PNI is equal to the Initiator's current PNI and the last Information PDU sent by the Initiator indicated chaining, then chaining MUST be continued. If the last Information PDU sent by the Initiator did not indicate chaining, then the Initiator MUST treat the ACK PDU as a Protocol Error.
14.12.4.4	<p>If an RTOX Request is received in response to an Information PDU or an ACK PDU, the Initiator MUST send an RTOX Response except for the case where the Initiator does not want to accept another RTOX Request anymore.</p> <p>The Initiator MUST respond to the first RTOX Request that it receives after sending an Information PDU or ACK/NACK PDU.</p> <p>If an RTOX Request is received in response to a NACK PDU or an ATN PDU then described above it MUST be treated as a Protocol Error.</p> <p><i>The Initiator MAY stop accepting subsequent RTOX Requests preventing the Initiator to be blocked by arbitrary long Target processing times. Nevertheless the Initiator SHOULD accept a number of RTOX Requests to allow the normal operation to finish on the Target side.</i></p>
14.12.4.5	<p>When a NACK PDU is received by the Initiator, it MUST be treated as a Protocol Error.</p> <p><i>When an ACK PDU is received with a PNI not equal to the Initiator's current PNI, then the Initiator MAY treat this as a Protocol Error or the Initiator MAY re-transmit the last Information PDU.</i></p>

#### Requirements 218: PDU Handling Rules for the Target

Target	
14.12.4.6	When an Information PDU not indicating chaining is received, the PDU MUST be acknowledged by an Information PDU or an RTOX Request.
14.12.4.7	When an ACK PDU is received, the PDU MUST be acknowledged by an RTOX Request or chaining MUST be continued.
14.12.4.8	When an ATN PDU is received, the PDU MUST be acknowledged by an identical ATN PDU.

### 14.12.5 Exception Processing

When errors are detected, the following error handling is attempted.

#### Requirements 219: Exception Processing – Target

---

##### Target

---

14.12.5.1 The Target **MUST** detect Transmission Errors and Protocol Errors.

---

14.12.5.2 The Target **MUST NOT** attempt any error recovery. The Target **MUST** always stay in receive mode when a Transmission Error or a Protocol Error occurs.

*Note that a NACK PDU is never sent by the Target.*

---

## Requirements 220: Exception Processing – Initiator

Initiator	
14.12.5.3	When entering the Initiator role, the NFC Forum Device MUST allocate an error counter, reset to zero.
14.12.5.4	<p>If a PDU with a Transmission Error is received, then the Initiator MUST send a NACK PDU if all of the following are true:</p> <ul style="list-style-type: none"> <li>• The CRC is incorrect or the frame has a parity error.</li> <li>• The PDU is encapsulated in a frame of at least 4 characters long.</li> </ul> <p>The Initiator MUST ignore all other Transmission Errors and MUST be ready to process a Valid PDU within a time <math>t_{RECOVERY}</math> after the end of the PDU with the Transmission Error.</p> <p>The Initiator MUST send the NACK PDU within a time <math>t_{RETRANSMISSION}</math> measured from the start of the PDU with the Transmission Error.</p> <p>If a Transmission Error is received after sending a NACK PDU the Initiator MUST send another NACK PDU until <math>n_{RETRY,NACK}</math> consecutive NACK PDUs have been sent. If no Valid PDU to all NACK PDUs is received, then the Initiator MUST RAISE THE Transmission Error PROTOCOL EXCEPTION.</p> <p>Refer to Appendix A.10 for the values of <math>t_{RECOVERY}</math>, <math>t_{RETRANSMISSION}</math> and <math>n_{RETRY,NACK}</math>.</p>
14.12.5.5	If a PDU with a Protocol Error is received, then the Initiator MUST RAISE THE Protocol Error PROTOCOL EXCEPTION.
14.12.5.6	<p>If a Timeout Error occurs, the NFC Forum Device MUST continue as follows.</p> <p>If the error counter equals <math>n_{TO,MAX}</math>, then the Initiator MUST RAISE THE Timeout Error PROTOCOL EXCEPTION and MUST reset the error counter.</p> <p>Otherwise, the Initiator MUST increment the error counter and proceed as follows:</p> <ul style="list-style-type: none"> <li>• If the last PDU sent before the Timeout Error occurred was a NACK PDU, then the NFC Forum Device MUST send up to <math>n_{RETRY,NACK}</math> NACK PDUs to ask for retransmission. The Initiator MUST send a NACK PDU before <math>t_{RETRANSMISSION}</math>.</li> <li>• If a Valid PDU is not received by any of the <math>n_{RETRY,NACK}</math> NACK PDUs, then the Initiator MUST RAISE THE Timeout Error PROTOCOL EXCEPTION.</li> <li>• Otherwise, the NFC Forum Device MUST reset the error counter and continue processing at the point before the Timeout Error occurred.</li> <li>• Otherwise, the NFC Forum Device MUST send up to <math>n_{RETRY,ATN}</math> ATN PDUs. The Initiator MUST send an ATN PDU before <math>t_{RETRANSMISSION}</math>.</li> <li>• If a Valid PDU is not received by any of the <math>n_{RETRY,ATN}</math> ATN PDUs, then the Initiator MUST RAISE THE Timeout Error PROTOCOL EXCEPTION.</li> <li>• Otherwise, the NFC Forum Device MUST continue processing at the point before the Timeout Error occurred.</li> <li>• Refer to Appendix A.10 for the values of <math>n_{TO,MAX}</math>, <math>t_{RETRANSMISSION}</math>, and <math>n_{RETRY,ATN}</math>.</li> </ul>

