



	1
	2
	3
	4
	5
	6
	7
LoRaWAN 1.0.4	8
End Device Certification	9
Requirements for All Regions	10
Version 1.0	11
	12



Table of Contents

14	References	8
15	1. Introduction	9
16	1.1. Scope of LoRaWAN Certification	9
17	1.2. LoRaWAN Certification Process	9
18 19	1.3. Changes made in this version for the requirement changes from LoRaWAN S v1.0.2 to LoRaWAN Specification v1.0.4	•
20	2. Functional Test Description for LoRaWAN Certification	11
21	2.1. Activation Pre-test	13
22	2.1.1. DUT Pre-condition Activation	13
23	2.1.1.a. Test Procedure Frame Sequence Chart	14
24	2.2. Over the Air Activation	16
25	2.2.1. Pre-Join Behaviour	16
26	2.2.1.a. For Dynamic Channel (DC) plan devices	16
27	2.2.1.a.i. Test Procedure Frame Sequence Chart	17
28	2.2.1.b. For Fixed Channel (FC) plan devices	21
29	2.2.1.b.i. Test Procedure Frame Sequence Chart	22
30	2.2.2. Join-Accept with DLSettings	26
31	2.2.2.a. Test Procedure Frame Sequence Chart	26
32	2.2.3. Join-Accept with Delay Settings	30
33	2.2.3.a. Test Procedure Frame Sequence Chart	30
34	2.2.4. Join-Accept with CFList	33
35	2.2.4.a. For Dynamic Channel (DC) plan devices	33
36	2.2.4.a.i. Test Procedure Frame Sequence Chart	33
37	2.2.4.b. For Fixed Channel (FC) plan devices	35
38	2.2.4.b.i. Test Procedure Frame Sequence Chart	36
39	2.3. Activation by Personalization	40
40	2.3.1. Dynamic channel plan devices	41
41	2.3.1.a. All regions - Test Procedure Frame Sequence Chart	41
42 43	2.3.1.b. For regions with Dwell Time limitation only - Test Procedure Frame Sec 44	γuence Chart
44	2.3.2. Fixed channel plan devices	46
45	2.3.2.a. All regions - Test Procedure Frame Sequence Chart	46





46 47	2.3.2.b. For regions with Dwell Time limitation only - Test Procedure Frame49	e Sequence Chart
48	2.4. Device Functionality Tests	52
49	2.4.1. Default Setting Tests	52
50	2.4.1.a. Cryptography:	52
51	2.4.1.a.i. AES Encryption	52
52	2.4.1.a.i.1. Test Procedure Frame Sequence Chart	53
53	2.4.1.a.ii. Message Integrity Code	53
54	2.4.1.a.ii.1. Test Procedure Frame Sequence Chart	54
55	2.4.1.b. Downlink Sequence Number	54
56	2.4.1.b.i. Test Procedure Frame Sequence Chart	55
57	2.4.2. Confirmed Frames	55
58	2.4.2.a. Confirmed Uplinks	55
59	2.4.2.a.i. Test Procedure Frame Sequence Chart	55
60	2.4.2.b. Confirmed Downlinks	57
61	2.4.2.b.i. Test Procedure Frame Sequence Chart	57
62	2.5. MAC Command Tests	60
63	2.5.1. DevStatusReq	60
64	2.5.1.a. Test Procedure Frame Sequence Chart	60
65	2.5.2. NewChannelReq	60
66	2.5.2.a. Fixed Channel plan devices	60
67	2.5.2.a.i. Test Procedure Frame Sequence Chart	61
68	2.5.2.b. For Dynamic Channel plan devices only	61
69	2.5.2.b.i. Read-only default channels	61
70	2.5.2.b.ii. Addition of a channel	61
71	2.5.2.b.iii. Removal of a channel	61
72	2.5.2.b.iv. Addition and removal of multiple channels	61
73	2.5.2.b.v. Invalid command processing - Frequency	62
74	2.5.2.b.vi. Invalid command processing – Data Rate Range	62
75	2.5.2.b.vii. Removal of default channels – not allowed	62
76	2.5.2.b.viii. Test Procedure Frame Sequence Chart	62
77	2.5.3. DIChannelReq	67
78	2.5.3.a. Fixed Channel plan devices	67
79	2.5.3.a.i. Test Procedure Frame Sequence Chart	68





80	2.5.3.b. Dynamic Channel plan devices	. 68
81	2.5.3.b.i. Test Procedure Frame Sequence Chart	. 68
82	2.5.4. RXParamSetupReq	. 71
83	2.5.4.a. Test Procedure Frame Sequence Chart	. 71
84	2.5.5. RXTimingSetupReq	. 74
85	2.5.5.a. Test Procedure Frame Sequence Chart	. 74
86	2.5.6. TXParamSetupReq MAC command	. 77
87	2.5.6.a. Test Procedure Frame Sequence Chart	. 77
88 89	2.5.6.b. Test Procedure Frame Sequence Chart- if TXParamSetupReq command applicable for the region tested	
90	2.5.7. LinkCheckReq tests	. 82
91	2.5.7.a. Test Procedure Frame Sequence Chart	. 82
92	2.5.8. LinkADRReq	. 82
93	2.5.8.a. TXPower	. 82
94	2.5.8.a.i. Test Procedure Frame Sequence Chart	. 83
95	2.5.8.b. Uplink Channel Management	. 86
96	2.5.8.b.i. For Dynamic channel plan devices	. 86
97	2.5.8.b.i.1. Unsupported data rates	. 86
98	2.5.8.b.i.1.1. Test Procedure Frame Sequence Chart	. 88
99	2.5.8.b.i.2. ChannelMask functionality & Disable all Channels	. 90
100 101	2.5.8.b.i.2.1. Test Procedure Frame Sequence Chart for Channel Mask and Disable Channels 90	al
102	2.5.8.b.ii. For Fixed channel plan Devices	. 92
103	2.5.8.b.ii.1. 125khz Uplink Channel Management	. 93
104	2.5.8.b.ii.1.1. Valid Command Processing	. 93
105	2.5.8.b.ii.1.1.1. Test Procedure Frame Sequence Chart	. 93
106	2.5.8.b.ii.1.2. Invalid Command Processing	. 95
107	2.5.8.b.ii.1.2.1. Test Procedure Frame Sequence Chart	. 96
108	2.5.8.b.ii.1.3. Enable All-Channels	. 98
109	2.5.8.b.ii.1.3.1. Test Procedure Frame Sequence Chart	. 98
110	2.5.8.b.ii.2. 500 kHz Uplink Channel Management	. 99
111	2.5.8.b.ii.2.1. Valid Command Processing	. 99
112	2.5.8.b.ii.2.1.1. Test Procedure Frame Sequence Chart	100
113	2.5.8.b.ii.2.2. Invalid Command Processing	102





114	2.5.8.b.ii.2.2.1. Test Procedure Frame Sequence Chart	102
115	2.5.8.b.ii.3. Disable all Channels (125kHz and 500kHz channels)	104
116	2.5.8.b.ii.3.1. Test Procedure Frame Sequence Chart	104
117	2.5.8.c. Redundancy	104
118	2.5.8.c.i. Test Procedure Frame Sequence Chart	105
119	2.5.8.d. Data Rate Decay	109
120	2.5.8.d.i. DR Decay test for all devices	109
121	2.5.8.d.i.1. Test Procedure Frame Sequence Chart	110
122 123	2.5.8.d.ii. Additional DR Decay test for only DC plan devices which support the data rates 115	optional
124	2.5.8.d.ii.1. Test Procedure Frame Sequence Chart	116
125	2.5.8.e. Command Block Channel Management	120
126	2.5.8.e.i. Dynamic channel plan devices	120
127	2.5.8.e.i.1. Successful LinkADRReq block	120
128	2.5.8.e.i.1.1. Test Procedure Frame Sequence Chart	121
129	2.5.8.e.i.2. Unsuccessful LinkADRReq block	123
130	2.5.8.e.i.2.1. Test Procedure Frame Sequence Chart	123
131	2.5.8.e.ii. Fixed channel plan devices	124
132	2.5.8.e.ii.1. 125kHz Sub-Band Channel Plan	124
133	2.5.8.e.ii.1.1. Test Procedure Frame Sequence Chart	125
134	2.5.9. DutyCycleReq	127
135	2.5.9.a. Test Procedure Frame Sequence Chart	127
136	2.5.10. DeviceTimeReq	128
137	2.5.10.a. Test Procedure Frame Sequence Chart	129
138	2.5.11. RX Window test	129
139	2.5.11.a. RX1 Receive Window Test	129
140	2.5.11.a.i. Test Procedure Frame Sequence Chart	130
141	2.5.11.b. RX2 Receive Window Test	135
142	2.5.11.b.i. Test Procedure Message Sequence Chart	135
143	2.5.11.c. RX1 and RX2 simultaneous frames	137
144	2.5.11.c.i. Test Procedure Message Sequence Chart	138
145	2.5.11.d. RX Oversized Payload	138
146	2.5.11.d.i. Test Procedure Frame Sequence Chart	138
147	2.5.11.e. Maximum Allowed Payload	141





148	2.5.11.e.i. Max Payload via Echo	142
149	2.5.11.e.i.1. Test Procedure Frame Sequence Chart	142
150	2.5.11.e.ii. Oversized Payload via Echo	145
151	2.5.11.e.ii.1. Test Procedure Frame Sequence Chart	145
152	2.5.12. MAC Command(s) in App-Payload and/or Frame Options	148
153	2.5.12.a. App-Payload Only (FPort = 0)	148
154	2.5.12.a.i. Test Procedure Frame Sequence Chart	149
155	2.5.12.b. Frame Options Only (FPort NOT = 0)	149
156	2.5.12.b.i. Test Procedure Frame Sequence Chart	150
157	2.5.12.c. App-Payload and Frame Options	150
158	2.5.12.c.i. Test Procedure Frame Sequence Chart	150
159	2.5.13. Incorrect MAC Commands	152
160	2.5.13.a. Test Procedure Frame Sequence Chart	152
161	2.5.14. Multiple MAC commands prioritization	154
162	2.5.14.a. Test Procedure Frame Sequence Chart	154
163	3. FPort 224 Deactivation	157
164	3.1. Test Procedure Message Sequence Chart	157
165	4. Test Case Mapping with LoRaWAN Specification [1]	
166	5. Notice of Use and Disclosure	162
167		
168	Figures	
169	Figure 1: Test Harness Architecture	11
170	TIBATE 1. TEST HATTIESS / WOUNTECOURTE	11
-		

171 Revision History

Version	Author	Updates	Date
Version	Nisha Bhaskaran /	Release for Class A devices	October 2020
1.0	Derek Hunt		



173 Glossary

LORaWAN Protocol specification developed and maintained by the LoRa Alliance.

Certification LogoLoRa Alliance defined logo that can be displayed on the Certified product

and any documentation and marketing information about the End-

Device.

Lora Test House Organization and corresponding facility accredited by the Lora Alliance

to perform Certification testing.

End Device Device submitted to a LoRa Test House for Certification.

174

175 Abbreviations & Acronyms

ABP Activation by Personalization

ADR Adaptive Data Rate

AS Application Server

ATH Authorized Test House

AWG Arbitrary Waveform Generator

DR Data Rate

DUT Device Under Test

ERP Equivalent Radiated Power compared to a dipole antenna (expressed

in dBd)

EIRP Equivalent Isotropic Radiated Power: ERP = EIRP – 2.15dB (expressed

in dBi)

ETSI European Telecommunications Standards Institute

FSK Frequency Shift Keying modulation technique.

GW Gateway

LORAWAN Conformance Test Tool

MAC Media Access Control

NS Network Server





OTAA Over-the-Air Activation

TCL Test Control Layer of the Test Harness

TRP Total Radiated Power

Definitions of terms used in this document

XXXX X is a valid number which would vary based on the region being tested

DC Dynamic Channel

FC Fixed Channel

Minimum Data Rate for the region tested, as specified in the Regional

Parameters Specification RP2 1.0.1

Maximum Data Rate using 125 kHz bandwidth for the region tested, as

specified in the Regional Parameters Specification RP2 1.0.1

Nb Number

Repeat until the condition specified within brackets [] is completed

References

[1]	LoRaWAN Specifications L2 1.0.4.
[2]	LoRaWAN Regional Parameters Specification RP2 1.0.1.
[3] LoRaWAN Certification Protocol Specification 1.0.0.	



1. Introduction

178

183

194

203

204

205

206

207208

209210211

213

214

215

- 179 This document specifies the minimum testing requirements for an End-Device to be designated
- 180 "LoRaWAN Certified". LoRaWAN Certification will confirm that the End-Device meets the Functional
- 181 Requirements of the LoRaWAN Specification Version L2 1.0.4 [1] for the corresponding regional
- parameters as defined in the LoRaWAN Regional Parameters Version RP2 1.0.1 [2].

1.1. Scope of LoRaWAN Certification

- 184 The scope of this test specification is limited to validating compliant implementation of the LoRaWAN
- 185 protocol for Class A Devices.
- 186 For LoRaWAN Certification it is not mandatory that the End-Device has all the Regulatory approvals,
- but these will be required before the product can be sold or operated in the respective countries.
- 188 Intended or otherwise, the inevitable variability of performance and quality of the radio
- 189 implementation among End-Devices is too high to allow normalized-, practical evaluation. RF
- 190 performance measurement, whether radiated or conducted, is therefore considered out of scope for
- the tests described herein. The RF performance of the End-Device will be tested solely as part of the
- 192 RF Performance Testing, which will be a separate test conducted at the Authorized Test House to
- 193 evaluate the RF performance of the device as part of LoRaWAN Certification testing.

1.2. LoRaWAN Certification Process

- 195 A party seeking LoRaWAN Certification for their End-Device must be a member of the LoRa Alliance in
- 196 good standing and only Test Houses designated accredited by the LoRa Alliance may perform the tests
- 197 described herein to earn it.
- 198 The Authorized Test Houses must communicate the complete results to the LoRa Alliance. If the device
- has passed all mandatory tests, the LoRa Alliance will issue a certificate for the End-Device with respect
- to a version of this document and in turn corresponding versions of [1] and [2]. Additionally, the
- 201 Alliance will publish both the status and a results summary on its web site along with data for any
- 202 optional features tested.

1.3. Changes made in this version for the requirement changes from LoRaWAN Specifications v1.0.2 to LoRaWAN Specification v1.0.4

- Combined all 5 regional Certification specifications into one.
- Certification Application has been removed. All testing will be done in Application mode. Corresponding updates are made throughout the document to
 - Note down the default datarate of the device and reset to default if modified during the test
 - Set the ADR bit if not already set, before performing LinkADRReq command
- 212 o Turn off Duty cycle
 - MAC commands added: DutyCycleReq and DeviceTimeReq
 - Requirement changes made
 - DeviceStatusReq: SNR Margin has been renamed to RadioStatus



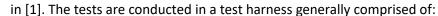


216	 NewChannelReq: For Fixed Channel plan devices, no test must be performed
217	o TXParamSetupReq: This requirement must be tested for regions other than Asian
218	region as well to confirm that the command is ignored
219	 LinkADRReq: The value [0x]F of either DataRate or TXPower means that the DUT mus
220	ignore that field and keep the current parameter values
221	 LinkADRReq - TXPower: When commanded to a valid TX power level lower than it is
222	capable of, the DUT must respond with an unsuccessful LinkADRAns and operates a
223	its previously configured TX power
224	 LinkADRReq – TXPower: When commanded to a valid TX power level greater than it is
225	capable of, the DUT must respond with a successful LinkADRAns and operates at its
226	maximum TX power
227	 New section added: Multiple MAC command prioritization
228	 Section added for Test Case mapping with LW 1.0.4
229	Formatting:
230	 Uplinks and the corresponding downlink are combined the same step to indicate the
231	exact sequence
232	



2. Functional Test Description for LoRaWAN Certification

234 The list of tests specified below reflects the functional requirements of a Class-A End-Device as defined 235



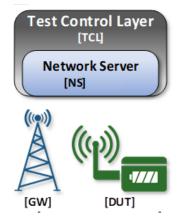


Figure 1: Test Harness Architecture

- A Test Control Layer [TCL]
- A LoRaWAN Network Server [NS]
- 8/16/64 channel LoRaWAN gateway [GW]
- The End-Device Under Test [DUT]

237 Note:

236

238

239

242

243

244 245

233

- A 16-channel gateway device will be used for all official Dynamic Channel Plan device certification
- 240 A 64-channel gateway device will be used for all official Fixed Channel Plan device certification 241
 - An 8-channel gateway device could be used for unofficial pre-testing conducted at LoRaWAN member labs. However, for official certification testing, 8-channel gateway devices will not be used. When using an 8-channel gateway, 125kHz Channels 0-7 and 500kHz Channel 64 must be configured on the gateway.

246

247

248 249

250

251 252

253

- Implementation of this harness architecture is expected to vary among test houses. The Test Control Layer [TCL] is assumed to be a framework of automated scripts and tools that manipulates the Network Server [NS] to facilitate the tests. Specifically, the TCL drives events in the harness, controlling application and network-control content of downlinks. It also decrypts, inspects and validates content of uplinks sent by the **DUT**. This allows test coverage to include:
 - Cryptography
 - Timing of the **DUT** Receive Windows
 - Frequency Channel usage and Data Rate adaptation
- 255 Max Payload handling
- 256 For brevity reasons, this document makes procedural reference to only the TCL, NS, the DUT.





258

259

260

261

265

266

267

269

274

275

276

277

278

279

280281

282

The LoRaWAN gateway [GW] and DUT are collocated in an RF-isolated environment, provisioned as necessary for reliable bi-directional communication. It is nonetheless expected that both the DUT and GW will not receive every frame intended for reception. The TCL should make reasonable effort to accommodate this inevitability. The RF-isolated environment mentioned above should mitigate any potential interference.

- Testing occurs to certify the **DUT** for each supported activation method, be it over-the-air activation (OTAA), activation-by-personalized (ABP), or both.
- The **TCL** must verify the following throughout the course of this certification test suite:
 - The **DUT**'s uplinks' size must respect the maximum allowed uplink size for the data rate used.
 - The length of the **DUT**'s uplinks based on the expected content to ensure no extraneous and unnecessary content is present.
- 268 When the **TCL** is restarted, the **DUT** must be set to the factory reset mode.
- 270 **MIC check Test Note**: When the tool encounters an invalid MIC, it must fail the specific test being performed.
- Downlink FPort Test Note: All MAC commands sent by the TCL will be sent on FPort 0, unless specified
 otherwise in the Sequence charts in this document.

Channel Mask configuration for LinkADRReq MAC command for pre-testing using an 8-channel gateway for Fixed Channel plan devices

For Fixed channel plan devices, when the MAC-CMD LinkADRReq is required to be sent by the TCL, if the device is being pre-tested using an 8-channel gateway, the LinkADRReq in the sequence charts must be replaced by the LinkADRReq commands mentioned below. The LinkADRReq must first disable all 125kHz channels, enable only the channel 64 – 500kHz, and then enable Channels 0-7 using a second LinkADRReq.





283	MAC-CMD LinkADRReq
284	ChMaskCntl = 7
285	ChMask = [0x]0001
286	
287	MAC-CMD LinkADRReq
288	ChMaskCntl = 0
289	ChMask = [0x]00FF
290	
291	Payload = [0x]03XXXXXXXX[0x]03XXXXXXXX
292	
293	If a 64-channel gateway is being used for testing, then the LinkADRReq must be the same as
294	mentioned in the Sequence charts.
295	2.1. Activation Pre-test
296	Section 2.1.1 – DUT Pre-condition Activation tests - must be executed as the first test when
297	executing a single test or multiple tests.
298	
299	Note : The TCL sends frames only on the RX2 window of the previous frame of the DUT for all tests
300	in this document, unless specified otherwise in the Sequence Charts of the Test cases.
301	
302	The DUT must support either over-the-air (OTA) activation or activation by personalization (ABP)
303	or both. If the device supports both OTAA and ABP, the device vendor must provide 2 separate
304	devices, one supporting OTAA and the other supporting ABP method of activation to the
305	Authorised Test House (ATH). The ATH will use the appropriate device for each activation test. In
306	the case where the device supports both OTAA and ABP for the same firmware version, the ATH
307	will run the complete test cycle for the ABP device and then test only the OTAA specific tests for
308	the OTAA device.
309	2.1.1. DUT Pre-condition Activation
310	After initial power-up – and activation if the DUT supports OTAA instead of ABP – the DUT
311	must transmit an uplink packet as soon as possible (recommended within 10 seconds).
312	Contents of this "I'm alive" packet are unimportant.
313	The TCL replies to this packet with a downlink payload frame of [0x]0601
314	(TxPeriodicityChangeReq) sent to port 224, setting the Uplink Periodicity value to 5 seconds.
315	Upon setting the uplink periodicity value, the DUT must try to send an unconfirmed/confirmed
316	uplink every 5 seconds.
317	If the uplink sent by the DUT is a Confirmed frame, the TCL sends a <i>TxFramesCtrlReq</i> command
318	to the DUT to send Unconfirmed frames thereafter.
319	The TCL then checks the ADR Bit setting of the DUT and if disabled, it enables the ADR Bit using
222	the AdvDitChesses IDeas accessed

the $\mbox{\it AdrBitChannelReq}$ command.





321	The TCL then sets the Data Rate to Max125kHzDR, refer [2], using the <i>LinkADRReq</i> command.
322	TCL finally sends the <i>DutVersionsReq</i> command to the DUT to obtain the version of the device.
323	Verify that
324	DUT increments the DevNonce on reset
325	DUT Uplink Periodicity is set to 5 seconds
326	DUT sends Unconfirmed uplink frames
327	DUT enables its ADR Bit
328	DUT sets the Data Rate to Max125kHzDR
329	• DUT sends the version information in the <i>DutVersionsAns</i> response. TCL must store
330	this version number for display in the final Test Report and Certificate.
331	
332	For more details on the Certification Protocol implementation, refer to the Certification
333	Protocol Specification [3].
334	2.1.1.a. Test Procedure Frame Sequence Chart
335	



Step	Procedure		Frame Sequence	Test Purpose	
		End Device - TCL	Frame	-	
1	If DUT = OTA device, DUT sends a Join- Request frame	→	DataRate (DR) = Any allowed DR, refer [2]		
	If DUT = ABP device, skip Step 1.		Note down the DevNonce		
2	TCL sends a Join-Accept frame DUT sends Unconfirmed or Confirmed frame FCntUp = n (where n = 0 or 1 for OTA devices and any random number for ABP devices)	← →	If the DUT is an ABP device and ADR Bit is set, DataRate (DR) = Minimum DR allowed by the DUT, refer [2]		
	The TCL sends Unconfirmed frame	+	If DUT sent Confirmed uplink frame, TCL must Acknowledge CP-CMD DutResetReq FPort = 224 Payload = [0x]01		
3	If DUT = OTA device, DUT sends a Join-Request frame If DUT = ABP device, skip Step 3.	→	DevNonce is greater than DevNonce of previous JR		
	TCL sends Join-Accept response on RX1 window	←			
4	DUT sends Confirmed or Unconfirmed frame FCntUp = m For OTA device, m = 0 or 1 For ABP device, m > n	→	If the DUT is an ABP device, and ADR Bit is set, DataRate (DR) = Minimum DR allowed by the DUT, refer [2]		
	The TCL sends Unconfirmed frame	+	CP-CMD TxPeriodicityChangeReq FPort = 224 Periodicity = 5 sec Payload = [0x]0601 If DUT sent Confirmed uplink frame, TCL must Acknowledge	Uplink Periodicity set	
5	DUT sends Confirmed or Unconfirmed frame FCntUp = m + 1	→	FPort = any allowed port except 224		
	If DUT sent a Confirmed frame, then The TCL sends Unconfirmed frame Else, this step must be skipped	+	CP-CMD TxFramesCtrlReq FPort = 224 Frame type = Unconfirmed Payload = [0x]0701 If DUT sent Confirmed uplink frame, TCL must Acknowledge		
6	DUT sends Unconfirmed frame FCntUp = m + 2	\rightarrow			



	If FCtrl ADR Bit = false, then	+	CP-CMD AdrBitChangeReq-ON	
	The TCL sends Unconfirmed frame		FPort = 224 Payload = [0x]0401	
	Else, this step is skipped		[engelie	
7	DUT sends Unconfirmed frame FCntUp = m + 3	→	FCtrl ADR bit = true	Turn on the ADR bit
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq DataRate = Max125kHzDR, refer [2] Payload = [0x]03XXXXXXXX	
			ChMaskCntl: DC = 0, FC = 6	
			ChMask: DC - Enable only default channels FC = [0x]00FF	
8	DUT sends Unconfirmed frame in 5 seconds FCntUp = m + 4)	MAC-CMD LinkADRAns Payload = [0x]0307	Changed the DR to Max125kHz DR
	The TCL sends Unconfirmed frame	+	CP-CMD DutVersionsReq FPort = 224 Payload = [0x]7F	
9	DUT sends Unconfirmed frame FCntUp = m + 5		CP-CMD DutVersionsAns FPort = 224 Payload = [0x]7FXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	DUT version obtained and stored for future

Note: The FCntUp value can either start form 0 or 1. Some secure-elements implementations verify that the counter value is strictly greater than the previous value before performing the crypto operations. Thus, when resetting to 0 the value of the counter in OTAA mode, and the first time the crypto operations are performed, the counter is incremented. On other implementations, the counter value is initialized to the maximum 32-bit unsigned value [0x]FFFFFFFF which then becomes 0 when incremented by 1 prior to performing the crypto operation.