## A. Values

Throughout the document symbols are used to identify the values of parameters. The actual values of the parameters are listed in this Appendix. For some of the parameters, a minimum and maximum value is defined. Other parameters are defined by a single value.

Parameters have a value for the NFC Forum Device in Poll Mode and for the NFC Forum Device in Listen Mode. Unless otherwise specified, the value for Poll Mode has to be used when the parameter is referenced in a Poll Mode requirement. The value for Listen Mode has to be used when referenced in a Listen Mode requirement.

### Requirements 221: Values

Poll Mode		Listen Mode	
A.0.1	The actual Poll Mode value of a specific parameter MUST be set according to the Poll Mode value of the equally named parameter specified in this appendix.	A.0.2	The actual Listen Mode value of a specific parameter MUST be set according to the Listen Mode value of the equally named parameter specified in this appendix.

## A.1 NFC-A Technology

Table 114: NFC-A Technology Poll Mode and Listen Mode Parameter Values

Parameter	Poll Mode Value		Listen Mode Value		Units
	Min	Max	Min	Max	
$FDT_{A,POLL,MIN}$	6780		1172		$1/f_C$
$FDT_{A,REACTIVATION}$	5.1		5.0		ms
$GT_A$			5.0		ms
$t_{nn}$			1408		$1/f_C$

## A.2 NFC-B Technology

Table 115: NFC-B Technology Poll Mode and Listen Mode Parameter Values

Parameter	Poll Mode Value		Listen Mode Value		Units
	Min	Max	Min	Max	
$EGT_{\text{POLL}}$	0	752	0	768	$1/f_c$
$EGT_{\text{LISTEN}}$	0	272	0	256	$1/f_c$
$FWT_{\text{SENSB}}$	7680		4096		$1/f_c$
$FWI_{\text{MAX}}$	14		8		-
$\Delta FWT$	16				$1/f_c$
$\Delta T_{\text{POLL}}$	20				ms
$SFGI_{\text{MAX}}$	14		8		-
$\Delta SFGT$		$384 \times 2^{\text{SFGI}}$			$1/f_c$
$FDT_{\text{B,REACTIVATION}}$	5.1		5.0		ms
$GT_{\text{B}}$			5.0		ms
$t_{\text{nn}}$			1408		$1/f_c$
$TR0_{\text{MIN}}$	1008		1024		$1/f_c$
$TR1_{\text{MIN}}$	1254		1280		$1/f_c$
$TR1_{\text{MAX}}$	3216		3200		$1/f_c$
$TR2_{\text{MIN,DEFAULT}}$	6780				$1/f_c$
$t_{\text{LISTEN,S,1}}$	1264	1424	1272	1416	$1/f_c$
$t_{\text{LISTEN,S,2}}$	240	400	248	392	$1/f_c$
$t_{\text{FSOFF}}$	0	272	0	256	$1/f_c$
$t_{\text{POLL,S,1}}$	1280	1416	1272	1424	$1/f_c$
$t_{\text{POLL,S,2}}$	248	392	240	400	$1/f_c$
$t_{\text{POLL,E}}$	1280	1416	1272	1424	$1/f_c$
$t_{\text{LISTEN,E}}$	1264	1424	1272	1416	$1/f_c$
$FSC_{\text{MIN}}$	16		16		-

### A.3 NFC-F Technology

Table 116: NFC-F Technology Poll Mode and Listen Mode Parameter Values

Parameter	Poll Mode Value		Listen Mode Value		Units
	Min	Max	Min	Max	
$TR_{F,LISTEN,MIN}$	42 x 64 – 16		42 x 64		$1/f_C$
$TR_{F,POLL,MIN}$	106 x 64 + 16		106 x 64		$1/f_C$
$FDT_{F,REACTIVATION}$	20.4		20.0		ms
$GT_F$			20.0		ms
$\Delta T_{POLL}$	1				ms
$T_{TIMESLOT}$	256 x 64		256 x 64		$1/f_C$
$T_{DELAY}$	512 x 64		512 x 64		$1/f_C$