2.2.Over the Air Activation

336

337

338

339

340

341

342

343

344

345

346347

348

349

350

351

This test must be performed if the **DUT** supports over the air activation. The test verifies the correct functionality of the over-the-air activation. Furthermore, the fields within the Join-Accept frame (DLSettings and RXDelay) used to modify the data rates and receive window timing are tested to verify compliance.

2.2.1. Pre-Join Behaviour

2.2.1.a. For Dynamic Channel (DC) plan devices

The **TCL** commands a re-join, and the **DUT** must respond with a Join-Request frame. The **TCL** ignores the Join-Request frames until the **DUT** responds on all the default channels, refer [2].





352	After this, the TCL responds with a Join-Accept frame. The TCL waits for a maximum of (number
353	of default channels * 3) Join-Request frames before responding with a Join-Accept frame.
354	Verify
355	• DUT sends Join-Request frames until the TCL responds with a Join-Accept frame.
356	 All the default channels must be used at least once in these requests.
357	Record the DataRate of the initial Join Requests.
358	 The duration between the Join-Request frames is greater than the
359	JOIN_ACCEPT_DELAY2 which is 6 seconds.
360	 Check if the Major version in the MAC header is correct and the RFU bits are set to 0
361	The DevNonce value sent by the DUT in the Join-Request must be incremented in each
362	Join-Request.
363	DUT successfully joins the network.
364	
365	
366	JoinNonce check
367	The TCL must trigger a Join-Request twice. The JoinNonce value of the second Join-Accept
368	frame must be the same as the first Join-Accept frame.
369	The TCL must trigger a Join-Request again and send a Join-Accept frame with a different
370	JoinNonce value.
371	
372	Verify
373	DUT accepts the first Join-Accept frame
374	DUT rejects the second Join-Accept frame
375	 DUT resends the Join-Request after rejecting the second Join-Accept frame
376	DUT joins the network after the TCL sends a Join-Accept frame with a different
377	JoinNonce value.
378	
379	2.2.1.a.i. Test Procedure Frame Sequence Chart
380	



Step	Procedure	ı	Frame Sequence	Test
		End Device - TCL	Frame	Purpose
1	DUT sends Unconfirmed frame	→		
	The TCL sends Unconfirmed frame	+	CP-CMD DutJoinReq FPort = 224 Payload = [0x]02	Device Reinitializes [Not Joined]
2	DUT sends a maximum of (3 * number of default channels) Join-Request frames, i.e. until Join-Request frames are sent on all the default channels TCL ignores all the Join-Request frames until all the default channels	→ R [3*NbCh] or [JR on AllCh] R [3*NbCh] or	 All Join-Request frames must be sent on all the default channels, refer [2], at least once. Duration between previous Join-Request and next Join-Request > 6 seconds for all JoinRequest frames Record the DataRate of the Join-Request frames Major version is correct RFU bits = 0 DevNonce is greater than DevNonce of previous JR 	
	are used for the Join-Request frames	[3"NBCrij or [JR on AllCh]		
3	DUT sends Join-Request frame again	→	DataRate (DR) = any allowed DR, refer [2] DevNonce is greater than DevNonce of previous JR	
	TCL sends Join-Accept response on RX1 window	+	·	Join accepted
4	DUT sends Unconfirmed or Confirmed frame FCntUp = n (where n = 0 or 1 for OTA devices)	→		
	The TCL sends Unconfirmed frame	+	CP-CMD TxPeriodicityChangeReq FPort = 224 Periodicity = 5 sec Payload = [0x]0601 If DUT sent Confirmed uplink frame, TCL must Acknowledge	Uplink Periodicity set
5	DUT sends Unconfirmed or Confirmed frame FCntUp = n + 1	→	FPort = any allowed port except 224	Next Uplink sent in 5 seconds



Step	Procedure		Frame Sequence	Test Purpose
		End Device - TCL	Frame	i uipose
	If DUT sent a Confirmed frame, then The TCL sends Unconfirmed frame	+	CP-CMD TxFramesCtrlReq FPort = 224 Frame type = Unconfirmed	
	Else, this step must be skipped		Payload = [0x]0701	
			If DUT sent Confirmed uplink frame, TCL must Acknowledge	
6	DUT sends Unconfirmed frame	\rightarrow		
	FCntUp = n + 2			
	If FCtrl ADR Bit = false, then	+	CP-CMD AdrBitChangeReq-	
	The TCL sends Unconfirmed frame		ON FPort = 224 Payload = [0x]0401	
	Else, this step is skipped			
7	DUT sends Unconfirmed frame FCntUp = n + 3	\rightarrow	FCtrl ADR bit = true	Turn on the ADR bit
	The TCL sends Unconfirmed frame	←	MAC-CMD LinkADRReq DataRate = Max125kHzDR, refer [2] Payload = [0x]03XXXXXXXX	
8	DUT sends Unconfirmed frame FCntUp = n + 4	→	MAC-CMD LinkADRAns Payload = [0x]0307	Changed the DR to Max125kHz DR
9	DUT sends Unconfirmed frame	→		
	The TCL sends Unconfirmed frame	+	CP-CMD DutJoinReq FPort = 224 Payload = [0x]02	Device Reinitializes [Not Joined]
10	DUT sends Join-Request	→	DevNonce is greater than DevNonce of previous JR	
	TCL sends Join-Accept response	+	JoinNonce = a	
11	DUT sends Unconfirmed or Confirmed frame	→		
	The TCL sends Unconfirmed frame	+	CP-CMD DutJoinReq FPort = 224 Payload = [0x]02	Device Reinitializes [Not Joined]
			If DUT sent Confirmed uplink frame, TCL must Acknowledge	
12	DUT sends Join-Request	→	DevNonce is greater than DevNonce of previous JR	
	TCL sends Join-Accept response	←	JoinNonce = a	JoinNonce value must be the same as the previous one



Step	Procedure	ı	Frame Sequence	Test
				Purpose
		End Device - TCL	Frame	
13	DUT sends Join-Request	\rightarrow	DevNonce is greater than	DUT rejects
	·		DevNonce of previous JR	the
				JoinAccept
				response
				and sends
				JoinReq
		,		again
	TCL sends Join-Accept response	←	JoinNonce = b, where b NOT = a	Join accepted
14	DUT sends Unconfirmed or	\rightarrow	- a	accepted
	Confirmed frame			
	FCntUp = n (where n = 0 or 1 for			
	OTA devices)			
	The TCL sends Unconfirmed frame	←	CP-CMD	Uplink
			TxPeriodicityChangeReq	Periodicity
			FPort = 224	set
			Periodicity = 5 sec	
			Payload = [0x]0601	
			If DUT sent Confirmed uplink	
			frame, TCL must	
			Acknowledge	
15	DUT sends Confirmed or	\rightarrow	FPort = any allowed port	
	Unconfirmed frame		except 224	
	FCntUp = n + 1			
	If DUT sent a Confirmed frame, then	+	CP-CMD TxFramesCtrlReq	
	The TCL sends Unconfirmed frame		FPort = 224	
	Elea this step must be skipped		Frame type = Unconfirmed Payload = [0x]0701	
	Else, this step must be skipped		Payloau	
			If DUT sent Confirmed uplink	
			frame, TCL must	
			Acknowledge	
16	DUT sends Unconfirmed frame	\rightarrow		
	FCntUp = n + 2		OD OND A LEVOL	
	If FCtrl ADR Bit = false, then	←	CP-CMD AdrBitChangeReq-	
	The TCL sends Unconfirmed frame		FPort = 224	
	The TOE series officerinified frame		Payload = [0x]0401	
	Else, this step is skipped			
17	DUT sends Unconfirmed frame	→	FCtrl ADR bit = true	Turn on the
	FCntUp = n + 3			ADR bit
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq	
			DataRate = Max125kHzDR,	
			refer [2]	
			Payload = [0x]03XXXXXXXX	
18	DUT sends Unconfirmed frame	\rightarrow	MAC-CMD LinkADRAns	Changed the
	within 5 seconds		Payload = [0x]0307	DR to
	FCntUp = n + 4			Max125kHz
				DR





2.2.1.b. For Fixed Channel (FC) plan devices

After the **TCL** commands a re-join to the DUT, the **DUT** sends a Join-Request frame on 125 kHz and 500 kHz channels using the minimum data rate allowed for these channels respectively (DR-X and DR-Y respectively), refer [2]. This is ignored by the server until a channel change from 125kHz at DR-X to 500kHz at DR-Y or vice versa is recognized by the **TCL**. The last Join-Request message indicating the channel change is processed and the server responds with a Join-Accept message.

389 390 Verify

382

383

384

385 386

387

388

391

392393

394

395396

397

398

399 400

401 402

403

404 405 406

407

408

410

411 412

413 414 415

416 417

418 419

- **DUT** sends Join-Request frames on 125 kHz channels using DR-X and 500 kHz channels using DR-Y. For example: For the US902-928 region, Join-Request must be sent on a random 125 kHz channel at DR0 and a random 500kHz channel at DR4.
- The duration between the Join-Request frames is greater than the JOIN_ACCEPT_DELAY2 which is 6 seconds.
- Check if the Major version in the MAC header is correct and the RFU bits are set to 0
- The DevNonce value sent by the DUT in the Join-Request must be incremented in each Join-Request.
- **DUT** successfully joins the network.

The **TCL** again commands a re-join, the DUT sends Join-Request messages on 125 kHz channels using DR-X and 500 kHz channels using DR-Y. The **TCL** does not respond to requests with the same DR as the one responded in the previous test. It responds only to Join Requests with the alternate DR.

Verify

• **DUT** joins successfully by way of a Join-Request sent at the DR not responded earlier.

409 JoinNonce check

The **TCL** must trigger a Join-Request twice. The JoinNonce value of the second Join-Accept frame must be the same as the first Join-Accept frame.

The **TCL** must trigger a Join-Request again and send a Join-Accept frame with the correct JoinNonce value.

Verify

- DUT accepts the first Join-Accept frame
- **DUT** rejects the second Join-Accept frame
- DUT resends the Join-Request after rejecting the second Join-Accept frame
- **DUT** joins the network after the **TCL** sends a Join-Accept frame with the correct JoinNonce value.

421





2.2.1.b.i.Test Procedure Frame Sequence Chart



Step	Procedure	Frame Sequence		Test
		End Device - TCL	Frame	Purpose
1	DUT sends Unconfirmed frame	→		
	The TCL sends Unconfirmed frame	+	CP-CMD DutJoinReq FPort = 224 Payload = [0x]02	Device Reinitializes [Not Joined]
2	TCL ignores the Join-Request frames and waits for a maximum of (2 * number of channels configured)	→ R [2*NbCh] or [DR-X NOT = DR-Y] R [2*NbCh] or [DR-X NOT	Join-Request @ minimum default data rate for - 125 kHz on Channel A and - 500 kHz channel on Channel B (where A is a random upstream channel utilizing 125 kHz and B is a random upstream channel utilizing 500 kHz), refer [2] i.e. Join-Request1 @ DR-X on Channel A Join-Request2 @ DR-Y on Channel B DevNonce is greater than DevNonce of previous JR	
	uplink transmissions, until a channel change from 125kHz at DR-X to 500kHz at DR-Y or vice versa is recognized by the TCL.	= DR-Y]		
	TCL sends Join-Accept response when it receives a Join-Request with a channel change, on RX1 window.	← [DR-X → DR-Y]	Join-Accept is sent to DUT without a CFList. Pre-testing with 8-channel gateway: CFListType = [0x]01 ChMask0 = [0x]00FF ChMask1 = [0x]0000 ChMask2 = [0x]0000 ChMask3 = [0x]0000 ChMask4 = [0x]0001	Join accepted
3	DUT sends Unconfirmed or Confirmed frame FCntUp = n (where n = 0 or 1 for OTA devices)	→		
	The TCL sends Unconfirmed frame	+	CP-CMD DutJoinReq FPort = 224 Payload = [0x]02	Device Reinitializes [Not Joined]



Step	Procedure	ı	Test Purpose	
		End Device - TCL	Frame	
4	DUT sends Join-Request frames.	→ R [2*NbCh] or [DR-X NOT = DR-Y]	Join-Request @ minimum default data rate for	
	TCL ignores the Join-Request frames and waits for a maximum of (2 * number of channels configured) uplink transmissions, until the DUT sends a Join-Request with a different DataRate from the one sent earlier.	R [2*NbCh] or [JR-DR-X NOT = JR- DR-Y]		
	TCL sends Join-Accept response when it receives the Join-Request with the alternate DR	← [JR-DR-X NOT = JR- DR-Y]	Join-Accept is sent to DUT without a CFList. Pre-testing with 8-channel gateway: CFListType = [0x]01 ChMask0 = [0x]00FF ChMask1 = [0x]0000 ChMask2 = [0x]0000 ChMask3 = [0x]0000 ChMask4 = [0x]0001	Join accepted
5	DUT sends Unconfirmed frame	→	-	
	The TCL sends Unconfirmed frame	+	CP-CMD DutJoinReq FPort = 224 Payload = [0x]02	Device Reinitializes [Not Joined]
6	DUT sends Join-Request	→	DevNonce is greater than DevNonce of previous JR	
	TCL sends Join-Accept response	+	JoinNonce = a	
7	DUT sends Unconfirmed or Confirmed frame	→		



Step	Procedure		Frame Sequence	Test
		End Device - TCL	Frame	Purpose
	The TCL sends Unconfirmed frame	+	CP-CMD DutJoinReq FPort = 224 Payload = [0x]02 If DUT sent Confirmed uplink frame, TCL must Acknowledge	Device Reinitializes [Not Joined]
8	DUT sends Join-Request	\rightarrow	DevNonce is greater than DevNonce of previous JR	
	TCL sends Join-Accept response	+	JoinNonce = a	JoinNonce value must be the same as the previous one
9	DUT sends Join-Request	→	DevNonce is greater than DevNonce of previous JR	DUT rejects the JoinAccept response and sends JoinReq again
	TCL sends Join-Accept response	←	JoinNonce = b, where b NOT = a Official certification (64-channel gateway): Join-Accept is sent to DUT without a CFList Pre-testing with 8-channel gateway: CFListType = [0x]01 ChMask0 = [0x]00FF ChMask1 = [0x]0000 ChMask2 = [0x]0000 ChMask3 = [0x]0000 ChMask4 = [0x]0001	Join accepted
10	DUT sends Unconfirmed or Confirmed frame FCntUp = n (where n = 0 or 1 for OTA devices)	→	La Tana	
	The TCL sends Unconfirmed frame	+	CP-CMD TxPeriodicityChangeReq FPort = 224 Periodicity = 5 sec Payload = [0x]0601 If DUT sent Confirmed uplink frame, TCL must Acknowledge	Uplink Periodicity set
11	DUT sends Confirmed or Unconfirmed frame FCntUp = n + 1	→	FPort = any allowed port except 224	



Step	Procedure		Frame Sequence	
				Purpose
		End Device	Frame	
		- TCL		
	If DUT sent a Confirmed frame, then	←	CP-CMD TxFramesCtrlReq	
	The TCL sends Unconfirmed frame		FPort = 224	
			Frame type = Unconfirmed	
	Else, this step must be skipped		Payload = [0x]0701	
			If DUT sent Confirmed uplink	
			frame, TCL must	
			Acknowledge	
12	DUT sends Unconfirmed frame	\rightarrow		
	FCntUp = n + 2			
	If FCtrl ADR Bit = false, then	←	CP-CMD AdrBitChangeReq-	
			ON	
	The TCL sends Unconfirmed frame		FPort = 224	
			Payload = [0x]0401	
	Else, this step is skipped			
13	DUT sends Unconfirmed frame	\rightarrow	FCtrl ADR bit = true	Turn on the
	FCntUp = n + 3			ADR bit
	The TCL sends Unconfirmed frame	←	MAC-CMD LinkADRReq	
			DataRate = Max125kHzDR,	
			refer [2]	
			Payload = [0x]03XXXXXXXX	
			ChMaskCntl = 6	
			ChMask = [0x]00FF	
14	DUT sends Unconfirmed frame in 5	→	MAC-CMD LinkADRAns	Changed the
	seconds		Payload = [0x]0307	DR to
	FCntUp = n + 4			Max125kHz
				DR

425

426

427

428

429

430

2.2.2. Join-Accept with DLSettings

After the TCL triggers a Join-Request, the DUT starts the Join procedure for over-the-air activation. The TCL responds with a Join-Accept frame with RX1DRoffset = 2 and RX2DataRate = any applicable DataRate, except the default RX2DataRate as defined in [2]. After the join procedure succeeds, the TCL downlinks an echo command targeting the RX1 window to which the DUT must respond correctly. Next the TCL downlinks an echo command targeting the RX2 window to which the **DUT** must respond correctly.

431 432

Verify

434

433 **DUT** successfully joins the network **DUT** implements RX1DRoffset correctly after processing the Join-Accept

435 436 **DUT** implements RX2DataRate correctly after processing the Join-Accept

437

2.2.2.a. Test Procedure Frame Sequence Chart



1		End Device -	Frame	Purpose
1	51.17	TCL		
	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame	+	CP-CMD DutJoinReq FPort = 224 Payload = [0x]02	Device Reinitializes [Not Joined]
2	DUT sends Join-Request	→	DR = Z (where Z = any allowed DR, refer [2]) DevNonce is greater than DevNonce of previous JR	
	TCL sends Join-Accept response	÷	RX1DROffset = 2 RX2DataRate = Any DR except default RX2 DR, as defined in [2] Join-Accept is sent to DUT without a CFList. Pre-testing with 8-channel gateway: CFListType = [0x]01 ChMask0 = [0x]00FF ChMask1 = [0x]0000 ChMask2 = [0x]0000 ChMask3 = [0x]0000 ChMask4 = [0x]0001	Join accepted with modified parameters
3	DUT sends Unconfirmed or Confirmed frame FCntUp = n (where n = 0 or 1 for OTA devices)	→		
	The TCL sends Unconfirmed frame	+	CP-CMD TxPeriodicityChangeReq FPort = 224 Periodicity = 5 sec Payload = [0x]0601 If DUT sent Confirmed uplink frame, TCL must Acknowledge	Uplink Periodicity set
4	DUT sends Unconfirmed or Confirmed frame FCntUp = n + 1	\rightarrow		
-	If DUT sent a Confirmed frame, then The TCL sends Unconfirmed frame Else, this step must be skipped	+	CP-CMD TxFramesCtrlReq FPort = 224 Frame type = Unconfirmed Payload = [0x]0701 If DUT sent Confirmed uplink frame, TCL must Acknowledge	
5	DUT sends Unconfirmed frame	\rightarrow		
	If FCtrl ADR Bit = false, then The TCL sends Unconfirmed frame Else, this step is skipped	←	CP-CMD AdrBitChangeReq-ON FPort = 224 Payload = [0x]0401	



Step	Procedure		Frame Sequence	Test Purpose
		End Device - TCL	Frame	i dipose
6	DUT sends Unconfirmed frame	\rightarrow	FCtrl ADR Bit = true	ADR Bit turned ON
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq DataRate = Max125kHzDR, refer [2] Payload = [0x]03XXXXXXXX	
			ChMaskCntl: DC = 0, FC = 6	
			ChMask: DC - Enable only default channels FC = [0x]00FF	
7	DUT sends Unconfirmed frame	→	MAC-CMD LinkADRAns Payload = [0x]0307	
	The TCL sends Unconfirmed frame on RX1 window	←	DR = Max125kHzDR DR = Max125kHzDR - 2 CP-CMD EchoPayloadReq	
			FPort 224 Payload = [0x]08010203	
8	DUT sends Unconfirmed frame	→	DR = Max125kHzDR CP-CMD EchoPayloadAns FPort = 224 Payload = [0x]08020304	RX1 reply sent
	The TCL sends Unconfirmed frame on RX2 window	+	RX2DataRate - As set in Join- Accept frame above	
			CP-CMD EchoPayloadReq FPort 224 Payload = [0x]080A0B0C	
9	DUT sends Unconfirmed frame	→	CP-CMD EchoPayloadAns FPort = 224 Payload = [0x]080B0C0D	RX2 reply sent
10	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame	+	CP-CMD DutJoinReq FPort = 224 Payload = [0x]02	Revert the device to default values
11	DUT sends Join-Request	→	DR = Z (where Z = any allowed DR, refer [2])	
			DevNonce is greater than DevNonce of previous JR	



Step	Procedure		Frame Sequence	Test Purpose
		End Device - TCL	Frame	Turpose
	TCL sends Join-Accept response	←	Join-Accept is sent to DUT without a CFList.	
			Pre-testing with 8-channel gateway: CFListType = [0x]01 ChMask0 = [0x]00FF ChMask1 = [0x]0000 ChMask2 = [0x]0000 ChMask3 = [0x]0000 ChMask4 = [0x]0001	
12	DUT sends Unconfirmed or Confirmed frame	\rightarrow	Chinash [chiques	
	The TCL sends Unconfirmed frame	(CP-CMD TxPeriodicityChangeReq FPort = 224 Periodicity = 5 sec Payload = [0x]0601 If DUT sent Confirmed uplink	Uplink Periodicity set
13	DUT sends Unconfirmed or	→	frame, TCL must Acknowledge	
10	Confirmed frame	·	CP-CMD TxFramesCtrlReq	
	If DUT sent a Confirmed frame, then The TCL sends Unconfirmed frame Else, this step must be skipped		FPort = 224 Frame type = Unconfirmed Payload = [0x]0701	
	2.00, and otop must 20 ompped		If DUT sent Confirmed uplink frame, TCL must Acknowledge	
14	DUT sends Unconfirmed frame	\rightarrow		
	If FCtrl ADR Bit = false, then The TCL sends Unconfirmed frame Else, this step is skipped	+	CP-CMD AdrBitChangeReq-ON FPort = 224 Payload = [0x]0401	
15	DUT sends Unconfirmed frame	\rightarrow	FCtrl ADR Bit = true	ADR Bit turned ON
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq DataRate = Max125kHzDR, refer [2] Payload = [0x]03XXXXXXXX ChMaskCntl:	
			DC = 0, FC = 6 ChMask: DC - Enable only default	
			channels FC = [0x]00FF	



Step	Procedure	Frame Sequence		Test Purpose
		End Device - TCL	Frame	
16	DUT sends Unconfirmed frame	→	MAC-CMD LinkADRAns Payload = [0x]0307 DR = Max125kHzDR	

441

442

2.2.3. **Join-Accept with Delay Settings**

443444445446447

After the **TCL** triggers a JoinRequest, the **DUT** starts the Join procedure for over-the-air activation as above. The **TCL** responds with a Join-Accept frame containing Delay Settings on RX2 window, such that RX1 and subsequently RX2 timing is increased (at least 2 seconds are recommended). The **TCL** sends an echo command targeted to RX1 to which the **DUT** must respond correctly. The **TCL** repeats this same downlink test against the RX2 window, to which the **DUT** must respond correctly.

448

Verify

449 450

DUT successfully joins the network

2.2.3.a. Test Procedure Frame Sequence Chart

451

DUT implements the new (non-default) Delay Settings

452

DUT restores the default settings for RXDelay



Step	Procedure	Frame Sequence		Test Purpose
		End Device - TCL	Frame	Purpose
1	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame on RX2 window	+	CP-CMD DutJoinReq FPort = 224 Payload = [0x]02	Device Reinitializes [Not Joined]
2	DUT sends Join-Request	→	DR = Any allowed DR, refer	
			DevNonce is greater than DevNonce of previous JR	
	TCL sends Join-Accept response on RX2 window	←	RXDelay = n, where 2 <= n <=15 Join-Accept is sent to DUT	Join accepted with modified parameters
			without a CFList.	parameters
			Pre-testing with 8-channel gateway: CFListType = [0x]01 ChMask0 = [0x]00FF ChMask1 = [0x]0000 ChMask2 = [0x]0000 ChMask3 = [0x]0000	
		,	ChMask4 = [0x]0001	
3	DUT sends Unconfirmed or Confirmed frame FCntUp = n (where n = 0 or 1 for OTA devices)	→		
	The TCL sends Unconfirmed frame	←	CP-CMD TxPeriodicityChangeReq FPort = 224 Periodicity = 5 sec Payload = [0x]0601	Uplink Periodicity set
			If DUT sent Confirmed uplink frame, TCL must Acknowledge	
4	DUT sends Unconfirmed or Confirmed frame FCntUp = n + 1	→		
	If DUT sent a Confirmed frame, then The TCL sends Unconfirmed frame	+	CP-CMD TxFramesCtrlReq FPort = 224 Frame type = Unconfirmed	
	Else, this step must be skipped		Payload = [0x]0701	
			If DUT sent Confirmed uplink frame, TCL must Acknowledge	
5	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame on RX1 window	+	RX1Delay = n seconds	
			CP-CMD EchoPayloadReq FPort 224 Payload [0x]08010203	



Step	Procedure	Frame Sequence		Test Purpose
		End Device - TCL	Frame	i dipose
6	DUT sends Unconfirmed frame	→	CP-CMD EchoPayloadAns FPort 224 Payload [0x]08020304	RX1 delay reply sent
	The TCL sends Unconfirmed frame on RX2 window	+	RX2Delay = n + 1 seconds	
			CP-CMD EchoPayloadReq FPort 224 Payload [0x]080A0B0C	
7	DUT sends Unconfirmed frame	→	CP-CMD EchoPayloadAns FPort = 224 Payload [0x]080B0C0D	RX2 delay reply sent
	The TCL sends Unconfirmed frame	+	CP-CMD DutJoinReq FPort = 224 Payload = [0x]02	Revert the device to default values
8	DUT sends Join-Request)	DR = Z (where Z = any allowed DR, refer [2]) DevNonce is greater than	
	TCL sends Join-Accept response	←	DevNonce of previous JR Join-Accept is sent to DUT without a CFList.	
			Pre-testing with 8-channel gateway: CFListType = [0x]01 ChMask0 = [0x]00FF ChMask1 = [0x]0000 ChMask2 = [0x]0000 ChMask3 = [0x]0000 ChMask4 = [0x]0001	
9	DUT sends Unconfirmed or Confirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame	←	CP-CMD TxPeriodicityChangeReq FPort = 224 Periodicity = 5 sec Payload = [0x]0601 If DUT sent Confirmed uplink frame, TCL must Acknowledge	Uplink Periodicity set
10	DUT sends Unconfirmed or Confirmed frame	→	-	
	If DUT sent a Confirmed frame, then The TCL sends Unconfirmed frame Else, this step must be skipped	+	CP-CMD TxFramesCtrlReq FPort = 224 Frame type = Unconfirmed Payload = [0x]0701	
			If DUT sent Confirmed uplink frame, TCL must Acknowledge	
11	DUT sends Unconfirmed frame	\rightarrow		



Step	Procedure		Test	
			Purpose	
		End Device	Frame	
		- TCL		
	If FCtrl ADR Bit = false, then	+	CP-CMD AdrBitChangeReq-	
	The TCL sends Unconfirmed frame		ON	
			FPort = 224	
	Else, this step is skipped		Payload = [0x]0401	
12	DUT sends Unconfirmed frame	\rightarrow	FCtrl ADR Bit = true	ADR Bit
				turned ON
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq	
			DataRate = Max125kHzDR,	
			refer [2]	
			Payload = [0x]03XXXXXXXX	
			ChMaskCntl:	
			DC = 0,	
			FC = 6	
			ChMask:	
			DC - Enable only default	
			channels	
			FC = [0x]00FF	
13	DUT sends Unconfirmed frame	\rightarrow	MAC-CMD LinkADRAns	
			Payload = [0x]0307	
			DR = Max125kHzDR	

456

457

458

459 460

461

462

463 464

465 466

467

2.2.4. Join-Accept with CFList

2.2.4.a. For Dynamic Channel (DC) plan devices

After the **TCL** triggers a Join-Request, the **DUT** starts the Join procedure for over-theair activation as above. The **TCL** responds with a Join-Accept frame containing an additional single channel in the CFList field. The **DUT** must use this additional channel together with the default channels within the following uplinks.

The **TCL** again triggers the DUT to send a Join-Request. The **TCL** responds with a Join-Accept frame containing a CFListType = 1. The **DUT** rejects the CFList and enables all default channels. The **DUT** must not use the additional channel added earlier.

Verify

- **DUT** successfully joins the network
- **DUT** uses the new channel in its random selection of frequencies
- **DUT** removes the additional channel added earlier.

468 469

2.2.4.a.i. Test Procedure Frame Sequence Chart



Step	Procedure	Frame Sequence		Test
		End Device - TCL	Frame	Purpose
1	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame	+	CP-CMD DutJoinReq FPort = 224 Payload = [0x]02	Device Reinitializes [Not Joined]
2	DUT sends Join-Request	→	DR = Any allowed DR, refer [2] DevNonce is greater than	
			DevNonce of previous JR	
	TCL sends Join-Accept response	+	CFList = add single channel CFListType = 0	Join accepted
3	DUT sends Unconfirmed or Confirmed frame FCntUp = n (where n = 0 or 1 for OTA devices)	→		
	The TCL sends Unconfirmed frame	←	CP-CMD TxPeriodicityChangeReq FPort = 224 Periodicity = 5 sec Payload = [0x]0601	Uplink Periodicity set
			If DUT sent Confirmed uplink frame, TCL must Acknowledge	
4	DUT sends Unconfirmed or Confirmed frame FCntUp = n + 1)		
	If DUT sent a Confirmed frame, then The TCL sends Unconfirmed frame Else, this step must be skipped	+	CP-CMD TxFramesCtrlReq FPort = 224 Frame type = Unconfirmed Payload = [0x]0701	
			If DUT sent Confirmed uplink frame, TCL must Acknowledge	
5	DUT sends Unconfirmed frame FCntUp = n + 2	→		
6	Wait until the new channel which was added has been used at least once. Wait for a maximum of [5* (number of channels configured)] uplink packets to be sent.	→ R [5*NbCh] or [AllCh used]	Channel added is used at least once	DUT adds the additional channel to its default channel plan
7	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame	←	CP-CMD DutJoinReq FPort = 224 Payload = [0x]02	Device Reinitializes [Not Joined]
8	DUT sends Join-Request	→	DR = Any allowed DR, refer [2] DevNonce is greater than DevNonce of previous JR	



Step	Procedure	Frame Sequence		Test Purpose	
		End Device - TCL	Frame		
	TCL sends Join-Accept response	+	CFList = add another channel CFListType = 1	Join-Accept sent with CFListType = 1	
9	DUT sends Unconfirmed or Confirmed frame FCntUp = n (where n = 0 or 1 for OTA devices))			
	The TCL sends Unconfirmed frame	←	CP-CMD TxPeriodicityChangeReq FPort = 224 Periodicity = 5 sec Payload = [0x]0601 If DUT sent Confirmed uplink frame, TCL must	Uplink Periodicity set	
10	DUT sends Unconfirmed or	→	Acknowledge		
	Confirmed frame FCntUp = n + 1				
	If DUT sent a Confirmed frame, then The TCL sends Unconfirmed frame	+	CP-CMD TxFramesCtrlReq FPort = 224 Frame type = Unconfirmed		
	Else, this step must be skipped		Payload = [0x]0701 If DUT sent Confirmed uplink frame, TCL must Acknowledge		
11	DUT sends Unconfirmed frame FCntUp = n + 2	→			
12	Wait for [5* (number of channels configured)] uplink packets to be sent.	→ R [5*NbCh] or [AllCh used]	 Default channels are used at least once. The additional channel is not used. 	DUT removes the additional channel from its default channel plan	

473

2.2.4.b. For Fixed Channel (FC) plan devices

474475476

After the TCL triggers a Join-Request, the DUT starts the Join procedure for over-the-air activation as above.

477

The TCL responds with a Join-Accept frame with CFListType set to [0x]01 in the CFList field. The ChMask fields must enable 3 channels (Channel 0, 1 and 64) and disable all other channels.

478479

Verify

480 481 • DUT successfully joins the network

482

• The DUT must use only the channels enabled for the uplinks.





The TCL triggers a Join-Request again and responds with a Join-Accept frame with 483 CFListType set to [0x]00 in the CFList field. The ChMask fields must enable 3 channels 484 (Channel 0, 1 and 64) and disable all other channels. 485 486 Verify 487 DUT successfully joins the network 488 The DUT must reject the CFList as the value of the CFListType = [0x]00 and 489 must use all channels for uplinks. For the sake of verification, verify that any 490 other channel other than the channels enabled in the ChMask bits are used for 491 the uplinks. 492 493 Finally, the TCL triggers a Join-Request again and responds with a Join-Accept frame with CFListType set to [0x]01 in the CFList field. The ChMask fields must be set to 494 [0x]FFFF. 495 496 Verify 497 • DUT successfully joins the network 498 The DUT must use all the channels for the uplinks. 499 2.2.4.b.i. 500 **Test Procedure Frame Sequence Chart** 501



Step	Procedure		Frame Sequence	Test Purpose
		End Device - TCL	Frame	i uipose
1	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame	+	CP-CMD DutJoinReq FPort = 224 Payload = [0x]02	Device Reinitializes [Not Joined]
2	DUT sends Join-Request frame	→	DevNonce is greater than DevNonce of previous JR	[Not Somed]
	TCL sends Join-Accept response	+	CFListType = [0x]01 ChMask0 = [0x]0003 ChMask1 = [0x]0000 ChMask2 = [0x]0000 ChMask3 = [0x]0000 ChMask4 = [0x]0001	Join accepted with CFList
3	DUT sends Unconfirmed or Confirmed frame FCntUp = n (where n = 0 or 1 for OTA devices)	→		
	The TCL sends Unconfirmed frame	←	CP-CMD TxPeriodicityChangeReq FPort = 224 Periodicity = 5 sec Payload = [0x]0601	Uplink Periodicity set
			If DUT sent Confirmed uplink frame, TCL must Acknowledge	
4	DUT sends Unconfirmed or Confirmed frame in 5 seconds FCntUp = n + 1	→	FPort = any allowed port except 224	Next Uplink sent in 5 seconds
5	Wait for a maximum of 5 uplinks to be sent.	→ R [max 5]	If DR = 125kHz DR, only Channels 0 and 1 must be used for all uplinks If DR = 500kHz DR, only channel 64 must be used for	Only the enabled channels are used for uplinks
			uplinks	
6	DUT sends a Confirmed or Unconfirmed frame	→		
	The TCL sends Unconfirmed frame	←	CP-CMD DutJoinReq FPort = 224 Payload = [0x]02 If DUT sent Confirmed uplink frame, TCL must Acknowledge	Device Reinitializes [Not Joined]
7	DUT sends Join-Request frames.	→	DevNonce is greater than DevNonce of previous JR	
	TCL ignores the Join-Request frames and waits for a maximum of (2 * number of channels configured) uplink transmissions, until the Join-Request channel is a 500kHz channel	R [2*NbCh] or [500kHz channel]		



Step	Procedure	1	Frame Sequence	Test Purpose	
		End Device - TCL	Frame		
	TCL sends Join-Accept response	+	CFListType = [0x]00 ChMask0 = [0x]0003 ChMask1 = [0x]0000 ChMask2 = [0x]0000 ChMask3 = [0x]0000 ChMask4 = [0x]0001	Join accepted with CFListType = [0x]00	
8	DUT sends Unconfirmed or Confirmed frame FCntUp = n (where n = 0 or 1 for OTA devices)	→			
	The TCL sends Unconfirmed frame	←	CP-CMD TxPeriodicityChangeReq FPort = 224 Periodicity = 5 sec Payload = [0x]0601 If DUT sent Confirmed uplink frame, TCL must Acknowledge	Uplink Periodicity set	
9	DUT sends Unconfirmed or Confirmed frame in 5 seconds FCntUp = n + 1	→	FPort = any allowed port except 224	Next Uplink sent in 5 seconds	
10	DUT sends Unconfirmed or Confirmed frame FCntUp = n (where n = 0 or 1 for OTA devices)	→			
	The TCL sends Unconfirmed frame	+	CP-CMD TxPeriodicityChangeReq FPort = 224 Periodicity = 5 sec Payload = [0x]0601 If DUT sent Confirmed uplink frame, TCL must Acknowledge	Uplink Periodicity set	
11	DUT sends Confirmed or Unconfirmed frame in 5 seconds FCntUp = n + 1	→	FPort = any allowed port except 224	Next Uplink sent in 5 seconds	
	If DUT sent a Confirmed frame, then The TCL sends Unconfirmed frame Else, this step must be skipped	+	CP-CMD TxFramesCtrlReq FPort = 224 Frame type = Unconfirmed Payload = [0x]0701		
			If DUT sent Confirmed uplink frame, TCL must Acknowledge		



Step	Procedure	ı	Frame Sequence	Test Purpose
		End Device - TCL	Frame	ruipose
12	If Data Rate = 500kHz DR, wait for (5 * number of 500kHz channels configured) uplink transmissions (max. 16 packets for 500kHz channels), i.e. until all 500kHz channels configured are used at least once If Data Rate = 125kHz DR, skip this	→ R [5*Nb500kH zCh] or [AllCh used]	Official certification (64-channel gateway): All 500kHz channels must be used at least once Pre-testing with 8-channel gateway: Channel 64 must be used at least once.	All 500kHz channels must be used at least once
13	step DUT sends a Confirmed or	→		
	Unconfirmed frame The TCL sends Unconfirmed frame	+	CP-CMD DutJoinReq FPort = 224 Payload = [0x]02 If DUT sent Confirmed uplink frame, TCL must Acknowledge	Device Reinitializes [Not Joined]
14	DUT sends Join-Request frames.	→	DevNonce is greater than DevNonce of previous JR	
	TCL ignores the Join-Request frames and waits for a maximum of (2 * number of channels configured) uplink transmissions, until the Join-Request channel is a 125kHz channel	R [2*NbCh] or [125kHz channel]		
	TCL sends Join-Accept response	←	Join-Accept must be sent without a CFList. Pre-testing with 8-channel gateway: CFListType = [0x]01 ChMask0 = [0x]00FF ChMask1 = [0x]0000 ChMask2 = [0x]0000 ChMask3 = [0x]0000 ChMask4 = [0x]0001	Join accepted without a CFList
15	DUT sends Unconfirmed or Confirmed frame FCntUp = n (where n = 0 or 1 for OTA devices)	÷		
	The TCL sends Unconfirmed frame	+	CP-CMD TxPeriodicityChangeReq FPort = 224 Periodicity = 5 sec Payload = [0x]0601 If DUT sent Confirmed uplink frame, TCL must Acknowledge	Uplink Periodicity set



Step	Procedure	ı	Test Purpose	
		End Device - TCL	Frame	·
16	DUT sends Confirmed or Unconfirmed frame in 5 seconds FCntUp = n + 1)	FPort = any allowed port except 224	Next Uplink sent in 5 seconds
	If DUT sent a Confirmed frame, then The TCL sends Unconfirmed frame Else, this step must be skipped	+	CP-CMD TxFramesCtrlReq FPort = 224 Frame type = Unconfirmed Payload = [0x]0701	
			If DUT sent Confirmed uplink frame, TCL must Acknowledge	
17	Wait for (2 * number of 125kHz channels configured) uplink transmissions (max. 128 packets for 125kHz channels), i.e. until all 125kHz channels configured are	→ R [2*Nb125kH zCh] or [AllCh used]	Official certification (64- channel gateway): All 125kHz channels must be used at least once	All 125kHz channels must be used at least once
	used at least once		Pre-testing with 8-channel gateway: Channels 0-7 must be used at least once.	
18	DUT sends Unconfirmed frame	\rightarrow		
	If FCtrl ADR Bit = false, then The TCL sends Unconfirmed frame	+	CP-CMD AdrBitChangeReq- ON FPort = 224 Payload = [0x]0401	
	Else, this step is skipped			
19	DUT sends Unconfirmed frame	→	FCtrl ADR bit = true	Turn on the ADR bit
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq DataRate = Max125kHzDR, refer [2] Payload = [0x]03XXXXXXXX	
			ChMaskCntl = 6 ChMask = [0x]00FF	
20	DUT sends Unconfirmed frame	→	MAC-CMD LinkADRAns Payload = [0x]0307	Changed the DR to Max125kHz DR

503

505

506507

2.3. Activation by Personalization

This test must be performed if the **DUT** supports activation by personalization.

For this test, the Authorised Test Lab must use the device provided by the device vendor to test the ABP activation functionality.

After initial power-up, the **DUT** must be enabled for testing as specified in Section 2.1.1.





509	Verify
510	DUT successfully joins the network.
511	
512	The Test Procedure Message Sequence Chart for this test is the same as Section 2.1.1.
513	
514	Additionally, the DUT must also retain its previous settings even after reset.
515	TCL sets the DUT's parameters using the MAC commands RXParamSetupReq, DIChannelReq
516	and RXTimingSetupReq.
517	The DUT is then reset using the CP-CMD DutResetReq.
518	
519	Verify
520	 The DUT's parameters must be retained even after reset.
521	2.3.1. Dynamic channel plan devices
522 523	2.3.1.a. All regions - Test Procedure Frame Sequence Chart



Step	Procedure		Frame Sequence	Test Purpose
		End Devic e - TCL	Frame	
1	Perform all steps mentioned in Section 2.1.1		Same results as mentioned in Section 2.1.1	
2	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame	←	MAC-CMD LinkADRReq TXPower = Minimum, refer [2] DataRate = Max125kHzDR, refer [2] ChMaskCntl = 0 ChMask = [0x]0001 MAC-CMD RxParamSetupReq RX1DRoffset = any allowed offset value except default, refer [2] RX2DataRate = Any DataRate allowed except default, refer [2] RX2Frequency = Y (where Y = any frequency allowed except default, refer [2]) MAC-CMD DIChannelReq ChIndex = C (where C = Any default channel, refer [2]) Freq = X (where X = any allowed frequency except default, refer [2]) MAC-CMD RxTimingSetupReq Delay (i) = 2 Payload = [0x]03XXXXXXXXXXX[0x]05XXXXXXXX [0x]0AXXXXXXXXXX[0x]08XX	Channel 0 enabled RxParamSet upReq, DIChannelR eq, RxTimingSet upReq commands executed
3	DUT sends Unconfirmed frame	→	MAC-CMD LinkADRAns MAC-CMD RxParamSetupAns MAC-CMD DIChannelAns MAC-CMD RxTimingSetupAns Payload = [0x]0307[0x]0507[0x]0A03[0x]08	
	The TCL sends Unconfirmed frame	+	CP-CMD DutResetReq FPort = 224 Payload = [0x]01	Reset the DUT
4	DUT sends Confirmed or Unconfirmed frame	→	If the ADR Bit is set, DataRate (DR) = Minimum DR allowed by the DUT, refer [2]	



	The TCL sends Confirmed frame on RX1 window	←	RX1DRoffset = as set in Step 2 RX1Delay = as set in Step 2 Freq = as set in Step 2	RX1DRoffset , RX1Delay and Freq
				retained after reset
			CP-CMD TxPeriodicityChangeReq FPort = 224	alter reset
			Periodicity = 5 sec Payload = [0x]0601	
			If DUT sent Confirmed uplink frame, TCL must Acknowledge	
5	DUT sends Confirmed or Unconfirmed frame	\rightarrow	ACK Bit = True	
	If DUT sent a Confirmed frame, then The TCL sends Unconfirmed frame	+	CP-CMD TxFramesCtrlReq FPort = 224	
	Else, this step must be skipped		Frame type = Unconfirmed Payload = [0x]0701	
			If DUT sent Confirmed uplink frame, TCL must Acknowledge	
6	Wait for a maximum of 5 * (number	→R	All default channels must be used at	All channels
	of default channels)	[5*Nb	least once	must be
		ChDC		enabled after
] or		the reset
		[AllCh used]		
7	DUT sends Unconfirmed frame	→ ·		
	If FCtrl ADR Bit = false, then	+	CP-CMD AdrBitChangeReq-ON	
			FPort = 224	
	The TCL sends Unconfirmed frame		Payload = [0x]0401	
	Else, this step is skipped			
8	DUT sends Unconfirmed frame	→	FCtrl ADR bit = true	Turn on the ADR bit
	The TCL sends Confirmed frame on RX2 window	←	RX2DataRate = as set in Step 2 RX2Frequency = as set in Step 2	RX2DataRat e, RX2Freq setting
			MAC-CMD RxParamSetupReq	retained
			RX1DRoffset = default, refer [2]	after reset
			RX2DataRate = default, refer [2]	
			RX2Frequency = default, refer [2]	Later, revert to default for
			MAC-CMD DIChannelReq ChIndex = C	all settings.
			Freq = default, refer [2]	Set DR to Max125kHz
			MAC-CMD RxTimingSetupReq Delay (i) = default, refer [2]	DR
			MAC-CMD LinkADRReq DataRate = Max125kHzDR, refer [2]	
			Payload = [0x]05XXXXXXXX[0x]0AXXXXXXX X[0x]08XX[0x]03XXXXXXXX	



9	DUT sends Unconfirmed frame	\rightarrow	ACK Bit = True	
			MAC-CMD RxParamSetupAns MAC-CMD DIChannelAns MAC-CMD RxTimingSetupAns MAC-CMD LinkADRAns Payload = [0x]0507[0x]0A03[0x]08[0x]0307	

2.3.1.b. For regions with Dwell Time limitation only - Test Procedure Frame Sequence Chart

These tests must be performed only for regions with Dwell Time limitation. **TCL** sets the **DUT**'s UplinkDwellTime using the TXParamSetupReq MAC command. The **DUT** is then reset using the CP-CMD DutResetReq.

Verify

524

525

526527

528

529530

531

532533

• The UplinkDwellTime setting must be retained even after reset.



Step	Procedure		Frame Sequence	Test Purpose
		End Devic e - TCL	Frame	, a pood
1	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame	+	MAC-CMD TXParamSetupReq UplinkDwellTime = 0 Payload = [0x]09XX	TXParamSet upReq command executed
2	DUT sends Unconfirmed frame	→	MAC-CMD TXParamSetupAns Payload = [0x]09	
	The TCL sends Unconfirmed frame	+	CP-CMD DutResetReq FPort = 224 Payload = [0x]01	Reset the DUT
3	DUT sends Confirmed or Unconfirmed frame	\rightarrow		
	The TCL sends Confirmed frame on RX1 window	+	CP-CMD TxPeriodicityChangeReq FPort = 224 Periodicity = 5 sec Payload = [0x]0601	Change periodicity to 5sec
			If DUT sent Confirmed uplink frame, TCL must Acknowledge	
4	DUT sends Confirmed or Unconfirmed frame	→	ACK Bit = True	
	If DUT sent a Confirmed frame, then The TCL sends Unconfirmed frame Else, this step must be skipped	←	CP-CMD TxFramesCtrlReq FPort = 224 Frame type = Unconfirmed Payload = [0x]0701	
			If DUT sent Confirmed uplink frame, TCL must Acknowledge	
5	DUT sends Unconfirmed frame	\rightarrow		
	If FCtrl ADR Bit = false, then The TCL sends Unconfirmed frame Else, this step is skipped	+	CP-CMD AdrBitChangeReq-ON FPort = 224 Payload = [0x]0401	
6	DUT sends Unconfirmed frame	→	FCtrl ADR bit = true	Turn on the ADR bit
	The TCL sends Unconfirmed frame on RX2 window	+	MAC-CMD LinkADRReq DataRate = MinDR, refer [2] Payload = [0x]03XXXXXXXX	UplinkDwellT ime setting retained after reset- tested by setting to MinDR
7	DUT sends Unconfirmed frame in 5 seconds	→	MAC-CMD LinkADRAns Payload = [0x]0307	
8	DUT sends Unconfirmed frame	\rightarrow		





	The TCL sends Unconfirmed frame	←	MAC-CMD LinkADRReq	Revert to
			DataRate = Max125kHzDR, refer [2]	Max125kHz
				DR and
			MAC-CMD TXParamSetupReq	revert
			UplinkDwellTime = default, refer [2]	UplinkDwellT
				ime to
			Payload =	default
			[0x]03XXXXXXXX[0x]09XX	
9	DUT sends Unconfirmed frame	\rightarrow	MAC-CMD LinkADRAns	
			MAC-CMD TXParamSetupAns	
			Payload = $[0x]0307[0x]09$	

535

2.3.2. Fixed channel plan devices

2.3.2.a. All regions - Test Procedure Frame Sequence Chart



Step	Procedure		Frame Sequence	Test Purpose
		End Devic e - TCL	Frame	
1	Perform all steps mentioned in Section 2.1.1		Same results as mentioned in Section 2.1.1	
2	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq TXPower = Minimum, refer [2] DataRate = Max500kHzDR, refer [2] ChMaskCntl = 7 ChMask = [0x]0001 MAC-CMD LinkADRReq TXPower = Minimum, refer [2] DataRate = Max125kHzDR, refer [2] ChMaskCntl = 0 ChMask = [0x]0003 MAC-CMD RxParamSetupReq RX1DRoffset = any allowed offset value except default, refer [2] RX2DataRate = Any DataRate allowed except default, refer [2] RX2Frequency = Y (where Y = any frequency allowed except default, refer [2]) MAC-CMD RxTimingSetupReq Delay (i) = 2 Payload = [0x]03XXXXXXXXXXX[0x]03XXXXXXXXX [0x]05XXXXXXXXXX[0x]08XX	Channels 0, 1 and 64 enabled RxParamSet upReq, RxTimingSet upReq commands executed
3	DUT sends Unconfirmed frame	→	MAC-CMD LinkADRAns MAC-CMD LinkADRAns MAC-CMD RxParamSetupAns MAC-CMD RxTimingSetupAns Payload = [0x]0307[0x]0307[0x]0507[0x]08	
4	The TCL sends Unconfirmed frame	+	CP-CMD DutResetReq FPort = 224 Payload = [0x]01	Reset the DUT
	DUT sends Confirmed or Unconfirmed frame	→		



	The TCL sends Confirmed frame on RX1 window	+	RX1DRoffset = as set in Step 2 RX1Delay = as set in Step 2 Freq = as set in Step 2 CP-CMD TxPeriodicityChangeReq FPort = 224 Periodicity = 5 sec Payload = [0x]0601	RX1DRoffset , RX1Delay and Freq retained after reset
5	DUT sends Confirmed or	→	If DUT sent Confirmed uplink frame, TCL must Acknowledge ACK Bit = True	
3	Unconfirmed frame	7	ACK Bit = True	
	If DUT sent a Confirmed frame, then The TCL sends Unconfirmed frame Else, this step must be skipped	+	CP-CMD TxFramesCtrlReq FPort = 224 Frame type = Unconfirmed Payload = [0x]0701 If DUT sent Confirmed uplink frame,	
			TCL must Acknowledge	
6	If DR = 125kHz DR, wait for a maximum of 2 * (number of 125kHz channels configured) uplink packets to be sent, i.e. until all channels are used at least once.	→ R [2*Nb 125k HzCh] or [5*Nb	If DR = 125kHz DR, all 125kHz channels including the channels which were disabled earlier must be used at least once If DR = 500kHz DR, all 500kHz	All channels must be enabled after the reset
	If DR = 500kHz DR, wait for a maximum of 5 * (number of 500kHz channels configured) uplink packets to be sent, i.e. until all channels are used at least once.	500k HzCh] or [AllCh used]	channels including the channels which were disabled earlier must be used at least once Pre-testing for FC plan with 8-channel gateway: Channels 0-7 (if DR = 125kHz DR) or channel 64 (if DR = 500kHz DR) must be used at least once.	
7	DUT sends Unconfirmed frame	→		
	If FCtrl ADR Bit = false, then The TCL sends Unconfirmed frame Else, this step is skipped	+	CP-CMD AdrBitChangeReq-ON FPort = 224 Payload = [0x]0401	
8	DUT sends Unconfirmed frame	\rightarrow	FCtrl ADR bit = true	Turn on the
				ADR bit



	The TCL sends Confirmed frame	+	RX2DataRate = as set in Step 2	RX2DataRat
	on RX2 window		RX2Frequency = as set in Step 2	e, RX2Freq,
				retained
				after reset
			MAC-CMD RxParamSetupReq	
			RX1DRoffset = default, refer [2]	Revert to
			RX2DataRate = default, refer [2]	default for all
			RX2Frequency = default, refer [2]	settings.
			MAC-CMD RxTimingSetupReq	Changed the
			Delay (i) = default, refer [2]	DR to
			20.03 (.) Condan, 10.01 [2]	Max125kHz
			MAC-CMD LinkADRReq	DR
			DataRate = Max125kHzDR, refer [2]	
			Payload =	
			[0x]05XXXXXXXX[0x]08XX[0x]03XX	
			XXXXXX	
9	DUT sends Unconfirmed frame	\rightarrow	ACK Bit = True	
			MAG OMB Bubana as Catara A	
			MAC-CMD RxParamSetupAns	
			MAC-CMD LinkADDAns	
			MAC-CMD LinkADRAns	
			Payload = [0x]0507[0x]08[0x]0307	

539540

541542

2.3.2.b. For regions with Dwell Time limitation only - Test Procedure Frame Sequence Chart

These tests must be performed only for regions with Dwell Time limitation. **TCL** sets the **DUT**'s UplinkDwellTime using the TXParamSetupReq MAC command. The **DUT** is then reset using the CP-CMD DutResetReq.