### A.4 Type 1 Tag Platform

Table 117: Type 1 Tag Platform Poll Mode and Listen Mode Parameter Values

Parameter	Poll Mode Value		Listen Mode Value		Units
	Min	Max	Min	Max	
$RRDD_{MIN,BIT0}$	320				$1/f_C$
$RRDD_{MIN,BIT1}$	384				$1/f_C$

### A.5 Type 2 Tag Platform

Table 118: Type 2 Tag Platform Poll Mode and Listen Mode Parameter Values

Parameter	Poll Mode Value		Listen Mode Value		Units
	Min	Max	Min	Max	
$FDT_{T2T,READ,MAX}$	5		4.75		ms
$FDT_{T2T,WRITE,MAX}$	10		9.5		ms
$FDT_{T2T,SL,MAX}$	1		0.95		ms
$\Delta T_{POLL}$	20				ms
$PAT_{T2T,SL,MAX}$	1.00		0.95		ms

## A.6 Type 3 Tag Platform

Table 119: Type 3 Tag Platform Poll Mode and Listen Mode Parameter Values

Parameter	Poll Mode Value		Listen Mode Value		Units
	Min	Max	Min	Max	
<b>T</b>				256 x 16	1/f <sub>C</sub>
<b>ΔT<sub>POLL</sub></b>	20				ms
<b>ΔRWT</b>	16				1/f <sub>C</sub>

## A.7 Type 4A Tag Platform

Table 120: Type 4A Tag Platform Poll Mode and Listen Mode Parameter Values

Parameter	Poll Mode Value		Listen Mode Value		Units
	Min	Max	Min	Max	
<b>FSD<sub>MIN</sub></b>	256		256		-
<b>FSC<sub>MIN</sub></b>	16		16		-
<b>FWI<sub>MAX</sub></b>	14		8		-
<b>FWT<sub>ACTIVATION</sub></b>	71680		65536		1/f <sub>C</sub>
<b>ΔFWT</b>	16				1/f <sub>C</sub>
<b>ΔT<sub>POLL</sub></b>	20				ms
<b>ΔSFGT</b>	384 x 2 <sup>SFGI</sup>				1/f <sub>C</sub>
<b>SFGI<sub>MAX</sub></b>	14		8		-

## A.8 Type 4B Tag Platform

Table 121: Type 4B Tag Platform Poll Mode and Listen Mode Parameter Values

Parameter	Poll Mode Value		Listen Mode Value		Units
	Min	Max	Min	Max	
<b>FSD<sub>MIN</sub></b>	256		256		-



## A.9 ISO-DEP Protocol

Table 122: ISO-DEP Protocol Poll Mode and Listen Mode Parameter Values

Parameter	Poll Mode Value		Listen Mode Value		Units
	Min	Max	Min	Max	
$t_{\text{RECOVERY}}$	0	1152			$1/f_{\text{C}}$
$t_{\text{RETRANSMISSION}}$	0	$4096 \times 2^{12}$			$1/f_{\text{C}}$
$\text{FWT}_{\text{DEACTIVATION}}$		71680		65536	$1/f_{\text{C}}$
$n_{\text{RETRY,ACK}}$	2	5			
$n_{\text{RETRY,NAK}}$	2	5			
$n_{\text{RETRY,WTX}}$	2	5			
$n_{\text{RETRY,DESELECT}}$	0	5			

## A.10 NFC-DEP Protocol

Table 123: NFC-DEP Protocol Poll Mode and Listen Mode Parameter Values

Parameter	Initiator Value		Target Value		Units
	Min	Max	Min	Max	
$\text{WT}_{\text{MAX}}$		14		8	-
$\Delta\text{RWT}$		16			$1/f_{\text{C}}$
$\Delta\text{T}_{\text{INITIATOR}}$		100			ms
$\text{RWT}_{\text{ACTIVATION}}$		$4096 \times 2^{12}$		$4096 \times 2^{12}$	$1/f_{\text{C}}$
$t_{\text{RECOVERY}}$	0	1280			$1/f_{\text{C}}$
$t_{\text{RETRANSMISSION}}$	0	$4096 \times 2^{12}$			$1/f_{\text{C}}$
$n_{\text{RETRY,NACK}}$	2	5			
$n_{\text{RETRY,ATN}}$	2	5			
$n_{\text{TO,MAX}}$	2	5			

## B. Revision History

Table 124 outlines the revision history of NFC Digital Protocol.

**Table 124: Revision History**

<b>Document Name</b>	<b>Revision and Release Date</b>	<b>Status</b>	<b>Change Notice</b>	<b>Supersedes</b>
NFC Digital Protocol	Version 1, November 2010	Final		