The UplinkDwellTime setting must be retained even after reset.

543544

545 Verify

546 547



Step	Procedure		Test Purpose	
		End Devic e - TCL	Frame	, a pood
1	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame	+	MAC-CMD TXParamSetupReq UplinkDwellTime = 0 Payload = [0x]09XX	TXParamSet upReq command executed
2	DUT sends Unconfirmed frame	→	MAC-CMD TXParamSetupAns Payload = [0x]09	
	The TCL sends Unconfirmed frame	+	CP-CMD DutResetReq FPort = 224 Payload = [0x]01	Reset the DUT
3	DUT sends Confirmed or Unconfirmed frame	\rightarrow		
	The TCL sends Confirmed frame on RX1 window	+	CP-CMD TxPeriodicityChangeReq FPort = 224 Periodicity = 5 sec Payload = [0x]0601	Change periodicity to 5sec
			If DUT sent Confirmed uplink frame, TCL must Acknowledge	
4	DUT sends Confirmed or Unconfirmed frame	→	ACK Bit = True	
	If DUT sent a Confirmed frame, then The TCL sends Unconfirmed frame Else, this step must be skipped	←	CP-CMD TxFramesCtrlReq FPort = 224 Frame type = Unconfirmed Payload = [0x]0701	
			If DUT sent Confirmed uplink frame, TCL must Acknowledge	
5	DUT sends Unconfirmed frame	\rightarrow		
	If FCtrl ADR Bit = false, then The TCL sends Unconfirmed frame Else, this step is skipped	+	CP-CMD AdrBitChangeReq-ON FPort = 224 Payload = [0x]0401	
6	DUT sends Unconfirmed frame	→	FCtrl ADR bit = true	Turn on the ADR bit
	The TCL sends Unconfirmed frame on RX2 window	+	MAC-CMD LinkADRReq DataRate = MinDR, refer [2] Payload = [0x]03XXXXXXXX	UplinkDwellT ime setting retained after reset- tested by setting to MinDR
7	DUT sends Unconfirmed frame in 5 seconds	→	MAC-CMD LinkADRAns Payload = [0x]0307	
8	DUT sends Unconfirmed frame	\rightarrow		





	The TCL sends Unconfirmed frame	←	MAC-CMD LinkADRReq	Revert to
			DataRate = Max125kHzDR, refer [2]	Max125kHz
				DR and
			MAC-CMD TXParamSetupReq	revert
			UplinkDwellTime = default, refer [2]	UplinkDwellT
				ime to
			Payload =	default
			[0x]03XXXXXXXX[0x]09XX	
9	DUT sends Unconfirmed frame	\rightarrow	MAC-CMD LinkADRAns	
			MAC-CMD TXParamSetupAns	
			Payload = [0x]0307[0x]09	



2.4. Device Functionality Tests

2.4.1. Default Setting Tests

				0										
The T	CL wi	ll test t	the b	asic fun	ctiona	lity of the C	UT υ	ising the o	curr	ent apı	olicativ	/e '	RxA	ppCnt
value	and	using	the	[0x]08	Echo	command.	The	purpose	of	these	tests	is	to	detec

implementation errors in the **DUT** early, instead of failing later tests.

554555

556

557

558

559

560

561

562563

564

565566

567

568

569570

550

551

552553

2.4.1.a. **Cryptography:**

Verifies that AES encryption and message integrity code (MIC) algorithms are correctly implemented by the **DUT**.

2.4.1.a.i. **AES Encryption**

TCL will send multiple [0x]08 Echo commands with varying length payloads to the **DUT**. The varying payloads lengths must contain lengths

- Test with a physical frame size smaller than 16 bytes. (Less than block of 16 bytes,)
- Test with a physical frame size equal to 16 bytes. (16 bytes,)
- Test with a physical frame where the size is between 17 and 31 bytes. (Greater than block of 16 bytes but lesser than the second block,)
- Test with a physical frame size equal to the maximum which is 255 bytes (Max payload length). For this test used data rate is important as the max length depends on it.

The echo-replies from the **DUT** are verified to contain the same payload where each byte is incremented by one. This test will only use echo commands whose payload is less than or equal to the maximum allowed payload of the **DUT**'s current uplink DR. Frame counter must increase for every received package. Oversized payloads are tested separately in Section 2.5.11.d.



2.4.1.a.i.1. Test Procedure Frame Sequence Chart

Step	Procedure		Test Purpose	
		End Device - TCL	Frame	
1	DUT sends Unconfirmed frame	\rightarrow		
	FCntUp = n			
	The TCL sends Unconfirmed frames	←	CP-CMD RxAppCntReq	
			FPort = 224	
			Payload = [0x]09	
2	DUT sends Unconfirmed frame	\rightarrow	CP-CMD RxAppCntAns	
	FCntUp = n + 1		FPort = 224	
			Payload = [0x]09XXXX	
			RxAppCnt = x	
	The TCL sends Unconfirmed frames	←	CP-CMD EchoPayloadReq	
			FPort = 224	
			Payload = [0x]08 (Various)	
			Note: See description for	
			Various details	
3	DUT sends Unconfirmed frame	→ R [3]	CP-CMD EchoPayloadAns	Multiple
	FCntUp = n + 1 + i		(repeat i times- where i = 1 to	Echo replies
			3)	sent
			FPort = 224	
			Payload = [0x]08 (Various)'	
			No ACK is sent for the	
			previous Unconfirmed frame	
	The TCL sends Unconfirmed frames	← R [3]	CP-CMD EchoPayloadReq	
			(repeat i times- where i = 1 to	
			3)	
			FPort = 224	
			Payload = [0x]08 (Various)	
4	DUT sends Unconfirmed frames	\rightarrow	CP-CMD EchoPayloadAns	Echo reply
	FCntUp = n + 5		FPort = 224	sent
			Payload = [0x]08 (Various)'	
	The TCL sends Unconfirmed frames	+	CP-CMD RxAppCntReq	
			FPort = 224	
			Payload = [0x]09	
5	DUT sends Unconfirmed frame	\rightarrow	CP-CMD RxAppCntAns	Downlink
	FCntUp = n + 6		FPort = 224	counter
			Payload = [0x]09XXXX	incremented
			RxAppCnt = x + 5	

577

578

579

580

576

2.4.1.a.ii. Message Integrity Code

TCL will send packets with purposely invalid message integrity codes. The **DUT** must ignore these packets.



2.4.1.a.ii.1. Test Procedure Frame Sequence Chart

Step	Procedure		Test Purpose	
		End Device - TCL	Frame	
1	DUT sends Unconfirmed frame FCntUp = n	→		
	The TCL sends Unconfirmed frames	(CP-CMD RxAppCntReq FPort = 224 Payload = [0x]09	
2	DUT sends Unconfirmed frame FCntUp = n + 1	→	CP-CMD RxAppCntAns FPort = 224 Payload = [0x]09XXXX RxAppCnt = x	
	The TCL sends Unconfirmed frames	+	CP-CMD EchoPayloadReq FPort = 224 Payload = [0x]08 (Various) MIC Invalid	
3	DUT sends Unconfirmed frame FCntUp = n + 1 + i	→ R [4]		MIC Invalid packets ignored
	The TCL sends Unconfirmed frames	← R [4]	CP-CMD EchoPayloadReq (repeat i times- where i = 1 to 4) FPort = 224 Payload = [0x]08 (Various) MIC Invalid	
4	DUT sends Unconfirmed frame FCntUp = n + 6	→		MIC Invalid packet ignored
	The TCL sends Unconfirmed frames	+	CP-CMD RxAppCntReq FPort = 224 Payload = [0x]09	
5	DUT sends Unconfirmed frame FCntUp = n + 7	→	CP-CMD RxAppCntAns FPort = 224 Payload = [0x]09XXXX RxAppCnt = x + 1	Downlink counter not incremented

582

581

583

584

585 586

587

588

589

590

2.4.1.b. **Downlink Sequence Number**

This test verifies the **DUT** properly handles the frame sequence numbers that are used to prevent replay attacks on the communication.

TCL sends several packets with decreasing sequence number i such that:

$1 \le i < Current FCntDown$

The **DUT** must ignore downlinks whose sequence numbers are less than its current internal downlink counter value.



2.4.1.b.i. Test Procedure Frame Sequence Chart

Step	Procedure		Test	
			Purpose	
		End	Frame	
		Device -		
4	DIT III (II	TCL		
1	DUT sends Unconfirmed frame FCntUp = n	\rightarrow		
	The TCL sends Unconfirmed frames	←	CP-CMD RxAppCntReq FPort = 224 Payload = [0x]09	
2	DUT sends Unconfirmed frame FCntUp = n + 1	→	CP-CMD RxAppCntAns FPort 224 Payload = [0x]09XXXX RxAppCnt = x	
	The TCL sends Unconfirmed frame FCntDown = a	←	CP-CMD TxFramesCtrlReq FPort = 224 Frame type = No change Payload = [0x]0700	
3	DUT sends Unconfirmed frame FCntUp = n + 1 + y Repeat y times	→ R [y]		
	The TCL sends Unconfirmed frame FCntDown = a – i (where i = [1, a-1]) Repeat y times	← R [y]	CP-CMD TxFramesCtrlReq FPort = 224 Frame type = No change Payload = [0x]0700	
4	DUT sends Unconfirmed frame FCntUp = n + 2 + y	\rightarrow		
	The TCL sends Unconfirmed frames	←	CP-CMD RxAppCntReq FPort = 224 Payload = [0x]09	
5	DUT sends Unconfirmed frame FCntUp = n + 3 + y	→	CP-CMD RxAppCntAns FPort = 224 Payload = [0x]09XXXX RxAppCnt = x + 2	DUT ignores y downlinks with FCntDown < current downlink counter

592593

594

595

596

598

599

591

2.4.2. Confirmed Frames

This test verifies that the **DUT** properly handles confirmed frames as both the sender (uplinks) and receiver (downlinks).

597 2.4.2.a. **Confirmed Uplinks**

The **TCL** sends the [0x]0702 TxFramesCtrlReq command to the **DUT**. Verify all subsequent uplinks from the **DUT** are frame type *ConfirmedUp*.

2.4.2.a.i. Test Procedure Frame Sequence Chart



Step	Procedure		Frame Sequence	Test Purpose
		End Device - TCL	Frame	- Tanpess
1	DUT sends Unconfirmed frame FCntUp = n	→		
	The TCL sends Unconfirmed frames	+	CP-CMD RxAppCntReq FPort = 224 Payload = [0x]09	
2	DUT sends Unconfirmed frame FCntUp = n + 1	→	CP-CMD RxAppCntAns FPort = 224 Payload = [0x]09XXXX RxAppCnt = x	
	The TCL sends Unconfirmed frame	+	CP-CMD TxFramesCtrlReq FPort 224 Frame Type = Confirmed Payload [0x]0702	
3	DUT sends Confirmed frame FCntUp >= n + 2	→		Confirmed frame sent
	The TCL sends Unconfirmed frame	+	Acknowledge No FPort and no payload	
4	DUT sends Confirmed frame FCntUp >= n + 3 The TCL sends Unconfirmed frame	<i>→</i>	Acknowledge	Confirmed frame sent
			CP-CMD TxFramesCtrlReq FPort = 224 Frame type = No change Payload [0x]0700	
5	DUT sends Confirmed frame FCntUp >= n + 4	→		Confirmed frame sent
	The TCL sends Unconfirmed frames	←	Acknowledge CP-CMD RxAppCntReq FPort = 224 Payload = [0x]09	
6	DUT sends Confirmed frame FCntUp >= n + 5	→	CP-CMD RxAppCntAns FPort = 224 Payload = [0x]09XXXX RxAppCnt >= x + 4	
7	DUT sends Confirmed frame FCntUp >= n + 6)		DUT increments the FCntUp even when no ACK is sent by TCL
	The TCL sends Unconfirmed frame	+	Acknowledge CP-CMD TxFramesCtrlReq FPort 224 Frame type = Unconfirmed Payload [0x]0701	





_				
	8	DUT sends Unconfirmed frame	\rightarrow	Switch back
		FCntUp >= n + 7		to
				Unconfirmed
				frame

6	0	2
v		

605

606

2.4.2.b. Confirmed Downlinks

TCL sends a *ConfirmedDown* packet. Verify the **DUT** sets the *ACK* bit in the subsequent uplink. The test also verifies that when the **TCL** sends a retransmission with the same FCntDown, the DUT ignores the downlink.

2.4.2.b.i.Test Procedure Frame Sequence Chart



Step	Procedure		Frame Sequence	Test Purpose
		End Device - TCL	Frame	
1	DUT sends Unconfirmed frame FCntUp = n			
	The TCL sends Unconfirmed frames	+	CP-CMD RxAppCntReq FPort = 224 Payload = [0x]09	
2	DUT sends Unconfirmed frame FCntUp = n + 1	→	CP-CMD RxAppCntAns FPort = 224 Payload = [0x]09XXXX RxAppCnt = x	
	The TCL send Confirmed frame FCntDown = m	+	CP-CMD TxFramesCtrlReq FPort 224 Frame type = Confirmed Payload [0x]0702	TCL sends Confirmed frame
3	DUT sends Confirmed frame FCntUp = n + 2)	ACK Bit = True Note: The DUT may split this frame into first an empty Unconfirmed frame with ACK, followed by a Confirmed frame. This must be accepted as well.	
	The TCL sends Confirmed frame FCntDown = m + 1	←	Acknowledge CP-CMD TxFramesCtrlReq FPort 224 Frame type = Unconfirmed Payload [0x]0701	TCL sends Confirmed frame
4	DUT sends Unconfirmed frame FCntUp = n + 3	→	ACK Bit = True	
	The TCL sends Confirmed frames FCntDown = m + 2	←	CP-CMD TxFramesCtrlReq FPort 224 Frame type = No change Payload [0x]0700	
5	DUT sends Unconfirmed frame FCntUp = n + 4	→	ACK Bit = True	
	The TCL sends Confirmed frames FCntDown = m + 2	+	CP-CMD TxFramesCtrlReq FPort 224 Frame type = No change Payload [0x]0700	
6	DUT sends Unconfirmed frame FCntUp = n + 5	→	No acknowledgement is sent	DUT ignores the downlink with incorrect frame counter
	The TCL sends Unconfirmed frames FCntDown = m + 3	+	CP-CMD RxAppCntReq FPort = 224 Payload = [0x]09	
7	DUT sends Unconfirmed frame FCntUp = n + 6	→	CP-CMD RxAppCntAns FPort = 224 Payload = [0x]09XXXX RxAppCnt = x + 4	





LoRaW4N°



2.5. MAC Command Tests

The following tests will validate the **DUT**'s implementation of MAC command processing and the associated functional areas being controlled by the **TCL**. As previously stated, the **TCL** should allow for some reasonable amount of packet loss while facilitating tests. Specifically, where MAC commands are concerned, it is acceptable to retry commanding the device in the absence of a MAC command answer. Retries should be limited to a maximum of 5 attempts. After 5 uplinks are received from the **DUT**, if the **TCL** still does not get the expected response from the **DUT**, the test must fail, and the tool must move to the next test.

2.5.1. DevStatusReq

TCL sends a DevStatusReg command to the DUT.

620621 Verify that

610

611612

613

614

615 616

617

618

619

622

623

624

625

- The **DUT** must reply with a *DevStatusAns* packet.
- The signal-to-noise information in the RadioStatus field in the reply is a signed integer of 6 bits with a minimum value of -32 and maximum value of 31.

2.5.1.a. Test Procedure Frame Sequence Chart

Step	Procedure	Frame Sequence		Test
				Purpose
		End Device	Frame	
		- TCL		
1	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame	←	MAC-CMD DevStatusReq	
			Payload [0x]06	
2	DUT sends Unconfirmed frame	\rightarrow	MAC-CMD DevStatusAns	DevStatusAn
			RadioStatus >= - 32 and <=	s sent and
			31	encoded
			Payload [0x]06XXXX	value tested

626

627

628 629

630

631

2.5.2. NewChannelReq

2.5.2.a. Fixed Channel plan devices

For Fixed channel plan devices, the NewChannelReq MAC command must be rejected and the **DUT** must silently drop the request packet. The **DUT** must continue normal operation.



2.5.2.a.i. **Test Procedure Frame Sequence Chart**

Step	Procedure		Frame Sequence	Test Purpose
		End Device - TCL	Frame	
1	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame	+	MAC-CMD NewChannelReq ChIndex = 4 Freq = any applicable frequency, refer [2] Payload = [0x]07XXXXXXXXXX	
2	DUT sends Unconfirmed frame	→		No response to the command but DUT continues normal operation

633 2.5.2.b. For Dynamic Channel plan devices only 634

TCL sends a *NewChannelReq* command to the **DUT** for configuring new channel frequencies.

NewChannelReq command is supported for only Dynamic channel plan devices.

For Fixed channel plan devices, DUT must silently drop the request packet.

636 637

638

639

640

641

642

643

644

645

646

647

648

649

650

651

652

653 654

655

635

632

2.5.2.b.i. Read-only default channels

The mandatory default channels are defined in [2]. The TCL will send NewChannelReq that tries to modify these channels. The DUT must reply with an unsuccessful NewChannelAns NOT = ([0x]0703).

2.5.2.b.ii. Addition of a channel

TCL sends a MAC command to add a single new channel.

The **DUT** must reply with a successful NewChannelAns and begin using the new channel in its random selection of frequencies. This test may use any frequency applicable for that region [2].

2.5.2.b.iii. Removal of a channel

TCL sends NewChannelReq to set the previously configured channel to 0 MHz frequency.

The **DUT** must reply with a successful NewChannelAns and stop using the additional channel in its uplink transmissions.

2.5.2.b.iv. Addition and removal of multiple channels

TCL sends multiple MAC commands in a single frame to configure the additional channels. Multiple such frames may be sent to configure all the additional non-default channels and to remove them.





656	The DUT must	reply with a successful <i>NewChannelAns</i> to each request. The DUT must
657	send an uplink	c on each channel configured.
658	2.5.2.b.v.	Invalid command processing - Frequency
659	TCL sends Nev	vChannelReq command including an invalid frequency located out of the
660	band to check	that the DUT replies correctly.
661	2.5.2.b.vi.	Invalid command processing - Data Rate Range
662	TCL sends Nev	vChannelReq command including an invalid data rate range to check that
663	the DUT replie	es correctly.
664	2.5.2.b.vii.	Removal of default channels - not allowed
665	TCL tries to re	emove the default channels. DUT must reject the command and must
666	continue to us	se the default channels.
667		
668	2.5.2.b.viii.	Test Procedure Frame Sequence Chart
669		



Step	Procedure	ı	Frame Sequence	Test Purpose
		End Device - TCL	Frame	i uipose
1	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame	+	MAC-CMD NewChannelReq ChIndex = 0 Freq = Any allowed frequency for the channel, refer [2] DRRange = Any valid range, refer [2] Payload = [0x]07XXXXXXXXXX	TCL attempts to modify default channels
2	DUT sends Unconfirmed frame	→ R [All default Ch]	MAC-CMD NewChannelAns Payload NOT = [0x]0703	
	The TCL sends Unconfirmed frame	← R [All default Ch]	MAC-CMD NewChannelReq ChIndex = All other default channels, refer [2] Freq = Any allowed frequency for that channel, refer [2] DRRange = Any valid range, refer [2] Payload = [0x]07XXXXXXXXXX	
3	DUT sends Unconfirmed frame	→	MAC-CMD NewChannelAns Payload NOT = [0x]0703	DUT shall not change its channel plan or transmission behaviour
	The TCL sends Unconfirmed frame	←	FPort = 0 MAC-CMD NewChannelReq ChIndex = 15 Freq = any applicable frequency, refer [2] Payload = [0x]07XXXXXXXXXX	
4	DUT sends Unconfirmed frame	→	MAC-CMD NewChannelAns Payload = [0x]0703	
5	Wait for a maximum of (5 * number of channels configured) uplink packets, i.e. until the channel configured is used at least once	→ R [5*NbCh] or [AllCh used]	The new channel configured must be used at least once	New channel added
6	DUT sends Unconfirmed frame The TCL sends Unconfirmed frame	→ ←	MAC-CMD NewChannelReq ChIndex = 15 Freq = 0 MHz Payload = [0x]07XXXXXXXXXX	
7	DUT sends Unconfirmed frame	→	MAC-CMD NewChannelAns Payload = [0x]0703	



Step	Procedure	ı	Test Purpose	
		End Device - TCL	Frame	•
8	Wait for 5 * number of channels configured uplink packets, to confirm that the removed channel is not used.	→ R [5*NbCh]	The channel removed must not be used	
9	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame	+	FPort = 0	
			MAC-CMD NewChannelReq	
			ChIndex = all non-default	
			channel indexes, refer [2]	
			Freq = any frequency	
			applicable for that region,	
			refer [2]. Each channel must	
			have a different frequency,	
			as supported by the gateway.	
			Payload = [0x]07XXXXXXXXXXX[0x]07X XXXXXXXXXX [repeat [0x]07XXXXXXXXX up to (16 - NbDefaultChannels)]	
			Note1: This downlink may be split into multiple downlinks so that the maximum FRMPayload is not exceeded	
			Note2: When performing pre-testing using an 8-channel gateway for DC plan devices, the Freq field must be set to 8 different allowed frequencies for that region as supported by the gateway for the first 8 channels. The same set of 8 frequencies must be repeated for the next set of 8 channels as	



Step	Procedure		Frame Sequence	Test Purpose
		End Device - TCL	Frame	
10	DUT sends Unconfirmed frame	→	FPort = 0 MAC-CMD NewChannelAns Payload = [0x]0703[0x]0703[0x]0703 [Repeat [0x]0703 "Y" times]	16 channels (default + additional channels) configured
			where Y is the number of channels configured Note: This uplink may be split into multiple uplinks so that the maximum	
			FRMPayload is not exceeded	
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq ChMaskCntl = 0 ChMask = [0x]F000 Payload = [0x]03XXXXXXXX	
11	DUT sends Unconfirmed frame	→	MAC-CMD LinkADRAns Payload = [0x]0307	Enable channel IDs 12-15
12	Wait for 5 * 4 (number of channels enabled) = 20 uplink packets to be sent	→ R [5*NbCh]	Only the 4 enabled channels must be used at least once, and the disabled channels must not be used	Checking channel usage
13	DUT sends Unconfirmed frame	→		
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq ChMaskCntl = 0 ChMask = [0x]0F00 Payload = [0x]03XXXXXXXX	
14	DUT sends Unconfirmed frame	→	MAC-CMD LinkADRAns Payload = [0x]0307	Enable channel IDs 8-11
15	Wait for 5 * 4 (number of channels enabled) = 20 uplink packets to be sent	→ R [5*NbCh]	Only the 4 enabled channels must be used at least once, and the disabled channels must not be used	Checking channel usage
16	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq ChMaskCntl = 0 ChMask = [0x]00F0 Payload = [0x]03XXXXXXXX	
17	DUT sends Unconfirmed frame	→	MAC-CMD LinkADRAns Payload = [0x]0307	Enable channel IDs 4-7
18	Wait for 5 * 4 (number of channels enabled) = 20 uplink packets to be sent	→ R [5*NbCh]	Only the 4 enabled channels must be used at least once, and the disabled channels must not be used	Checking channel usage
19	DUT sends Unconfirmed frame	\rightarrow		



Step	Procedure	į į	Frame Sequence	Test Purpose
		End Device - TCL	Frame	ruipose
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq ChMaskCntl = 0 ChMask = [0x]000F Payload = [0x]03XXXXXXXX	
20	DUT sends Unconfirmed frame	→	MAC-CMD LinkADRAns Payload = [0x]0307	Enable channel IDs 0-3
21	Wait for 5 * 4 (number of channels enabled) = 20 uplink packets to be sent	→ R [5*NbCh]	Only the 4 enabled channels must be used at least once, and the disabled channels must not be used	Checking channel usage
22	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame	←	FPort = 0 MAC-CMD NewChannelReq	
			ChIndex = all non-default channel indexes, refer [2]	
			Freq = 0 MHz	
			Payload = [0x]07XXXXXXXXXX[0x]07X XXXXXXXXX [repeat	
			[0x]07XXXXXXXXXX for all channels removed]	
23	DUT sends Unconfirmed frame	→	FPort = 0 MAC-CMD NewChannelAns Payload = [0x]0703[0x]0703[0x]0703 [Repeat [0x]0703 "Y" times] – where Y is the number of channels removed	All additional Channels removed
24	Wait for 5 * (default number of channels) uplink packets to be sent	→ R [5*NbDefCh]	The additional channels which were removed must not be used and only the default channels must be used	Checking channel usage
25	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame	+	CMD NewChannelReq ChIndex = Any channel index other than the default, refer [2] Freq = Any invalid Freq, refer [2] Payload = [0x]07XXXXXXXXXX	
26	DUT sends Unconfirmed frame	→	CMD NewChannelAns Payload NOT = [0x]0703	DUT shall not modify its frequency or transmission behaviour



Step	Procedure	i	Frame Sequence	Test
			T	Purpose
		End Device	Frame	
		- TCL		
	The TCL sends Unconfirmed frame	←	MAC-CMD NewChannelReq	
			ChIndex = Any channel	
			index other than the default,	
			refer [2]	
			Freq = Default, refer [2]	
			DRRange = An invalid data	
			range, refer [2]	
			Payload = [0x]07XXXXXXXX	
27	DUT sends Unconfirmed frame	\rightarrow	MAC-CMD NewChannelAns	DUT shall
			Payload is NOT = [0x]0703	not add the
				channel due
				to invalid
				Data Range
	The TCL sends Unconfirmed frame	←	MAC-CMD NewChannelReq	
			ChIndex = All default	
			channels, refer [2]	
			Freq = 0 MHz	
			Payload =	
			[0x]07XXXXXXXX[Repeat	
			[0x]07XXXXXXXXX "Y" times]	
			 where Y is the number of 	
			default channels	
28	DUT sends Unconfirmed frame	\rightarrow	MAC-CMD NewChannelAns	Default
			Payload NOT	channels not
			=[0x]0703[Repeat "Y" times]	affected
			 – where Y is the number of 	
			default channels	
29	Wait for 5 * (number of channels	→ R	Only the default channels	Checking
	configured) uplink packets to be sent	[5*NbDefCh]	must be used at least once	channel
				usage

671 **2.5.3. DIChannelReq**

670

672

673674

2.5.3.a. Fixed Channel plan devices

For Fixed channel plan devices, the **DUT** must silently drop the DIChannelReq MAC command request. The **DUT** must continue normal operation.



2.5.3.a.i. Test Procedure Frame Sequence Chart

Step	Procedure	I	Frame Sequence	Test
				Purpose
		End Device	Frame	
		- TCL		
1	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame	←	MAC-CMD DIChannelReq	
			ChIndex = Any default	
			channel, refer [2]	
			Freq = Any frequency other	
			than default freq, refer [2])	
			Payload = [0x]0AXXXXXXXX	
2	DUT sends Unconfirmed frame	\rightarrow		No response
				to the
				command but
				DUT
				continues
				normal
				operation

676

677

678 679

680

681 682

683

684

685

675

2.5.3.b. **Dynamic Channel plan devices**

The **TCL** sends a MAC command to change the downlink frequency in RX1 for an existing channel. The **TCL** tests RX1 downlink window using the new frequency and then restores the default values. Retransmission is tested, as well. The **TCL** waits for an uplink, while the MAC answer should be [0x]0A03. Then the **TCL** sends a downlink and waits for the next uplink, while the MAC answer should not contain [0x]0A03. Finally, standard settings are applied and tested again. Additionally, the **TCL** sends *DIChannelReq* commands including invalid frequency and channel values to check that the **DUT** replies appropriately.

686 687

2.5.3.b.i. Test Procedure Frame Sequence Chart



Step	Procedure	ı	Test Purpose	
		End Device - TCL	Frame	
1	DUT sends Unconfirmed frame	→		
	The TCL sends Unconfirmed frame on RX1 window	←	MAC-CMD DIChannelReq ChIndex = C (where C = Any default channel, refer [2])	
			Freq = X (where X = any allowed frequency other than the default frequency, refer [2])	
		, _	Payload = [0x]0AXXXXXXXX	
2	DUT sends Unconfirmed frame	→ R [max 3]	MAC-CMD DIChannelAns Payload = [0x]0A03	
	Repeat up to 3 times until a downlink is received confirming the receipt of the DIChannelAns			
3	DUT sends Unconfirmed frame	\rightarrow	MAC-CMD DIChannelAns Payload = [0x]0A03	
	The TCL must send an Unconfirmed	←	CP-CMD EchoPayloadReq	Tests
	frame on RX1 window		FPort = 224 Payload = [0x]080A0B0C	downlink frequency settings for
			Chindex = Any one of the default channels, refer [2] Freq = The frequency set	the default channel
4	DUT sends Unconfirmed frame	→ R [5*NbDefCh]	If the EchoPayloadReq was sent in the previous	
	Repeat for a maximum of (5 * number of default channels) until the	OR [All default	downlink,	
	DUT sends an uplink on all default channels	channels]	CP-CMD EchoPayloadAns FPort = 224 Payload = [0x]080B0C0D	
	If the DUT sends an uplink on	← R	CP-CMD EchoPayloadReq	Tests
	another default channel, then the TCL sends Unconfirmed frame on RX1 window	[All default channels]	FPort = 224 Payload = [0x]080A0B0C	downlink frequency settings for
	Repeat for all default channels		ChIndex = Each of the remaining default channels, refer [2]	all default channels
5	DUT sends Unconfirmed frame	→	Freq = The frequency set CP-CMD EchoPayloadAns	Tests
J	DOT Serius Officerimmed frame		FPort = 224 Payload = [0x]080B0C0D	downlink frequency settings for the last default channel



Step	Procedure		Frame Sequence		
		End Device - TCL	Frame	Purpose	
	TCL sends Unconfirmed frame	+	MAC-CMD DIChannelReq ChIndex = C		
			Freq = default frequency, refer [2]		
			Payload = [0x]0AXXXXXXXX		
6	DUT sends Unconfirmed frame Repeat up to 3 times until a downlink is received confirming the receipt of the DIChannelAns	→ R [max 3]	MAC-CMD DIChannelAns Payload = [0x]0A03	DUT returns to its default settings	
7	DUT sends Unconfirmed frame	→	MAC-CMD DIChannelAns Payload = [0x]0A03		
	The TCL must send an Unconfirmed frame on RX1 window	+	CP-CMD EchoPayloadReq FPort = 224 Payload = [0x]080A0B0C ChIndex = Any one of the default channels, refer [2] Freq = Default frequency	Tests default frequency settings for the default channel	
8	DUT sends Unconfirmed frame Repeat for a maximum of (5 * number of default channels) until the DUT sends an uplink on all default channels	→ R [5*NbDefCh] OR [All default channels]	If the EchoPayloadReq was sent in the previous downlink, CP-CMD EchoPayloadAns FPort = 224		
	If the DUT sends an uplink on another default channel, then the TCL sends Unconfirmed frame on RX1 window Repeat for all default channels	← R [All default channels]	Payload = [0x]080B0C0D CP-CMD EchoPayloadReq FPort = 224 Payload = [0x]080A0B0C ChIndex = Each of the remaining default channels, refer [2]	Tests default frequency settings for all default channels	
9	DUT sends Unconfirmed frame	→	Freq = Default frequency CP-CMD EchoPayloadAns FPort = 224 Payload = [0x]080B0C0D	Tests default frequency settings for the last default channel	
	The TCL sends Unconfirmed frame	+	MAC-CMD DIChannelReq ChIndex = C Freq = Any invalid frequency, refer [2] Payload = [0x]0AXXXXXXXX		
10	DUT sends Unconfirmed frame Repeat up to 3 times until a downlink is received confirming the receipt of the DIChannelAns	→ R [max 3]	MAC-CMD DIChannelAns Payload is NOT = [0x]0A03	Unsuccessfu I for invalid frequency	



Step	Procedure		Frame Sequence	Test
				Purpose
		End Device	Frame	
		- TCL		
11	DUT sends Unconfirmed frame	\rightarrow	MAC-CMD DIChannelAns	
			Payload is NOT = [0x]0A03	
	The TCL must wait until the DUT	←	MAC-CMD DIChannelReq	
	sends an uplink on the same channel		ChIndex = Any channel not	
	configured in Step 4 and then sends		configured, refer [2]	
	Unconfirmed frame			
			Freq = default frequency	
			Payload = [0x]0AXXXXXXXX	
12	DUT sends Unconfirmed frame	→ R	MAC-CMD DIChannelAns	Unsuccessfu
		[max 3]	Payload is NOT = [0x]0A03	I for channel
	Repeat up to 3 times until a downlink			not
	is received confirming the receipt of			configured
	the DIChannelAns			
13	DUT sends Unconfirmed frame	\rightarrow	MAC-CMD DIChannelAns	
			Payload is NOT = [0x]0A03	
	The TCL sends Unconfirmed frame	-	CP-CMD TxFramesCtrlReq	
			FPort = 224	
			Frame type = No change	
			Payload = [0x]0700	
14	DUT sends Unconfirmed frame	\rightarrow		

689

690

691

692

693

694 695

696

697

2.5.4. RXParamSetupReq

The **TCL** sends a MAC command to configure new *RX1DRoffset*, *RX2DataRate* and *Frequency*. The **DUT** must include a successful *RXParamSetupAns* in all subsequent uplinks until a downlink is received. The **TCL** will wait for more than one uplink containing the affirmative *RXParamSetupAns* of [0x]0507. The **TCL** then sends a downlink and verifies the next uplink does not contain the *RXParamSetupAns* [0x]0507. Both the RX1 and RX2 downlink windows are then tested using the new parameters. Finally, default settings are restored by way of MAC command and both RX windows tested again.

The **TCL** sends a MAC command with invalid RX2DRoffset, RX2DataRate and Frequency that is not supported. Verify that the **DUT** ignores the command and the previous parameters are kept.

2.5.4.a. Test Procedure Frame Sequence Chart



Step	Procedure	I	Frame Sequence	Test Purpose
		End Device - TCL	Frame	Furpose
1	DUT sends Unconfirmed frame FCntUp = y + 1	→		
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq TXPower = Maximum, refer [2] DataRate = Max125kHzDR, refer [2] ChMaskCntl = 0 (for DC) and 6 (for FC) ChMask = Enable only the default channels for DC, refer [2] and [0x]00FF for FC NbTrans = 1 Payload = [0x]03XXXXXXXXX	
2	DUT sends Unconfirmed frame FCntUp = y + 2)	MAC-CMD LinkADRAns DataRate = X Payload = [0x]0307	DataRate set to value configured
	The TCL sends Unconfirmed frame	+	MAC-CMD RxParamSetupReq RX1DRoffset = any allowed offset value, refer [2] RX2DataRate = Any DataRate allowed except the one set in Step 2 Frequency = Y (where Y = any frequency allowed, refer [2]) Payload = [0x]05XXXXXXXX	
3	DUT sends Unconfirmed frame Repeat up to 3 times until a downlink is received confirming the receipt of the RxParamSetupAns	→ R [max 3]	MAC-CMD RxParamSetupAns Payload = [0x]0507	
4	FCntUp >= y + 3 DUT sends Unconfirmed frame	→	MAC-CMD RxParamSetupAns Payload = [0x]0507	
	The TCL sends Unconfirmed frame on RX1 window	+	CP-CMD EchoPayloadReq FPort = 224 DR = X - RX1DRoffset Payload = [0x]08010203	
5	DUT sends Unconfirmed frame	→	CP-CMD EchoPayloadAns FPort = 224 Payload = [0x]08020304	Tested for new parameters on RX1
	The TCL sends Unconfirmed frame on RX2 window	←	CP-CMD EchoPayloadReq FPort = 224 Payload = [0x]08121314 Freq = Y DataRate = RX2DataRate as set	



Step	Procedure		Frame Sequence	Test Purpose
		End Device - TCL	Frame	Purpose
6	DUT sends Unconfirmed frame	→	CP-CMD EchoPayloadAns FPort = 224 Payload = [0x]08131415	Tested for new parameters on RX2
	The TCL sends Unconfirmed frame	+	MAC-CMD RxParamSetupReq RX1DRoffset = 0 RX2DataRate = Default Frequency = Default Payload = [0x]05XXXXXXXX The default values are defined in [2]	
7	DUT sends Unconfirmed frame	→	MAC-CMD RxParamSetupAns Payload = [0x]0507	Restored to default settings
	The TCL sends Unconfirmed frame on RX1 window	+	CP-CMD EchoPayloadReq FPort = 224 Payload = [0x]08010203	
8	DUT sends Unconfirmed frame	→	CP-CMD EchoPayloadAns FPort = 224 RX1DRoffset = 0 Payload = [0x]08020304	Default Rx1 Params verified
	The TCL sends Unconfirmed frame on RX2 window	+	CP-CMD EchoPayloadReq RX2DataRate = Default Frequency = Default FPort = 224 Payload = [0x]08121314	
9	DUT sends Unconfirmed frame	→	CP-CMD EchoPayloadAns FPort = 224 Payload = [0x]08131415	Default Rx2 Params verified
	The TCL sends Unconfirmed frame	+	MAC-CMD RxParamSetupReq RX1DRoffset = Invalid, refer [2] RX2DataRate = default Frequency = default Payload = [0x]05XXXXXXXX	Invalid RX1DRoffset
10	DUT sends Unconfirmed frame	→	MAC-CMD RxParamSetupAns Payload is NOT = [0x]0507	DUT confirms that the parameters were not set
	The TCL sends Unconfirmed frame	+	MAC-CMD RxParamSetupReq RX2DataRate = Invalid, refer [2] RX1DROffset = default Frequency = default Payload = [0x]05XXXXXXXX	Invalid RX2DataRat e



Step	Procedure	Frame Sequence		Test
				Purpose
		End Device	Frame	
		- TCL		
11	DUT sends Unconfirmed frame	\rightarrow	MAC-CMD	DUT
			RxParamSetupAns	confirms that
			Payload is NOT = [0x]0507	the
				parameters
				were not set
	The TCL sends Unconfirmed frame	←	MAC-CMD	Invalid Freq
			RxParamSetupReq	
			RX1DROffset = default	
			RX2DataRate = default	
			Frequency = Invalid, refer [2]	
			Payload = [0x]05XXXXXXXX	
12	DUT sends Unconfirmed frame	\rightarrow	MAC-CMD	DUT
			RxParamSetupAns	confirms that
			Payload is NOT = [0x]0507	the
				parameters
				were not set
	The TCL sends Unconfirmed frame	+	CP-CMD EchoPayloadReq	
	on RX1 window		FPort = 224	
			Payload = [0x]08010203	
13	DUT sends Unconfirmed frame	\rightarrow	CP-CMD EchoPayloadAns	Default Rx1
			FPort = 224	Params
			Payload = [0x]08020304	verified
	The TCL sends Unconfirmed frame	+	CP-CMD EchoPayloadReq	
	on RX2 window		FPort = 224	
			Payload = [0x]08121314	
14	DUT sends Unconfirmed frame	\rightarrow	CP-CMD EchoPayloadAns	Default Rx2
			FPort = 224	Params
			Payload = [0x]08131415	verified

2.5.5. RXTimingSetupReq

The **TCL** sends a MAC command to change the timing of the reception windows. The **DUT** must reply with [0x]08 (and no *RXTimingSetupAns* payload). The **TCL** tests RX1 and RX2 downlink windows using the new parameters.

Additionally, retransmission is tested. The **TCL** sends a MAC command to change the timing of the reception windows. The **TCL** waits for an uplink with *RXTimingSetupAns* [0x]08. The **TCL** does not send a downlink and instead waits for subsequent uplinks that must contain the *RXTimingSetupAns*. Once satisfied, the **TCL** sends a downlink and waits for the next uplink to ensure it does not contain the *RXTimingSetupAns* [0x]08. Finally, standard settings are applied and tested again with Echo command.

2.5.5.a. **Test Procedure Frame Sequence Chart**

710 711

700

701

702

703

704

705

706

707

708



Step	Procedure		Test Purpose	
		End Device - TCL	Frame	i dipose
1	DUT sends Unconfirmed frame FCntUp = y	→		
	The TCL sends Unconfirmed frame	+	MAC-CMD RxTimingSetupReq Payload = [0x]08XX Delay (i) = [3-14]	
2	DUT sends Unconfirmed frame Repeat up to 3 times until a downlink is received confirming the receipt of the RxTimingSetupAns	→ R [max 3]	MAC-CMD RxTimingSetupAns Payload = [0x]08	
	FCntUp >= y + n	→	MAC CMP	
3	DUT sends Unconfirmed frame	-	MAC-CMD RxTimingSetupAns Payload = [0x]08	
	The TCL sends Unconfirmed frame on RX1 window	←	CP-CMD EchoPayloadReq FPort = 224 Payload = [0x]08010203	
			TXDelay = (i) seconds	
4	DUT sends Unconfirmed frame	→	CP-CMD EchoPayloadAns FPort = 224 Payload = [0x]08020304	Tested for new parameters on RX1
	The TCL sends Unconfirmed frame on RX2 window	+	CP-CMD EchoPayloadReq FPort = 224 Payload = [0x]08121314	
			TXDelay = (i + 1) seconds	
5	DUT sends Unconfirmed frame	→	CP-CMD EchoPayloadAns FPort = 224 Payload = [0x]08131415	Tested for new parameters on RX2
	The TCL sends Unconfirmed frame	(MAC-CMD RxTimingSetupReq Payload = [0x]08XX Delay = 2	Setting Delay = 2
6	DUT sends Unconfirmed frame Repeat up to 3 times until a downlink is received confirming the receipt of the RxTimingSetupAns	→ R [max 3]	MAC-CMD RxTimingSetupAns Payload = [0x]08	
7	DUT sends Unconfirmed frame	→	MAC-CMD RxTimingSetupAns Payload = [0x]08	
	The TCL sends Unconfirmed frame on RX1 window	+	CP-CMD EchoPayloadReq FPort = 224 Payload = [0x]08010203	
			TXDelay = 2 sec	





8 DUT sends Unconfirmed frame	adAns Tested for
	dolov
Doubsed [0:100000	delay
Payload = [0x]080203	304 settings on RX1
The TCL sends Unconfirmed frame ← CP-CMD EchoPayloa	
on RX2 window FPort = 224	aartoq
Payload = [0x]081213	314
T ayload = [0x]001211	314
TXDelay = 3 seconds	S
9 DUT sends Unconfirmed frame → CP-CMD EchoPayloa	
FPort = 224	delay
Payload = [0x]081314	415 settings on
	RX2
The TCL sends Unconfirmed frame ← MAC-CMD	Setting
RxTimingSetupReq	Delay = 15
Payload = [0x]08XX	
Delay = 15	
10 DUT sends Unconfirmed frame → R MAC-CMD	
Repeat up to 3 times until a downlink [max 3] RxTimingSetupAns	
is received confirming the receipt of Payload = [0x]08	
the RxTimingSetupAns	
11 DUT sends Unconfirmed frame → MAC-CMD	
RxTimingSetupAns	
Payload = [0x]08	
The TCL sends Unconfirmed frame ← CP-CMD EchoPayloa	adReq
on RX1 window FPort = 224	
Payload = [0x]080103	203
TXDelay = 15 sec	
12 DUT sends Unconfirmed frame → CP-CMD EchoPayloa	
FPort = 224	delay
Payload = [0x]080203	_
	RX1
The TCL sends Unconfirmed frame ← CP-CMD EchoPayloa	adReq
on RX2 window FPort = 224	
Payload = [0x]081213	314
TXDelay = 16 second	de
13 DUT sends Unconfirmed frame → R CP-CMD EchoPayloa	
FPort = 224	delay
Payload = [0x]08131-	
r ayıoad = [0x]061314	RX2
The TCL sends Unconfirmed frame ← MAC-CMD	Setting
RxTimingSetupReq	Delay to
Payload = [0x]08XX	default = 0
Delay = 0	
14 DUT sends Unconfirmed frame → R MAC-CMD	
Repeat up to 3 times until a downlink [max 3] RxTimingSetupAns	
is received confirming the receipt of Payload = [0x]08	
the RxTimingSetupAns	
15 DUT sends Unconfirmed frame → MAC-CMD	
RxTimingSetupAns	
	1



	The TCL sends Unconfirmed frame on RX1 window	←	CP-CMD EchoPayloadReq FPort = 224 Payload = [0x]08010203 TXDelay = 1 sec	
16	DUT sends Unconfirmed frame	÷	CP-CMD EchoPayloadAns FPort = 224 Payload = [0x]08020304	Tested for delay settings on RX1
17	The TCL sends Unconfirmed frame on RX2 window	←	CP-CMD EchoPayloadReq FPort = 224 Payload = [0x]08121314 TXDelay = 2 seconds	
18	DUT sends Unconfirmed frame	→	CP-CMD EchoPayloadAns FPort = 224 Payload = [0x]08131415	Tested for delay settings on RX2

713

717

2.5.6. TXP aram Setup Req MAC command

- Refer to [2] to obtain the list of regions for which the TXParamSetupReq command is applicable. If the
- 715 TXParamSetupReq command is not applicable for the region being tested, the **DUT** must continue
- 716 normal operation after receiving the TXParamSetupReq command hence ignoring the command.

2.5.6.a. **Test Procedure Frame Sequence Chart**

Step	Procedure		Frame Sequence	Test
				Purpose
		End Device	Frame	
		- TCL		
1	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame	+	MAC-CMD	
			TXParamSetupReq	
			Payload = [0x]09XX	
			UplinkDwellTime = 0	
2	DUT sends Unconfirmed frame	\rightarrow		No response
				to the
				command but
				DUT
				continues
				normal
				operation

718 719

720

721

722723

724

725

726

If the TXParamSetupReq command is applicable for the region being tested, the following tests must be performed.

- 1. No dwell time setting: The **TCL** sends *LinkADRReq* to set the **DUT** to each valid DR which is not in the Join-Request Data Rate range, refer [2]. **DUT** must reply with an unsuccessful *LinkADRAns* and the **TCL** verifies that no packet is received using this data rate.
- 2. *UplinkDwellTime* set to 0 (unlimited): The **TCL** sends a *TXParamSetupReq* to set *UplinkDwellTime* to 0 (unlimited) and the **DUT** must reply with a *TXParamSetupAns*. Then, the **TCL** sends *LinkADRReq* to set the **DUT** to the Minimum Data Rate up to the Maximum Data Rate using 125kHz, refer [2].





- The **DUT** must reply to each request with a successful *LinkADRAns*. The **TCL** verifies that the uplink data rate is as requested.
- 3. *UplinkDwellTime* set to 1 (400ms): The **TCL** sends a *TXParamSetupReq* to set *UplinkDwellTime* to 1 (400ms) and the **DUT** must reply with a *TXParamSetupAns*. Then, the **TCL** sends *LinkADRReq* to set the **DUT** to each DR which is in the invalid Data Rate range, refer [2]. The **DUT** must reply with an unsuccessful *LinkADRAns* and the server verifies that no packet is received using this data rate. For the data rates in the Join-Request Data Range, the **DUT** must reply to each request with a successful *LinkADRAns*. The **TCL** verifies that the uplink data rate is as requested.
- 735 4. MaxEIRP setting:

- a. Set MaxEIRP = Highest: Uplink signal level is monitored on the gateway side (1).
- b. Set MaxEIRP = Lowest: Uplink signal level is monitored on the gateway side (2).
- 738 c. Check that (1) (2) is greater than 6 dB
- 739 2.5.6.b. Test Procedure Frame Sequence Chart- if TXParamSetupReq command is
 740 applicable for the region tested



Step	Procedure	I	Frame Sequence	Test Purpose
		End Device - TCL	Frame	
1	DUT sends Unconfirmed frame	\rightarrow		
ļ	The TCL sends Unconfirmed frame	←	MAC-CMD LinkADRReq	
			DataRate = The first DR	
ļ			which is not in the Join-	
ļ			Request Data Rate range,	
ļ			refer [2]	
			Payload = [0x]03XXXXXXXX	
2	DUT sends Unconfirmed frame	→ R	MAC-CMD LinkADRAns	
ļ		[All DR not	Payload NOT = [0x]0307	
ļ		in JR range]		
ļ	The TCL sends Unconfirmed frame	← R	MAC-CMD LinkADRReq	
ļ		[All DR not	DataRate = All other DR	
ļ		in JR range]	which are not in the Join-	
			Request Data Rate range,	
ļ			refer [2]	
			Payload = [0x]03XXXXXXXX	
3	DUT sends Unconfirmed frame	\rightarrow	MAC-CMD LinkADRAns	LinkADRAns
			Payload NOT = [0x]0307	Status = Unsuccessful
ļ			Note: This response is for	
ļ			the last LinkADRReq with	
ļ			DR not in the JR range	
ļ	The TCL sends Unconfirmed frame	+	MAC-CMD	
			TXParamSetupReq	
ļ			UplinkDwellTime = 0	
			Payload = [0x]09XX	
4	DUT sends Unconfirmed frame	\rightarrow	MAC-CMD	Uplink Dwell
ļ			TXParamSetupAns	Time set to 0
ļ			Payload = [0x]09	
ļ	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq	
			DataRate = MinDR, refer [2]	
	BUT III (, ,	Payload = [0x]03XXXXXXXX	5 . 5
5	DUT sends Unconfirmed frame	→ R	MAC-CMD LinkADRAns	DataRate set
ļ		[All DR <	Payload = [0x]0307	as required
ļ		Max125kHz	DataRate = X	
	The TCL sends Unconfirmed frame	DR] ← R	MAC CMD LinkADDDog	
ļ	The TCL sends Unconfirmed frame	[All DR up to	MAC-CMD LinkADRReq	
ļ	Repeat this step for each DR up to	Max125kHz	DataRate = Each other allowed DR up to	
	the maximum DR	DR]	Max125kHzDR, refer [2]	
ļ	THE MAXIMUM DK	DKJ	Payload = [0x]03XXXXXXXX	
6	DUT sends Unconfirmed frame	→	MAC-CMD LinkADRAns	DataRate set
o l	Do i serius oriconiimieu mame		Payload = [0x]0307	as required
ļ			DataRate = Max125kHzDR,	as required
			refer [2]	
			Note: This response is for	
			the last LinkADRReq with	
	İ	I		1
			May125kHzDR	
Testing	g UplinkDwellTime = 1		Max125kHzDR.	



Step	Procedure	1	Frame Sequence	Test Purpose
		End Device - TCL	Frame	
	The TCL sends Unconfirmed frame	+	MAC-CMD TXParamSetupReq UplinkDwellTime = 1 (400ms) Payload = [0x]09XX	
8	DUT sends Unconfirmed frame	→	MAC-CMD TXParamSetupAns Payload = [0x]09	Uplink Dwell Time set to 1 (400ms)
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq DataRate = The first DR in the invalid Data Rate range, refer [2] Payload = [0x]03XXXXXXXX	
9	DUT sends Unconfirmed frame	→ R [All invalid data rate range]	MAC-CMD LinkADRAns Payload NOT = [0x]0307	
	The TCL sends Unconfirmed frame	← R [All invalid data rate range]	MAC-CMD LinkADRReq DataRate = Each other DR which is in the invalid Data Rate range, refer [2] Payload = [0x]03XXXXXXXX	
10	DUT sends Unconfirmed frame	→	MAC-CMD LinkADRAns Payload NOT = [0x]0307	LinkADRAns Status = Unsuccessfu
	The TCL sends Unconfirmed frame	←	MAC-CMD LinkADRReq Data Rate = The first DR in the Join-Request Data Rate range, refer [2]) Payload = [0x]03XXXXXXXX	
11	DUT sends Unconfirmed frame	→ R [All DR in the JR range]	MAC-CMD LinkADRAns Payload = [0x]0307 DataRate = Y	
	The TCL sends Unconfirmed frame Repeat this step until DataRate reaches the maximum data rate in the Join-Request DataRange, refer [2]	← R [All DR in JR range]	MAC-CMD LinkADRReq Data Rate = Y (where Y = Each other DR in the Join- Request Data Rate range, refer [2]) Payload = [0x]03XXXXXXXX	
12	DUT sends Unconfirmed frame	→	MAC-CMD LinkADRAns Payload = [0x]0307 DataRate = Y	DataRate set to Y
MaxEl	RP testing			
13	DUT sends Unconfirmed frame The TCL sends Unconfirmed frame	→ ←	MAC-CMD TXParamSetupReq Max EIRP = Highest, refer [2] Payload = [0x]09XX	



Step	Procedure		Frame Sequence	Test Purpose
		End Device - TCL	Frame	. u.pess
14	DUT sends Unconfirmed frame	→	MAC-CMD TXParamSetupAns Payload = [0x]09	Max EIRP set to max value
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq TXPower = Maximum, refer [2] Payload = [0x]03XXXXXXXX	
15	DUT sends Unconfirmed frame	\rightarrow	MAC-CMD LinkADRAns Payload = [0x]0307	TXPower set to max value
16	DUT sends Unconfirmed frame	\rightarrow	Get RSSI value	
17	DUT sends Unconfirmed frame	\rightarrow	Get RSSI value	
18	DUT sends Unconfirmed frame	→	Get RSSI value X = Avg of last 3 RSSI value	RSSI value avg checked
	The TCL sends Unconfirmed frame	←	MAC-CMD TXParamSetupReq Max EIRP = Lowest, refer [2] Payload = [0x]09XX	
19	DUT sends Unconfirmed frame	→	MAC-CMD TXParamSetupAns Payload = [0x]09	Max EIRP set to min value
	The TCL sends Unconfirmed frame	←	MAC-CMD LinkADRReq TXPower = Maximum, refer [2] Payload = [0x]03XXXXXXXX	
20	DUT sends Unconfirmed frame	→	MAC-CMD LinkADRAns Payload = [0x]0307	TXPower set to max value
21	DUT sends Unconfirmed frame	→	Get RSSI value	to max value
22	DUT sends Unconfirmed frame	→	Get RSSI value	
23	DUT sends Unconfirmed frame	\rightarrow	Get RSSI value	RSSI value avg checked
			Y = Avg of last 3 RSSI value $Confirm: X - Y > 6 dB$	Difference of RSSI values checked
	The TCL sends Unconfirmed frame	←	MAC-CMD LinkADRReq DataRate = Max125kHzDR, refer [2] MAC-CMD TXParamSetupReq Max EIRP = Highest, refer [2] UplinkDwellTime = default, refer [2] Payload = [0x]03XXXXXXXX [0x]09XX	



Step	Procedure	Frame Sequence		Test Purpose
		End Device - TCL	Frame	
24	DUT sends Unconfirmed frame	→	MAC-CMD LinkADRAns MAC-CMD TXParamSetupAns	DUT reverted to default settings
			Payload = [0x]0307[0x]09	

743

744

2.5.7. LinkCheckReq tests

After the **TCL** triggers a LinkCheckReq, the **DUT** must send a LinkCheckReq frame. The **TCL** responds with a LinkCheckAns frame.

745746747

Verify

• After receiving the LinkCheckAns, the **DUT** continues normal operation and sends uplinks.

748 749

750

2.5.7.a. **Test Procedure Frame Sequence Chart**

Step	Procedure		Frame Sequence	Test Purpose
		End Device - TCL	Frame	
1	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame	+	CP-CMD LinkCheckReq FPort 224 Payload [0x]20	
2	DUT sends Unconfirmed frame	→	MAC-CMD LinkCheckReq Payload = [0x]02	
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkCheckAns Payload = [0x]02XXXX	
3	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame	+	CP-CMD EchoPayloadReq FPort 224 Payload [0x]08010203	
4	DUT sends Unconfirmed frame	→	CP-CMD EchoPayloadAns FPort = 224 Payload [0x]08020304	DUT responds normally afte LinkCheck

751

752

2.5.8. LinkADRReq

The following tests validate each aspect of the *LinkADRReq* command.

753754

755

756 757

2.5.8.a. **TXPower**

This test validates the **DUT** handles the TXPower indicated in the *LinkADRReq* command as an upper-limit (i.e. allowed maximum).





The **TCL** separately tests *LinkADRReq* commands with TXPower within and outside the range specified in [2]. As part of testing values within the valid range, the **TCL** will command the device to its minimum and maximum allowable TX power levels. TCL also checks for a TXPower value of [0x]F.

Verify

758

759

760 761

762

763

764

765

766 767

768

769 770

771

772

773774

775

776

777

- When commanded to a valid TX power level lower than it is capable of, the **DUT** responds with an unsuccessful LinkADRAns and operates at its previously configured TX power
- When commanded to a valid TX power level greater than it is capable of, the DUT responds with a successful LinkADRAns and operates at its maximum TX power
- The **DUT** responds with a successful *LinkADRAns* for those commands whose TXPower is within the specified range. The **TCL** will monitor the RSSI reported by the gateway and verify a minimum difference of at least 6 dB between these settings.
- Commands whose TX power is outside the range or [0x]F, the **DUT** responds with an unsuccessful *LinkADRAns* and keep the current parameter values.

Note: The ChMaskCntl and ChMask for Dynamic channel (DC) and Fixed channel (FC) plan devices are different. The difference is specified in the Sequence charts below.

2.5.8.a.i. Test Procedure Frame Sequence Chart



Step	Procedure	ı	Frame Sequence	Test
		End Device	Frame	Purpose
		- TCL	Traino	
1	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame	←	MAC-CMD LinkADRReq	
			TXPower = Minimum, refer	
			[2]	
			DataRate = Max125kHzDR,	
			refer [2]	
			ChMaskCntl = 0 (for DC) and	
			6 (for FC) ChMask = Enable only the	
			default channels for DC,	
			refer [2] and [0x]00FF for FC	
			NbTrans = 1	
			Payload = [0x]03XXXXXXXX	
2	DUT sends Unconfirmed frame	\rightarrow	MAC-CMD LinkADRAns	Command is
				rejected if
			If minimum TXPower the	minimum
			DUT is capable of is greater	TXPower the
			than the minimum allowed	DUT is
			TXPower as refer [2], then PowerACK = false	capable of is
			Payload NOT = [0x]0307 and	greater than the minimum
			DUT retains its previous	allowed
			TXPower	TXPower
				else the
			Else, PowerACK = true	command is
			Payload = [0x]0307	accepted
3	If in Step 2, the minimum TXPower	\rightarrow		
	the DUT is capable of is greater than			
	the minimum allowed TXPower, then perform Steps 3 and 4. Else, skip to			
	the step 5.			
	the stop o.			
	DUT sends Unconfirmed frame	_		
	The TCL sends Unconfirmed frame	←	MAC-CMD LinkADRReq TXPower = Minimum	
			supported by DUT, as	
			mentioned in the Certification	
			Questionnaire.	
			DataRate = Max125kHzDR,	
			refer [2]	
			ChMaskCntl = 0 (for DC) and	
			6 (for FC)	
			ChMask = Enable only the	
			default channels for DC,	
			refer [2] and [0x]00FF (for FC)	
			NbTrans = 1	
			Payload = [0x]03XXXXXXXX	
4	DUT sends Unconfirmed frame	\rightarrow	MAC-CMD LinkADRAns	
			PowerACK = true	
			Payload = [0x]0307	
5	DUT sends Unconfirmed frame	→	Get the RSSI value	
6	DUT sends Unconfirmed frame	\rightarrow	Get the RSSI value	



Step	Procedure		Frame Sequence	Test Purpose
		End Device - TCL	Frame	
7	DUT sends Unconfirmed frame	→	Get the RSSI value	Avg RSSI checked
			X = Avg of last 3 RSSI value	
	The TCL sends Unconfirmed frame	÷	MAC-CMD LinkADRReq TXPower = Maximum, refer [2] DataRate = Max125kHzDR for the region, refer [2] ChMaskCntl = 0 (for DC) and 6 (for FC) ChMask = Enable only the default channels for DC, refer [2] and [0x]00FF for FC NbTrans = 1 Payload = [0x]03XXXXXXXXX	
8	DUT sends Unconfirmed frame	→	MAC-CMD LinkADRAns PowerACK = true DataRateACK = true ChannelMaskACK = true Payload = [0x]0307	Command accepted
9	DUT sends Unconfirmed frame	\rightarrow	Get the RSSI value	
10	DUT sends Unconfirmed frame	→	Get the RSSI value	
11	DUT sends Unconfirmed frame	→	Get the RSSI value Y = Avg of last 3 RSSI value Confirm: X – Y > 6dB	RSSI value avg checked Difference of RSSI values checked
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq TXPower = RFU, refer [2] DataRate = Max125kHzDR, refer [2] ChMaskCntl = 0 (for DC) and 6 (for FC) ChMask = Enable only the default channels for DC, refer [2] and [0x]00FF for FC NbTrans = 1 Payload = [0x]03XXXXXXXX Note: For US902-928, this step must be skipped as the TXPower cannot be set to RFU	
12	DUT sends Unconfirmed frame)	MAC-CMD LinkADRAns Payload NOT = [0x]0307 Note: For US902-928, this step must be skipped as the TXPower cannot be set to RFU	Command rejected



Step	p Procedure Frame Sequence		Test Purpose	
		End Device - TCL	Frame	i urposc
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq TXPower = [0x]F DataRate = Max125kHzDR, refer [2] ChMaskCntl = 0 (for DC) and 6 (for FC) ChMask = Enable only the default channels for DC, refer [2] and [0x]00FF for FC NbTrans = 1 Payload = [0x]03XXXXXXXXX	
13	DUT sends Unconfirmed frame	→	MAC-CMD LinkADRAns PowerACK = true DataRateACK = true ChannelMaskACK = true Payload = [0x]0307	Command accepted
14	DUT sends Unconfirmed frame	\rightarrow	Get the RSSI value	
15	DUT sends Unconfirmed frame	\rightarrow	Get the RSSI value	
16	DUT sends Unconfirmed frame	→	Get the RSSI value Z = Avg of last 3 RSSI value Confirm: Y = Z ± 3dB There must be no change to the TXPower and it must be the same as set earlier.	RSSI value avg checked No change to TXPower
	The TCL sends Unconfirmed frame	←	MAC-CMD LinkADRReq TXPower = Maximum, refer [2] DataRate = Max125kHzDR, refer [2] ChMaskCntl = 0 (for DC) and 6 (for FC) ChMask = Enable only the default channels for DC, refer [2] and [0x]00FF for FC NbTrans = 1 Payload = [0x]03XXXXXXXXX	
17	DUT sends Unconfirmed frame	→	MAC-CMD LinkADRAns Payload = [0x]0307	DUT reverted to default settings

779 **2.5.8.b.i. For Dynamic channel plan devices -**780 This section is applicable for only Dynamic channel plan devices.

2.5.8.b.i.1. Unsupported data rates

The **TCL** sends MAC commands to change the **DUT** to an unsupported data rate, refer [2].

2.5.8.b. **Uplink Channel Management**

778

781

782





784	The DUT must reply with an unsuccessful <i>LinkADRAns</i> and the uplink data rate must
785	not change.
786	Verify
787	 The DUT replies to each valid request with an unsuccessful <i>LinkADRAns</i>.
788	 The uplink data rate does not change.





2.5.8.b.i.1.1. Test Procedure Frame Sequence Chart



Step	Procedure		Test Purpose	
		End Device - TCL	Frame	
1	DUT sends Unconfirmed frame FCntUp = n	→		
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq TXPower = Maximum DataRate = Unsupported data rate, refer [2] ChMaskCntl = 0 ChMask - Enable only the default channels, refer [2] NbTrans = 1 Payload = [0x]03XXXXXXXX	
2	DUT sends Unconfirmed frame FCntUp = n	→R	MAC-CMD LinkADRAns Payload NOT = [0x]0307	Request rejected
	The TCL sends Unconfirmed frame	← R [All optional data rates]	MAC-CMD LinkADRReq TXPower = Maximum DataRate = An optional data rate except default data rate, refer [2] ChMaskCntl = 0 ChMask - Enable only the default channels, refer [2] NbTrans = 1 Payload = [0x]03XXXXXXXX Note: Repeat for all optional data rates.	
3	DUT sends Unconfirmed frame FCntUp = n + 1	→	MAC-CMD LinkADRAns Payload NOT = [0x]0307	Request rejected
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq TXPower = Maximum DataRate = [0x]F ChMaskCntl = 0 ChMask - Enable only the default channels, refer [2] NbTrans = 1 Payload = [0x]03XXXXXXXX	,
4	DUT sends Unconfirmed frame FCntUp = n + 2	→	MAC-CMD LinkADRAns Payload = [0x]0307	Request accepted and no change to DR
	The TCL sends Unconfirmed frame	←	MAC-CMD LinkADRReq TXPower = Maximum, refer [2] DataRate = Max125kHzDR, refer [2] ChMaskCntl = 0 ChMask - Enable only the default channels, refer [2] NbTrans = 1 Payload = [0x]03XXXXXXXXX	



5	DUT sends Unconfirmed frame	\rightarrow	MAC-CMD LinkADRAns	DUT
	FCntUp = n + 3		Payload = [0x]0307	reverted to
				default
				settings

792 793

794 795

796 797

798

799 800

801

802 803

804 805

806

807

808 809 810 2.5.8.b.i.2. **ChannelMask functionality & Disable all Channels**

The TCL sends NewChannelReq command to configure a new channel and LinkADRReq to disable that channel using the channel mask. The TCL verifies that no transmissions are sent on the new channel. The TCL then sends a LinkADRReq to enable the new channel. The TCL verifies that the DUT begins using the new channel in a random sequence.

Finally, the additional channel is removed.

Disable all Channels

Lastly, the TCL tries to disable all channels by sending a LinkADRReq command with channel mask control index of 0 and a channel mask of [0x]0000. The **DUT** must reply with an unsuccessful LinkADRAns, (i.e. not [0x]0307).

The TCL verifies that DUT still uses the default uplink channels in random sequence at an appropriate data rate.

2.5.8.b.i.2.1. Test Procedure Frame Sequence Chart for Channel Mask and Disable all Channels



Step	Procedure		Frame Sequence	Test Purpose
		End Device - TCL	Frame	
1	DUT sends Unconfirmed frame FCntUp = n	→		
	The TCL sends Unconfirmed frame	+	MAC-CMD NewChannelReq ChIndex = X (where X = any unused optional channel index) Freq = Any allowed frequency except the default frequency, refer [2] DRRange = Max125kHzDR to Max125kHzDR, refer [2]	Adds new channel and disables it
			MAC-CMD LinkADRReq TXPower = Maximum, refer [2] DataRate = Max125kHzDR, refer [2] ChMaskCntl = 0 ChMask - Enable only the default channels, refer [2] NbTrans = 1	
			Payload = [0x]07XXXXXXXXXX[0x]03X XXXXXXX	
2	DUT sends Unconfirmed frame FCntUp = n + 1	→	DataRate = Max125kHzDR, refer [2] MAC-CMD NewChannelAns	
			MAC-CMD LinkADRAns Payload = [0x]0703[0x]0307	
3	Wait until all channels configured are used at least once to confirm the channel plan [Wait for a maximum of (5 * number of channels configured) uplink packets]	→ R [5*NbCh]	All default channels are used at least once. The newly added channel is not used.	DUT does not modify its channel plan
4	DUT sends Unconfirmed frame FCntUp = i	\rightarrow		
	The TCL sends Unconfirmed frame	←	MAC-CMD LinkADRReq TXPower = Maximum, refer [2] DataRate = Max125kHzDR, refer [2] ChMaskCntl = 0 ChMask - Enable the default channels and the newly added channel only, refer [2] NbTrans = 1 Payload = [0x]03XXXXXXXXX	Enables the new channel



Step	Procedure		Frame Sequence	Test Purpose
		End Device - TCL	Frame	
5	DUT sends Unconfirmed frame FCntUp = i + 1	→	DataRate = Max125kHzDR, refer [2] MAC-CMD LinkADRAns Payload = [0x]0307	
6	Wait until all channels configured are used at least once to confirm the channel plan [Wait for a maximum of (5 * number of channels configured) uplink packets]	→ R [5*NbCh]	All default channels and the newly added channel are used at least once	New channel added to default plan
7	DUT sends Unconfirmed frame FCntUp = j	→		
	The TCL sends Unconfirmed frame	+	MAC-CMD NewChannelReq ChIndex = X Freq = 0 MHz Payload = [0x]07XXXXXXXXXX	Removes new channel
8	DUT sends Unconfirmed frame FCntUp = j + 1	→	MAC-CMD NewChannelAns Payload = [0x]0703	Channel removed
9	DUT sends Unconfirmed frame FCntUp = k The TCL sends Unconfirmed frame	→		Tries to disable all
	THE TOE Serius Officentiffied traffie	+	MAC-CMD LinkADRReq TXPower = Maximum, refer [2] DataRate = Max125kHzDR, refer [2] ChMaskCntl = 0 ChMask = [0x]0000 NbTrans = 1 Payload = [0x]03XXXXXXXX	channels
10	DUT sends Unconfirmed frame FCntUp = k + 1	\rightarrow	MAC-CMD LinkADRAns Payload NOT = [0x]0307	Channel plan not modified
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq TXPower = Maximum, refer [2] DataRate = Max125kHzDR, refer [2] ChMaskCntl = 0 ChMask - Enable only the default channels, refer [2] NbTrans = 1 Payload = [0x]03XXXXXXXXX	
11	DUT sends Unconfirmed frame FCntUp = k + 2	→	MAC-CMD LinkADRAns Payload = [0x]0307	DUT reverted to default settings

2.5.8.b.ii. For Fixed channel plan Devices

This section is applicable for only Fixed channel plan devices.

811





813	2.5.8.b.ii.1. 125khz Uplink Channel Management
814	This test validates the DUT 's ability to process commands to operate at Data Rates
815	allowed for 125 kHz uplink channels as defined in <a>[2] . It also validates the DUT replies
816	appropriately to invalid LinkADRReq commands with unsuccessful LinkADRAns.
817	2.5.8.b.ii.1.1. Valid Command Processing
818	The TCL sends LinkADRReq commands to change the DUT's DataRate to each of
819	the allowed Data Rates for 125 kHz uplink channels, refer [2]. The commands use
820	channel mask control 6 and a channel mask value of [0x]00FF.
821	
822	Verify
823	 The DUT replies to each valid request with a successful LinkADRAns.
824	 The DataRate of subsequent uplinks is as commanded.
825	2.5.8.b.ii.1.1.1 Test Procedure Frame Sequence Chart
826	



Step	Procedure		Frame Sequence	Test Purpose
		End Device - TCL	Frame	
1	DUT sends Unconfirmed frame FCntUp = n	→		
	The TCL sends Unconfirmed frames	+	CP-CMD RegionalDutyCycleCtrlReq- OFF Payload = [0x]0500 FPort = 224	
2	DUT sends Unconfirmed frame FCntUp = n + 1	\rightarrow		
	The TCL sends Unconfirmed frame	←	MAC-CMD LinkADRReq TXPower = Maximum refer [2] DataRate = The first allowed data rates for 125 kHz uplink channels, refer [2] ChMaskCntl = 6 ChMask = [0x]00FF NbTrans = 1 Payload = [0x]03XXXXXXXX	
3	DUT sends Unconfirmed frame FCntUp >= n + 2 The TCL sends Unconfirmed frame Repeat the test for all the remaining allowed Data Rates for 125 kHz uplink channels	→ R [All DR for 125kHz] ← R [All DR for 125kHz]	MAC-CMD LinkADRAns DataRate = As set in the LinkADRReq Payload = [0x]0307 MAC-CMD LinkADRReq TXPower = Maximum refer [2] DataRate = Each of the other allowed data rates for 125 kHz uplink channels, refer [2] ChMaskCntl = 6 ChMask = [0x]00FF NbTrans = 1 Payload = [0x]03XXXXXXXXX	
4	DUT sends Unconfirmed frame	→	MAC-CMD LinkADRAns DataRate = As set in the LinkADRReq Payload = [0x]0307	All allowed DRs tested
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq TXPower = Maximum DataRate = [0x]F ChMaskCntl = 6 ChMask - [0x]00FF NbTrans = 1 Payload = [0x]03XXXXXXXX	Testing for DR = [0x]F
5	DUT sends Unconfirmed frame	→	MAC-CMD LinkADRAns Payload = [0x]0307	Request accepted and no change to DR



Step	Procedure		Frame Sequence	Test Purpose
		End	Frame	
		Device		
		- TCL		
	The TCL sends Unconfirmed	+	MAC-CMD LinkADRReq	
	frame		TXPower = Maximum, refer	
			[2]	
			DataRate = Max125kHzDR,	
			refer [2]	
			ChMaskCntl = 6	
			ChMask = [0x]00FF	
			NbTrans = 1	
			Payload =	
			[0x]03XXXXXXX	
6	DUT sends Unconfirmed frame	\rightarrow	MAC-CMD LinkADRAns	DUT reverted to
			Payload = [0x]0307	default settings
	The TCL sends Unconfirmed	+	CP-CMD	
	frames		RegionalDutyCycleCtrlReq-	
			ON	
			FPort = 224	
			Payload = [0x]0501	
7	DUT sends Unconfirmed frame	\rightarrow		

2.5.8.b.ii.1.2. Invalid Command Processing

827

828 829

830

831 832

834

835 836

837 838 839

840 841

842843

844

845

846847

848 849 850

833 834

Verify

• The **DUT** replies to the first command with a successful LinkADRAns ([0x]0307) and to the second with an unsuccessful LinkADRAns. Acceptable values of the second LinkADRAns include (bad DR) [0x]0305, or (bad ChMask) [0x]0306 or (both bad DR & bad Channel) [0x]0304.

Using separate downlinks, the TCL sends two commands: the first is a valid

command disabling all 500 kHz uplink Channels with mask index 6, channel mask

of [0x]0000 and an allowed Data Rate for 125 kHz uplink channels, refer [2]. The

second is an invalid LinkADRReq command with channel mask control 6, channel

mask of [0x]0000 and an allowed Data Rates for 500 kHz uplink channels, refer [2]

Next, using separate downlinks, the **TCL** sends two commands: the first is a valid command disabling all 125 kHz uplink Channels with mask index 7, channel mask of [0x]00FF and an allowed Data Rate for 500 kHz uplink channels, refer [2]. The second is an invalid command with a mask index 0, channel mask of [0x]0001 and an allowed Data Rate for 125 kHz uplink channels, refer [2].

Verify

• The **DUT** replies to the first command with a successful LinkADRAns ([0x]0307) and to the second with an unsuccessful LinkADRAns. Acceptable values of the second LinkADRAns include (bad DR) [0x]0305, or (bad ChMask) [0x]0306 or (both bad DR & bad Channel) [0x]0304.





2.5.8.b.ii.1.2.1. Test Procedure Frame Sequence Chart



Step	Procedure		Frame Sequence	Test Purpose
		End Device - TCL	Frame	
1	DUT sends Unconfirmed frame	→		
	FCntUp = n The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq TXPower = Maximum refer [2] DataRate = An allowed data rate for 125 kHz uplink channels, refer [2] ChMaskCntl = 6 ChMask = [0x]0000 NbTrans = 1 Payload = [0x]03XXXXXXXX	
2	DUT sends Unconfirmed frame FCntUp = n + 1	→	MAC-CMD LinkADRAns DataRate = As set in the LinkADRReq Payload = [0x]0307	Successful LinkADRAns
	The TCL sends Unconfirmed frame	←	MAC-CMD LinkADRReq TXPower = Maximum refer [2] DataRate = An allowed data rates for 500 kHz uplink channels, refer [2] ChMaskCntl = 6 ChMask = [0x]0000 NbTrans = 1 Payload = [0x]03XXXXXXXX	
3	DUT sends Unconfirmed frame FCntUp = n + 2)	MAC-CMD LinkADRAns Payload = Any one of these- (bad DR) [0x]0305, or (bad ChMask) [0x]0306 or (both bad DR & bad Channel) [0x]0304.	Unsuccessful LinkADRAns
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq TXPower = Maximum refer [2] DataRate = An allowed data rate for 500 kHz uplink channels, refer [2] ChMaskCntl = 7 ChMask = [0x]00FF NbTrans = 1 Payload = [0x]03XXXXXXXX	
4	DUT sends Unconfirmed frame FCntUp = n + 3	→	MAC-CMD LinkADRAns DataRate = As set in the LinkADRReq Payload = [0x]0307	Successful LinkADRAns



Step	Procedure		Frame Sequence	Test Purpose
		End Device - TCL	Frame	
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq TXPower = Maximum refer [2] DataRate = An allowed data rate for 125 kHz uplink channels, refer [2] ChMaskCntl = 0 ChMask = [0x]0001 NbTrans = 1 Payload = [0x]03XXXXXXXX	
5	DUT sends Unconfirmed frame FCntUp = n + 4	→	MAC-CMD LinkADRAns Payload = Any one of these- (bad DR) [0x]0305, or (bad ChMask) [0x]0306 or (both bad DR & bad Channel) [0x]0304.	Unsuccessful LinkADRAns
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq TXPower = Maximum, refer [2] DataRate = Max125kHzDR, refer [2] ChMaskCntl = 6 ChMask = [0x]00FF NbTrans = 1 Payload = [0x]03XXXXXXXX	
6	DUT sends Unconfirmed frame	→	MAC-CMD LinkADRAns Payload = [0x]0307	DUT reverted to default settings

854 855

856 857 858

859 860

861

2.5.8.b.ii.1.3. Enable All-Channels

The **TCL** then sends a LinkADRReq to enable all channels. In this case the command's channel mask control index is 6 and the DR specified must be appropriate for 125 kHz channels. The channel mask must be applied to the 500 kHz channels. The **DUT** *must* reply with a successful *LinkADRAns*. The TCL verifies that DUT resumes using the full range (64) of 125kHz uplink channels in random sequence at an appropriate DR.

2.5.8.b.ii.1.3.1. Test Procedure Frame Sequence Chart



Step	Procedure		Frame Sequence	Test Purpose
		End	Frame	
		Device -		
		TCL		
1	DUT sends Unconfirmed frame	\rightarrow]	
	FCntUp = n			
	The TCL sends Unconfirmed	+	MAC-CMD LinkADRReq	
	frame		TXPower = Maximum refer	
			[2]	
			DataRate = Max data rate	
			for 125 kHz uplink	
			channels, refer [2]	
			ChMaskCntl = 6	
			ChMask = [0x]00FF	
			NbTrans = 1	
			Payload =	
			[0x]03XXXXXXX	
2	DUT sends Unconfirmed frame	\rightarrow	MAC-CMD LinkADRAns	
	FCntUp = n + 1		DataRate = As set in the	
			previous step	
		, 5	Payload = [0x]0307	All I
3	Wait for a maximum of 2 *	→ R	Official certification: DUT resumes using the full	All channels
	(number of channels configured)	[2*NbCh]	range (64) of 125kHz	must be used at
	uplink packets to be sent, i.e. until all channels are used at least	or	uplink channels in random	least once
		[All Ch	sequence at the DR set	
	once.	used]	Sequence at the Bit set	
			Pre-testing with 8-channel	
			gateway:	
			Channels 0-7 and Channel	
			64 must be used at least	
			once.	
4	DUT sends Unconfirmed frame	→		
	The TCL sends Unconfirmed	←	MAC-CMD LinkADRReq	
	frame		TXPower = Maximum, refer	
			[2] Data Data	
			DataRate =	
			Max125kHzDR, refer [2]	
			ChMaskCntl = 6	
			ChMask = [0x]00FF NbTrans = 1	
			Payload =	
			[0x]03XXXXXXXX	
5	DUT sends Unconfirmed frame	\rightarrow	MAC-CMD LinkADRAns	DUT reverted to
٥	Do i serius oriconilimeu name	7	Payload = [0x]0307	default settings
		<u> </u>	r ayluau = [Ux]USU1	uciauli sellings

2.5.8.b.ii.2. 500 kHz Uplink Channel Management

This test validates the **DUT**'s ability to process commands to operate at Data Rates allowed for 500 kHz uplink channels as defined in [2]. It also validates the **DUT** replies appropriately to invalid *LinkADRReq* commands with an unsuccessful *LinkADRAns*.

2.5.8.b.ii.2.1. Valid Command Processing

The **TCL** sends *LinkADRReq* commands the **DUT**'s Data Rate to an allowed DataRate for 500 kHz uplink channels, refer [2]. The commands use channel mask control 7 and a channel mask value of [0x]00FF.

862

863

864 865

866

867





870	
871	Verify
872	• The DUT replies to the valid request with a successful LinkADRAns ([0x]0307)
873	 The Data Rate of subsequent uplinks is as set by TCL command, and all 500kHz
874	uplink channels are used.
875	Next the TCL sends a single LinkADRReq command to the DUT with an allowed
876	DataRate for 500 kHz uplink channels, refer [2], channel mask control 7, and
877	channel mask value of [0x]0001.
878	Verify
879	• The DUT replies to the valid request with a successful LinkADRAns ([0x]0307)
880	 All subsequent uplinks are sent on Channel 64 at the DataRate set by the TCL
881	command.
882	2.5.8.b.ii.2.1.1. Test Procedure Frame Sequence Chart
883	



Step	Procedure		Frame Sequence	Test Purpose
		End	Frame	
		Device - TCL		
1	DUT sends Unconfirmed frame	- ICL →		
'	FCntUp = n			
	The TCL sends Unconfirmed	+	MAC-CMD LinkADRReq	
	frame		TXPower = Maximum refer	
			Data Data Amallawa di data	
			DataRate = An allowed data rate for 500 kHz uplink	
			channels, refer [2]	
			ChMaskCntl = 7	
			ChMask = [0x]00FF	
			NbTrans = 1	
			Payload =	
2	DUT sends Unconfirmed frame	→	[0x]03XXXXXXXX MAC-CMD LinkADRAns	All allowed DRs
_	FCntUp = n + 1	-	DataRate = As set in the	tested
			previous step	
			Payload = [0x]0307	
	The TCL sends Unconfirmed	←	MAC-CMD LinkADRReq TXPower = Maximum, refer	
	frame		[2]	
			DataRate = An allowed DR	
			for 500 kHz, refer [2]	
			ChMaskCntl = 7	
			ChMask = [0x]0001 NbTrans = 1	
			Payload =	
			[0x]03XXXXXXX	
3	DUT sends Unconfirmed frame	→	MAC-CMD LinkADRAns	
	FCntUp = n + 2		Payload = [0x]0307	
4	DUT sends Unconfirmed frame	\rightarrow	DUT sends uplinks only on	Channel and
	FCntUp = n + 3		Channel 64 at the DataRate set	DR tested
			361	
5	DUT sends Unconfirmed frame	\rightarrow	DUT sends uplinks only on	
	FCntUp = n + 4		Channel 64 at the DataRate	
	T. TO		set	
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq TXPower = Maximum, refer	
	name		[2]	
			DataRate = Max125kHzDR,	
			refer [2]	
			ChMaskCntl = 6	
			ChMask = [0x]00FF NbTrans = 1	
			Payload =	
			[0x]03XXXXXXX	
6	DUT sends Unconfirmed frame	→	MAC-CMD LinkADRAns	DUT reverted to
	FCntUp = n + 5		Payload = [0x]0307	default settings





898

899

900

2.5.8.b.ii.2.2. Invalid Command Processing 885 886 Using separate downlinks, the TCL sends two commands: the first is a valid command disabling all 125 kHz uplink channels with mask index 7, channel mask 887 888 of [0x]00FF and an allowed DataRate for 500 kHz uplink channels, refer [2]. The 889 second is an invalid LinkADRReq command with channel mask control 7, channel 890 mask of [0x]0000 (or [0x]FF00) and an allowed DataRate for 125 kHz uplink 891 channels, refer [2]. 892 893 Verify 894 The **DUT** must reply to the first command with a successful LinkADRAns 895 ([0x]0307) and to the second with an unsuccessful LinkADRAns of (bad DR) [0x]0305, or (bad ChMask) [0x]0306 or (both bad DR & bad Channel) [0x]0304. 896

The DataRate of subsequent uplinks is as set by the TCL command.

2.5.8.b.ii.2.2.1. Test Procedure Frame Sequence Chart

([0x]0304).



Step	Procedure		Frame Sequence	Test Purpose
		End Device - TCL	Frame	
1	DUT sends Unconfirmed frame FCntUp = n	>		
	The TCL sends Unconfirmed frame	←	MAC-CMD LinkADRReq TXPower = Maximum refer [2] DataRate = An allowed data rate for 500 kHz uplink channels, refer [2] ChMaskCntl = 7 ChMask = [0x]00FF NbTrans = 1 Payload = [0x]03XXXXXXXX	
2	DUT sends Unconfirmed frame FCntUp = n + 1	→	MAC-CMD LinkADRAns DataRate = As set in the LinkADRReq Payload = [0x]0307	Successful LinkADRAns
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq TXPower = Maximum refer [2] DataRate = An allowed data rate for 125 kHz uplink channels, refer [2] ChMaskCntl = 7 ChMask = [0x]0000 NbTrans = 1 Payload = [0x]03XXXXXXXX	
3	DUT sends Unconfirmed frame FCntUp = n + 2	→	MAC-CMD LinkADRAns Payload = Any one of these- (bad DR) [0x]0305, or (bad ChMask) [0x]0306 or (both bad DR & bad Channel) [0x]0304.	Unsuccessful LinkADRAns
	The TCL sends Unconfirmed frame	←	MAC-CMD LinkADRReq TXPower = Maximum, refer [2] DataRate = Max125kHzDR, refer [2] ChMaskCntl = 6 ChMask = [0x]00FF NbTrans = 1 Payload = [0x]03XXXXXXXX	
4	DUT sends Unconfirmed frame	\rightarrow	MAC-CMD LinkADRAns Payload = [0x]0307	DUT reverted to default settings



905 906

907

908

909

910

911 912

913

914

915

916

917

918

919

920

921

922 923

924

925

926

2.5.8.b.ii.3. Disable all Channels (125kHz and 500kHz channels)

Lastly, the **TCL** then tries to disable all channels by sending a LinkADRReq command with channel mask control index of 7 and a channel mask of [0x]0000. The **DUT** *must* reply with an unsuccessful *LinkADRAns* of (bad DR) [0x]0305, or (bad ChMask) [0x]0306 or (both bad DR & bad ChMask) [0x]0304.

2.5.8.b.ii.3.1. Test Procedure Frame Sequence Chart

Step	Procedure		Frame Sequence	Test Purpose
		End Device - TCL	Frame	
1	DUT sends Unconfirmed frame FCntUp = n	→		
	The TCL sends Unconfirmed frame	←	MAC-CMD LinkADRReq TXPower = Maximum refer [2] DataRate = An allowed data rate, refer [2] ChMaskCntl = 7 ChMask = [0x]0000 NbTrans = 1 Payload = [0x]03XXXXXXXX	
2	DUT sends Unconfirmed frame FCntUp = n + 1	→	MAC-CMD LinkADRAns DataRate = As set in the LinkADRReq Payload = Any one of these- (bad DR) [0x]0305, or (bad ChMask) [0x]0306 or (both bad DR & bad ChMask) [0x]0304.	Unsuccessful LinkADRAns when disabling all channels

2.5.8.c. **Redundancy**

This test validates the **DUT**'s correct implementation *NbTrans* setting within the *LinkADRReq* command. The **TCL** sends *LinkADRReq* requesting a repetition count of 2.

Verify

- All subsequent unconfirmed/confirmed uplink frames from the **DUT** are transmitted twice (the same sequence number is received twice by the **TCL**). The number of redundant uplinks verified by the harness is at the discretion of the Test Tool vendor.
- The **DUT** does not repeat the transmission if a downlink is received during the RX1 window.
- The **DUT** does not repeat the transmission if a downlink is received during the RX2 window.
- For Fixed channel plan devices only: The DUT appropriately performs frequency hopping for repeated transmissions.

Once the above criteria are verified for a *NbTrans* setting of 2, the **DUT** is commanded back to the default of 1 by the **TCL** using a *LinkADRReq* command wherein the *NbTrans* value is 0. This default setting is also validated.

The above test is repeated for NbTrans = 3 within the LinkADRReq command.





2.5.8.c.i. Test Procedure Frame Sequence Chart



Step	Procedure		Test Purpose	
		End Device - TCL	Frame	
1	DUT sends Unconfirmed frame FCntUp = n	→		
	The TCL sends Unconfirmed frames	+	CP-CMD RxAppCntReq FPort = 224 Payload = [0x]09	
2	DUT sends Unconfirmed frame FCntUp = n + 1	→	CP-CMD RxAppCntAns FPort = 224 Payload = [0x]09XXXX RxAppCnt = x	
3	DUT sends Unconfirmed frame FCntUp = n + 2	\rightarrow		
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq TXPower = Maximum, refer [2] DataRate = Max125kHzDR, refer [2] ChMaskCntl = 0 (for DC) and 6 (for FC) ChMask = Enable only the default channels for DC, refer [2] and [0x]00FF for FC NbTrans = 2 Payload = [0x]03XXXXXXXX	
4	DUT sends Unconfirmed frame FCntUp = n + 3	→	MAC-CMD LinkADRAns Payload = [0x]0307	
5	DUT sends Unconfirmed frame FCntUp = n + 3	→	MAC-CMD LinkADRAns Payload = [0x]0307 FC plan only: The frequency is different from the previous uplink.	Uplink sent twice
6	DUT sends Unconfirmed frame FCntUp = n + 4	→		
7	DUT sends Unconfirmed frame FCntUp = n + 4	→		
8	DUT sends Unconfirmed frame FCntUp = n + 5	→		
	The TCL sends Unconfirmed frames on RX1 window	←	CP-CMD RxAppCntReq FPort = 224 Payload = [0x]09	
9	DUT sends Unconfirmed frame FCntUp = n + 6)	CP-CMD RxAppCntAns FPort = 224 Payload = [0x]09XXXX RxAppCnt = x + 1	Transmit not repeated when downlink received on RX1



Step	Procedure		Test	
		End Device	Frame	Purpose
		- TCL	Frame	
10	DUT sends Unconfirmed frame FCntUp = n + 6)	CP-CMD RxAppCntAns FPort = 224 Payload = [0x]09XXXX RxAppCnt = x + 1 FC plan only: The frequency is different from the previous uplink.	
11	DUT sends Unconfirmed frame FCntUp = n + 7	→		
	The TCL sends Unconfirmed frame on RX2 window	+	CP-CMD RxAppCntReq FPort = 224 Payload = [0x]09	
12	DUT sends Unconfirmed frame FCntUp = n + 8)	CP-CMD RxAppCntAns FPort = 224 Payload = [0x]09XXXX RxAppCnt = x + 2	Transmit not repeated when downlink received on RX2
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq TXPower = Maximum, refer [2] DataRate = Max125kHzDR, refer [2] ChMaskCntl = 0 (for DC) and 6 (for FC) ChMask = Enable only the default channels for DC, refer [2] and [0x]00FF for FC NbTrans = 1 Payload = [0x]03XXXXXXXXX	
13	DUT sends Unconfirmed frame FCntUp = n + 9	→	MAC-CMD LinkADRAns Payload = [0x]0307	Uplink sent once
14	DUT sends Unconfirmed frame FCntUp = n + 10	→		
	The TCL sends Unconfirmed frame	+	CP-CMD TxFramesCtrlReq FPort = 224 Frame type = Confirmed Payload = [0x]0702	Switch to Confirmed frames
15	DUT sends Confirmed frame FCntUp = n + 11	→		



Step	Procedure	Frame Sequence		Test Purpose
		End Device - TCL	Frame	•
	The TCL sends Unconfirmed frame	+	Acknowledge	
			MAC-CMD LinkADRReq TXPower = Maximum, refer [2] DataRate = Max125kHzDR, refer [2] ChMaskCntl = 0 (for DC) and 6 (for FC) ChMask = Enable only the default channels for DC, refer [2] and [0x]00FF for FC NbTrans = 3	
10	DUT sends Confirmed frame	→	Payload = [0x]03XXXXXXXX MAC-CMD LinkADRAns	
16	FCntUp = n + 12	7	Payload = [0x]0307	
17	DUT sends Confirmed frame	\rightarrow	MAC-CMD LinkADRAns	
	FCntUp = n + 12		Payload = [0x]0307	
			FC plan only: The frequency is different from the previous uplink.	
18	DUT sends Confirmed frame FCntUp = n + 12	→	MAC-CMD LinkADRAns Payload = [0x]0307 FC plan only: The frequency is different from the previous 2 uplinks.	Uplink sent thrice
19	DUT sends Confirmed frame FCntUp = n + 13	→		
	The TCL sends Unconfirmed frame	+	Acknowledge CP-CMD TxFramesCtrlReq FPort = 224 Frame type = Unconfirmed Payload = [0x]0701	Revert to Unconfirmed frames
20	DUT sends Unconfirmed frame FCntUp = n + 14	→		
	The TCL sends Unconfirmed frame	+	CP-CMD RxAppCntReq FPort = 224 Payload = [0x]09	
21	DUT sends Unconfirmed frame FCntUp = n + 15)	CP-CMD RxAppCntAns FPort = 224 Payload = [0x]09XXXX RxAppCnt = x + 6	Transmit not repeated when downlink received



Step	Procedure		Frame Sequence	Test
			1 =	Purpose
		End Device - TCL	Frame	
22	DUT sends Unconfirmed frame FCntUp = n + 15)	CP-CMD RxAppCntAns FPort = 224 Payload = [0x]09XXXX RxAppCnt = x + 6 FC plan only: The frequency is different from the previous	
23	DUT sends Unconfirmed frame FCntUp = n + 15	→	uplink. CP-CMD RxAppCntAns FPort = 224 Payload = [0x]09XXXX RxAppCnt = x + 6 FC plan only: The frequency is different from the previous	
	The TCL sends Unconfirmed frame	←	2 uplinks. MAC-CMD LinkADRReq TXPower = Maximum, refer [2] DataRate = Max125kHzDR, refer [2] ChMaskCntl = 0 (for DC) and 6 (for FC) ChMask = Enable only the default channels for DC, refer [2] and [0x]00FF for FC NbTrans = 0 Payload = [0x]03XXXXXXXX	
24	DUT sends Unconfirmed frame FCntUp = n + 16	→	MAC-CMD LinkADRAns Payload = [0x]0307	Uplink sent once
25	DUT sends Unconfirmed frame FCntUp = n + 17	→		
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq TXPower = Maximum, refer [2] DataRate = Max125kHzDR, refer [2] ChMaskCntl = 0 (for DC) and 6 (for FC) ChMask = Enable only the default channels for DC, refer [2] and [0x]00FF for FC NbTrans = 1 Payload = [0x]03XXXXXXXX	
26	DUT sends Unconfirmed frame FCntUp = n + 18	→	MAC-CMD LinkADRAns Payload = [0x]0307	DUT reverted to default settings

2.5.8.d. **Data Rate Decay**

929

930 **2.5.8.d.i. DR** Decay test for all devices





931	This test validates the DUT 's implementation of the <i>ADR_ACK_LIMIT</i> ,
932	ADR_ACK_DELAY, and use of the ADRACKReq bit in its uplinks' frame header. A
933	tolerance of +/- 2 uplinks is allowed for the DUT's implementation of both the
934	ADR_ACK_LIMIT and ADR_ACK_DELAY parameters.
935	
936	The TCL adds a new channel.
937	The TCL sends <i>LinkADRReq</i> to configure the DUT to use
938	 the Max125kHzDR data rate for DC and 500kHz for FC,
939	- TXPower Index = 1,
940	- For DC devices: Disable all default channels and enable only the newly added
941	channel,
942	- For FC devices: Enable channel 0, 1 and 64,
943	- NbTrans set to 2.
944	
945	It then stops sending downlinks until the DUT decays to the minimum default data
946	rate, refer [2] .
947	
948	Verify
949	 After sending ADR_ACK_LIMIT (64) uplinks the DUT must - in a sustained
950	absence of downlinks - set the ADRACKReq bit in the next ADR_ACK_DELAY
951	(32) uplinks' frame headers.
952	 As of its 96th consecutive uplink in the continued absence of downlinks, the
953	DUT must change the TXPower Index to 0 and leave the ADRACKReq bit set.
954	 As of its 128th consecutive uplink in the continued absence of downlinks, the
955	DUT must lower its data rate to Default, leave the TXPower Index set to 0 and
956	ADRACKReq bit set.
957	 The DUT further lowers its data rate by 1 each time another ADR_ACK_DELAY
958	cycle occurs (32) uplinks are sent in the sustained absence of downlinks.
959	 For FC plan devices, when the DR is decayed from a 500kHz DR to a 125KHz
960	DR, Channel 64 must be discontinued and Channels 0 and 1 must be used.
961	 After the DR reaches MinDR, in the continued absence of downlinks, the DUT
962	must re-enable all default uplink frequency channels for DC plan devices, re-
963	enable all channels for FC plan devices, reset <i>NbTrans</i> to 1, retain the
964	TXPower at Default, retain the DataRate at Default and the ADRACKReq bit is
965	kept set in subsequent uplinks.
966	 After receiving a downlink, the DUT unsets the ADRACKReq bit.
967	, , , , , , , , , , , , , , , , , , , ,
968	
969	2.5.8.d.i.1. Test Procedure Frame Sequence Chart
970	2.3.0.u.i.1. rest rioceunie rianie sequence chait
J, U	



Step	Procedure		Frame Sequence	Test Purpose
		End Device - TCL	Frame	розг
1	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frames	←	CP-CMD RegionalDutyCycleCtrlReq- OFF FPort = 224 Payload = [0x]0500	
2	DUT sends Unconfirmed frame	\rightarrow	,	
	The TCL sends Unconfirmed frame Note: This step is required for only regions with Dwell Time limitation	+	MAC-CMD TXParamSetupReq UplinkDwellTime = 0 Payload = [0x]09XX	Set TXParamSet upReq for regions with Dwell time limitation
3	DUT sends Unconfirmed frame	→	For regions with Dwell time limitation only MAC-CMD TXParamSetupAns Payload = [0x]09	
	The TCL sends Unconfirmed frame Note: This step must be performed for DC plan devices only	+	MAC-CMD NewChannelReq ChIndex = The first non- default channel, refer [2] Freq = Any allowed frequency for the channel, refer [2] DRRange = 0-5, refer [2] Payload = [0x]07XXXXXXXXXXX	Create a new channel
4	DUT sends Unconfirmed frame Note: This step must be performed for DC plan devices only	→	MAC-CMD NewChannelAns Payload = [0x]0703	
	The TCL sends Unconfirmed frame Note: This step must be performed for DC plan devices only	←	MAC-CMD LinkADRReq TXPower Index = 1, refer [2] DataRate = X (where X = Max125kHzDR), refer [2] ChMaskCntl = 0 ChMask = Disable all default channels and enable only the newly added channel, refer [2] NbTrans = 2 Payload = [0x]03XXXXXXXX	
5	DUT sends Unconfirmed frame)	If the LinkADRReq was sent in the previous step, MAC-CMD LinkADRAns Payload = [0x]0307 NbTrans = 2	



Step	Procedure		Frame Sequence	Test
		End Device	Frame	Purpose
		- TCL		
	The TCL sends Unconfirmed frame	←	MAC-CMD LinkADRReq	
			TXPower Index = 1, refer [2]	
	Note: This step must be performed		DataRate = Max500kHzDR,	
	for FC plan devices only		refer [2]	
			ChMaskCntl = 7	
			ChMask = [0x]0001	
			NbTrans = 2	
			MAC-CMD LinkADRReq	
			TXPower Index = 1, refer [2]	
			DataRate = X (where X =	
			Max500kHzDR), refer [2]	
			ChMaskCntl = 0	
			ChMask = [0x]0003	
			NbTrans = 2	
			Payload =	
			[0x]03XXXXXXXX[0x]03XXX	
			XXXXX	
6	DUT sends Unconfirmed frame	\rightarrow	MAC-CMD LinkADRAns	
			MAC-CMD LinkADRAns	
	Note: This step must be performed		NbTrans = 2	
	for FC plan devices only		Payload = [0x]0307[0x]0307	
	This is a repeated frame	1	MAC-CMD LinkADRAns	Repeat as
			MAC-CMD LinkADRAns	NbTrans = 2
	DUT sends Unconfirmed frame		NbTrans = 2	
			Payload = $[0x]0307[0x]0307$	
	Note: This step must be performed			
	for FC plan devices only			
7	DUT sends Unconfirmed frame	→ R [63]	FCtrl ADRAckReq = False	
	FCntUp = n		DataRate = X	
	D 100 11 11 1 1 1		NbTrans = 2	
	Repeat 63 times without receiving any downlinks		TXPower Index = 1	
	ary downlinks		Only the channels enabled	
			must be used	
	This is a repeated frame	1	FCtrl ADRAckReq = False	Repeat as
			DataRate = X	NbTrans = 2
	DUT sends Unconfirmed frame		NbTrans = 2	
	FCntUp = n		TXPower Index = 1	
	Repeat 63 times without receiving		Only the channels enabled	
	any downlinks		must be used	
8	DUT sends Unconfirmed frame	→ R [32]	FCtrl ADRAckReq = True	
	FCntUp = n + 63	, [02]	DataRate = X	
			NbTrans = 2	
	Repeat 32 times without receiving		TXPower Index = 1	
	any downlinks (starting with n + 63)			
			Only the channels enabled	
			must be used	



Step	Procedure		Frame Sequence	Test Purpose
		End Device - TCL	Frame	
	This is a repeated frame		FCtrl ADRAckReq = True	Repeat as
			DataRate = X	NbTrans = 2
	DUT sends Unconfirmed frame		NbTrans = 2	
	FCntUp = n + 63		TXPower Index = 1	
	Repeat 32 times without receiving		Only the channels enabled	
	any downlinks (starting with n + 63)		must be used	
9	DUT sends Unconfirmed frame FCntUp = n + 95	→ R [32]	FCtrl ADRAckReq = True DataRate = X	
			NbTrans = 2	
	Repeat 32 times without receiving any downlinks (starting with n + 95)		TXPower Index = 0	
	, (commig		Only the channels enabled	
			must be used	
	This is a repeated frame		FCtrl ADRAckReq = True	Repeat as
	This is a repeated frame		DataRate = X	NbTrans = 2
	DUT sends Unconfirmed frame		NbTrans = 2	
	FCntUp = $n + 95$		TXPower Index = 0	
	Repeat 32 times without receiving		Only the channels enabled	
	any downlinks (starting with n + 95)		must be used	
10	DUT sends Unconfirmed frame	→ R [32]	FCtrl ADRAckReq = True	DUT
	FCntUp = FCntUp (previous) + 32	until minDR	DataRate = Next lower DR,	switches to
			refer [2], until it reaches the	next lower
	Repeat 32 times without receiving		minimum DR	DR, until it
	any downlinks		NbTrans = 2	decays to
	Note: Repeat this step until the		TXPower Index = 0	the minimun
	DUT decays to the minimum DR		Only the channels enabled	
	for the region, refer [2]. The test is repeated even when DR = minDR		must be used	
	repeated even when bit = inilibit		For FC only: If DataRate	
			decays from 500kHz DR to	
			125kHz DR, DUT must	
			discontinue usage of Channel	
			64, and instead use Channel	
			0 and 1	
	This is a repeated frame		FCtrl ADRAckReg = True	DUT
	The io a repeated frame		DataRate = Next lower DR,	switches to
	DUT sends Unconfirmed frame		refer [2], until it reaches the	next lower
	FCntUp = FCntUp (previous) + 32		minimum DR	DR, until it
	Sinop = 1 Sinop (provious) 1 02		NbTrans = 2	decays to
	Repeat 32 times without receiving		TXPower Index = 0	the minimum
	any downlinks			DR
			Only the channels enabled	
	Note: Repeat this step until the		must be used	
	DUT decays to the minimum DR		F F0 / W5 / 5 /	
	for the region, refer [2]. The test is		For FC only: If DataRate	
	repeated even when DR = minDR		decays from 500kHz DR to	
			125kHz DR, DUT must	
			discontinue usage of Channel	
			64, and instead use Channel	
			0 and 1	



Step	Procedure		Frame Sequence	Test
				Purpose
		End Device - TCL	Frame	
11	DUT sends Unconfirmed frame)	FCtrl ADRAckReq = True NbTrans = 1 DataRate = MinDR, refer [2] TXPower = 0	DUT switches to default settings for Nbtrans and channels
12	Wait for a <u>Dynamic channel</u> : maximum of 5 * (number of channels configured) <u>Fixed channel</u> : maximum of 2 * (number of channels configured) uplink packets to be sent, i.e. until all channels are used at least once.	→ R [5*NChDC] or [2*NbChFC] or [AllCh used]	Official certification (DC plan and FC plan): All channels configured must be used at least once Pre-testing for DC Plan: All default channels must be used at least once. Pre-testing for FC plan with 8-channel gateway: Channels 0-7 must be used at least once.	All channels must be used at least once
13	DUT sends Unconfirmed frame The TCL sends Unconfirmed frame	→	MAC-CMD LinkADRReq TXPower = Maximum, refer [2] DataRate = Max125kHzDR, refer [2] ChMaskCntl = 0 (for DC) and 6 (for FC) ChMask = Enable only the default channels for DC, refer [2] and [0x]00FF for FC NbTrans = 1 Payload = [0x]03XXXXXXXXX	
14	DUT sends Unconfirmed frame The TCL sends Unconfirmed frame	→	FCtrl ADRAckReq = False MAC-CMD LinkADRAns Payload = [0x]0307 CP-CMD RegionalDutyCycleCtrlReq-ON FPort = 224 Payload = [0x]0501	DUT reverted to default settings
15	DUT sends Unconfirmed frame	→		
	The TCL sends Unconfirmed frame Note: This step must be performed only for DC plan devices	\	MAC-CMD NewChannelReq ChIndex = The non-default channel added in Step 2 of this table. Freq = 0 MHz Payload = [0x]07XXXXXXXXXX	Remove additional channel added



Step	Procedure		Frame Sequence	Test Purpose
		End Device - TCL	Frame	
16	DUT sends Unconfirmed frame	→	If the NewChannelReq was sent in the previous step, MAC-CMD NewChannelAns Payload = [0x]0703	
	The TCL sends Unconfirmed frame Note: This step is required for only regions with Dwell Time limitation	←	MAC-CMD TXParamSetupReq UplinkDwellTime = default, refer [2] Payload = [0x]09XX	Reset Dwell time setting to default
17	DUT sends Unconfirmed frame	→	For regions with Dwell time limitation only MAC-CMD TXParamSetupAns Payload = [0x]09	

2.5.8.d.ii. Additional DR Decay test for only DC plan devices which support the optional data rates

This test validates that the **DUT** re-enables all the default channels when it decays from the Maximum optional Data Rate to Max125kHzDR.

The **TCL** adds a new channel for each optional data rate supported.

The TCL sends LinkADRReq to configure the DUT to use

- the Maximum data rate,
- TXPower Index = 1,
- Disable all default channels and enable only the newly added channel(s),
- NbTrans set to 2.

It then stops sending downlinks until the **DUT** decays from maximum optional data rate to Max125kHzDR, refer [2].

Verify

971

972

973974

975976

977

978

979

980

981

982

983 984

985 986

987

988 989

990

991

992

993 994

995

- After sending ADR_ACK_LIMIT (64) uplinks the **DUT** must in a sustained absence of downlinks set the ADRACKReq bit in the next ADR_ACK_DELAY (32) uplinks' frame headers.
- As of its 96th consecutive uplink in the continued absence of downlinks, the
 DUT must change the TXPower Index to 0 and leave the ADRACKReq bit set.
- As of its 128th consecutive uplink in the continued absence of downlinks, the
 DUT must lower its data rate to Default, leave the TXPower Power set to 0
 and ADRACKReq bit set.
- The **DUT** further lowers its data rate by 1 each time another *ADR_ACK_DELAY* cycle occurs (32) uplinks are sent in the sustained absence of downlinks.





997	After the Di	R reaches decays from an optional DataRate to a Max125kHzDR,
998	in the conti	nued absence of downlinks, the DUT must re-enable all default
999	uplink frequ	uency channels.
1000		
	0 = 0 1 !! 4	
1001	2.5.8.d.ii.1.	Test Procedure Frame Sequence Chart
1002		



Step	Procedure		Frame Sequence	Test Purpose
		End Device - TCL	Frame	T di posc
1	DUT sends Unconfirmed frame	→		
	The TCL sends Unconfirmed frames	+	CP-CMD RegionalDutyCycleCtrlReq- OFF FPort = 224 Payload = [0x]0500	
2	DUT sends Unconfirmed frame	→		
	The TCL sends Unconfirmed frame Note: This step is required for only regions with Dwell Time limitation	+	MAC-CMD TXParamSetupReq UplinkDwellTime = 0 Payload = [0x]09XX	Set TXParamSet upReq for regions with Dwell time limitation
3	DUT sends Unconfirmed frame	→	For regions with Dwell time limitation only MAC-CMD TXParamSetupAns Payload = [0x]09	
	The TCL sends Unconfirmed frame Note: This step must be performed only if the DUT supports 250kHz channels	+	MAC-CMD NewChannelReq ChIndex = An unused non- default channel, refer [2] Freq = Any allowed freq for a 250kHz channel, refer [2] DRRange = BW250OptionalDR- BW250OptionalDR, refer [2] Payload = [0x]07XXXXXXXXXXX	Create a new channel for 250kHz channel
4	DUT sends Unconfirmed frame	→	If the NewChannelReq was sent in the previous step, MAC-CMD NewChannelAns Payload = [0x]0703	
	The TCL sends Unconfirmed frame Note: This step must be performed only if the DUT supports FSK channels	+	MAC-CMD NewChannelReq ChIndex = An unused non- default channel, refer [2] Freq = Any allowed frequency for an FSK channel, refer [2] DRRange = FSK50OptionalDR- FSK50OptionalDR, refer [2] Payload = [0x]07XXXXXXXXXXX	Create a new channel for FSK channel if supported
5	DUT sends Unconfirmed frame	→	If the NewChannelReq was sent in the previous step, MAC-CMD NewChannelAns Payload = [0x]0703	



Step	Procedure		Frame Sequence	Test Purpose
		End Device - TCL	Frame	-
	The TCL sends Unconfirmed frame	*	MAC-CMD LinkADRReq TXPower Index = 1, refer [2] DataRate = X (where X = Maximum Supported Data Rate), refer [2] ChMaskCntl = 0 ChMask = Disable all default channels and enable only the newly added channel(s), refer [2] NbTrans = 1	
6	DUT sends Unconfirmed frame	→	Payload = [0x]03XXXXXXXX MAC-CMD LinkADRAns Payload = [0x]0307	
7	DUT sends Unconfirmed frame FCntUp = n Repeat 63 times without receiving any downlinks	→ R [63]	FCtrl ADRAckReq = False DataRate = X TXPower Index = 1 Only the channels enabled	
8	DUT sends Unconfirmed frame FCntUp = n + 63	→ R [32]	must be used FCtrl ADRAckReq = True DataRate = X TXPower Index = 1	
	Repeat 32 times without receiving any downlinks (starting with n + 63)		Only the channels enabled must be used	
9	DUT sends Unconfirmed frame FCntUp = n + 95 Repeat 32 times without receiving	→ R [32]	FCtrl ADRAckReq = True DataRate = X TXPower Index = 0	
	any downlinks (starting with n + 95)		Only the channels enabled must be used	
10	DUT sends Unconfirmed frame FCntUp = FCntUp (previous) + 32 Repeat 32 times without receiving any downlinks	→ R [32] Until DUT switches from the OptionalDR	FCtrl ADRAckReq = True DataRate = Next lower DR, refer [2], until it reaches Max125kHzDR, refer [2] TXPower Index = 0	DUT switches to next lower DR, until it decays to
	Note: Repeat this step until the DUT decays to the Max125kHzDR, refer [2]. When DR = Max125kHzDR, skip to the next	to Max125kHz DR	Only the channels enabled must be used	the Max125kHz DR
11	DUT sends Unconfirmed frame	→	FCtrl ADRAckReq = True DataRate = Max125kHzDR, refer [2] TXPower Index = 0 All default channels are enabled	DUT reenables all default channels



Step	Procedure		Frame Sequence	Test
			T	Purpose
		End Device - TCL	Frame	
12	Wait for a maximum of 5 * (number	→ R	All default channels must be	All default
12	of channels configured) uplink	[5*NChDC]	used at least once	channels
	packets to be sent, i.e. until all	or	used at least office	must be
	channels are used at least once.			used at least
	channels are used at least once.	[AllDefCh		
40	DUT and the confinered forces	used]		once
13	DUT sends Unconfirmed frame	→	1440 049 1: 14999	
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq	
			TXPower = Maximum, refer	
			[2]	
			DataRate = Max125kHzDR,	
			refer [2]	
			ChMaskCntl = 0	
			ChMask = Enable only the	
			default channels, refer [2]	
			NbTrans = 1	
			Payload = [0x]03XXXXXXXX	
14	DUT sends Unconfirmed frame	\rightarrow	FCtrl ADRAckReq = False	DUT
				reverted to
			MAC-CMD LinkADRAns	default
			Payload = [0x]0307	settings
	The TCL sends Unconfirmed frame	←	CP-CMD	
			RegionalDutyCycleCtrlReq-	
			ON	
			FPort = 224	
			Payload = [0x]0501	
15	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame	←	MAC-CMD	Reset Dwell
			TXParamSetupReq	time to
	Note: This step is required for only		UplinkDwellTime = default,	default
	regions with Dwell Time limitation		refer [2]	
			Payload = [0x]09XX	
16	DUT sends Unconfirmed frame	→	For regions with Dwell time	
			limitation only	
			MAC-CMD	
			TXParamSetupAns	
		_	Payload = [0x]09	
	The TCL sends Unconfirmed frame	←	MAC-CMD NewChannelReq	Removes
			ChIndex = The channel used	the 250kHz
	Note: This step must be performed		in Step3	channel
	only if the DUT supports 250kHz		Freq = 0	
	channels		DRRange =	
			BW250OptionalDR-	
			BW250OptionalDR, refer [2]	
			Payload =	
			[0x]07XXXXXXXXX	
17	DUT sends Unconfirmed frame	\rightarrow	If the NewChannelReq was	
			sent in the previous step,	
			MAC-CMD NewChannelAns	
			Payload = [0x]0703	



Step	Procedure		Frame Sequence	Test
			T =	Purpose
		End Device	Frame	
		- TCL		
	The TCL sends Unconfirmed frame	←	MAC-CMD NewChannelReq	Removes
			ChIndex = The channel used	the FSK
	Note: This step must be performed		in Step4	channel
	only if the DUT supports FSK		Freq = 0	
	channels		DRRange =	
			FSK50OptionalDR-	
			FSK50OptionalDR, refer [2]	
			Payload =	
			[0x]07XXXXXXXXX	
18	DUT sends Unconfirmed frame	\rightarrow	If the NewChannelReq was	
			sent in the previous step,	
			MAC-CMD NewChannelAns	
			Payload = [0x]0703	

1004

1005

1006 1007

1008

1009 1010

1011

1012

1013

10141015

1016

1017

1018 1019

2.5.8.e. Command Block Channel Management

This test validates the **DUT** correctly processes blocks of multiple *LinkADRReq* commands included in the same downlink. The **DUT** is expected to service each of these MAC commands in the same sequence as it is ordered in the FOpts or FRMPayload field.

2.5.8.e.i. Dynamic channel plan devices

2.5.8.e.i.1.Successful LinkADRReq block

The TCL sends multiple *LinkADRReq* commands in a single downlink as shownError! Reference source not found.Error! Reference source not found. in Table 1: Successful LinkADRReq blockError! Reference source not found.Error! Reference source not found.Error! Reference source not found. The first command tries to disable all channels. The channel mask the second command enables is *channel 0*, and any default DR except the minimum DR, is appropriate for this channel. The last command enables all defined channels using the channel mask control 6 and configures any other default data rate except the ones used earlier for this test. The **DUT** is expected to respond with *LinkADRAns* [0x]0307 for all *LinkADRReq* commands.

LinkADRReq	DR	TXPower	Channel	MaskCntl	NbTrans
CMD Index			Mask		
1	Minimum	Any allowed	[0x]0000	0	0
		TXPower			
2	Any default	Any allowed	[0x]0001	0	0
	DR except the	TXPower,			
	one set above	other than			
		the one set			
		above			



3	Any default	Any other	[0x]0000	6	1
	DR except the	allowed			
	ones set	TXPower			
	above				

Table 1:	Successful	LINKADKK	led plock	(
----------	------------	----------	-----------	---

Note:

1021 1022

10231024

1025

1026

1027

1028

1029

1030

1031

1032

1033

1034

1035

1036

1037

1038

- The DR, TX Power, and NbTrans values of only the last command are implemented by the **DUT**. Values for these parameters in any preceding command must be ignored.
- After processing the command block shown above, the **DUT** is expected to be configured as follows:

Channel Plan: Default channels only

Data Rate: As set in Step 3 in Table 1: Successful LinkADRReq block

TXPower: As set in Step 3 in Error! Reference source not found. Table 1:

Successful LinkADRReq block

Verify

- The **DUT** replies with an uplink containing a successful *LinkADRAns* for each command in the block.
- The uplink containing the answers and all subsequent transmissions are sent using the default channels at the commanded DR.

2.5.8.e.i.1.1. Test Procedure Frame Sequence Chart



Step	Procedure	i	Frame Sequence	Test Purpose
		End Device - TCL	Frame	- supres
1	DUT sends Unconfirmed frame	→		
	FCntUp = n			
	The TCL sends Unconfirmed frame	+	MAC-CMD1 LinkADRReq	
			TXPower = Any allowed	
			TXPower, refer [2]	
			DataRate = MinDR, refer [2]	
			ChMaskCntl = 0	
			ChMask = [0x]0000	
			NbTrans = 0	
			MAC-CMD2 LinkADRReq	
			TXPower = Any other	
			allowed TXPower, refer [2]	
			DataRate = Any default DR,	
			except the one set above in	
			this step, refer [2]	
			ChMaskCntl = 0	
			ChMask = [0x]0001 NbTrans = 0	
			Notialis = 0	
			MAC-CMD3 LinkADRReq	
			TXPower = Any other	
			TXPower, refer [2]	
			DataRate = Any default DR,	
			except the ones set above in	
			this step, refer [2]	
			ChMaskCntl = 6	
			ChMask = [0x]0000 NbTrans = 1	
			INDITIALIS = 1	
			Payload =	
			[0x]03XXXXXXX[0x]03XXX	
			XXXXX[0x]03XXXXXXX	
2	DUT sends Unconfirmed frame	→	DataRate = same as set in	Uplinks as
	FCntUp = n + 1		CMD3 above	configured
			MAC-CMD1 LinkADRAns	
			MAC-CMD2 LinkADRAns	
			MAC-CMD3 LinkADRAns	
			Payload =	
			[0x]0307[0x]0307[0x]0307	
3	Wait for a maximum of 5 * (number	→ R	All default channels must be	All default
	of channels configured) = uplink	[5*NbCh] or	used at least once	channels
	packets to be sent, i.e. until all	[All Ch used]		must be
	default channels are used at least			used at least
1	once. DUT sends Unconfirmed frame	→		once
4	Do i serius unconfirmed frame	フ		Ì



Step	Procedure	F	Frame Sequence	Test Purpos	se .
		End Device - TCL	Frame		
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq TXPower = Maximum, refer [2] DataRate = Max125kHzDR, refer [2] ChMaskCntl = 0 ChMask = Enable only the default channels, refer [2] NbTrans = 1 Payload = [0x]03XXXXXXXX		
5	DUT sends Unconfirmed frame	→	MAC-CMD LinkADRAns Payload = [0x]0307	DUT reverted default settings	to

1040 1041

1042

1043

1044

2.5.8.e.i.2. Unsuccessful LinkADRReq block

The **TCL** sends several *LinkADRReq* commands in a single downlink as shown in Table 2: Unsuccessful LinkADRReq block. **Error! Reference source not found.**In this case the last command requires all channels to be disabled, all commands must be rejected.

LinkADRReq CMD Index	DR	TXPow er	Channel Mask	MaskCntl	NbTrans
1	Any mandatory	Any	[0x]0001	0	0
	DR, except				
	Max125kHzDR				
2	Any other	Any	[0x]0000	6	1
	mandatory DR,				
	except				
	Max125kHzDR				
3	Max125kHzDR	Any	[0x]0000	0	1

Table 2: Unsuccessful LinkADRReq block

1045

1046 Verify

1047

1048 1049

1050

1051 1052 The **DUT** replies with an uplink containing an unsuccessful LinkADRAns for each command in the block.

 The uplink data rate does not change and the **DUT** continues using the default channels at the previously configured data rate.

2.5.8.e.i.2.1. Test Procedure Frame Sequence Chart



Step	Procedure		Test	
			Purpose	
		End Device - TCL	Frame	
1	DUT sends Unconfirmed frame FCntUp = n	→		
	The TCL sends Unconfirmed frame	\	MAC-CMD1 LinkADRReq TXPower = Any allowed value, refer [2] DataRate = X (where X = any mandatory DR, except Max125kHzDR, refer [2]) ChMaskCntl = 0 ChMask = [0x]0001 NbTrans = 0 MAC-CMD2 LinkADRReq TXPower = Any allowed value, refer [2] DataRate = Any other mandatory DR, except Max125kHzDR, refer [2] ChMaskCntl = 6 ChMask = [0x]0000 NbTrans = 1 MAC-CMD3 LinkADRReq TXPower = Any allowed value, refer [2] DataRate = Max125kHzDR, refer [2] DataRate = Max125kHzDR, refer [2] ChMaskCntl = 0	
			ChMask = [0x]0000 NbTrans = 1 Payload = [0x]03XXXXXXXXX[0x]03XXXXXXXX[0x]03XXXXXXXXX	
2	DUT sends Unconfirmed frame FCntUp = n + 1	→	MAC-CMD1 LinkADRAns MAC-CMD2 LinkADRAns MAC-CMD3 LinkADRAns Payload = [0x]030X[0x]030X[0x]030X (where X is NOT = 7)	All commands are rejected as the last command requires all channels to be disabled

2.5.8.e.ii. Fixed channel plan devices

2.5.8.e.ii.1. 125kHz Sub-Band Channel Plan

The **TCL** sends two *LinkADRReq* commands in a single downlink as shown in Table 3: 125kHz Sub-Band Channel Plan Command Block. The first command disables all 125 kHz channels and simultaneously enables the *channel 64* (500 kHz channel). The DataRate of the first command is any data rate allowed for uplinking on 500 kHz enabled channels. The channel mask control of the second command enables *channel 0 – 7*. The DataRate must be an allowed data rate allowed for 125 kHz

1053

1054

10551056

1057

1058

1059



uplink channels, refer [2]. The **DUT** is expected to respond with *LinkADRAns* [0x]0307 for both *LinkADRReq* commands.

LinkADRReq	DR	TXPower	Channel	Mask	NbTrans
CMD Index			Mask	Index	
1	Any DR	Maximum	[0x]0001	7	1
	allowed for				
	500 kHz uplink				
	channels				
2	Any DR	Any allowed	[0x]00FF	0	1
	allowed for	TXPower			
	125 kHz uplink	other than			
	channels	max			
		TXPower			

Table 3: 125kHz Sub-Band Channel Plan Command Block

1063 1064

1065

106610671068

1069

10701071

1072

1073

10741075

10761077

10781079

1080

Note:

- The DR, TX Power, and NbTrans values of only the last command are implemented by the **DUT**. Values for these parameters in any preceding command must be ignored.
- After processing the command block shown above, the **DUT** is expected to be configured as follows:

Channel Plan: Channels 0-7

Data Rate: DataRate set in the second command above **TXPower**: TXPower set in the second command above

Verify

- The **DUT** replies with an uplink containing a successful *LinkADRAns* for each command in the block.
- The uplink containing the answers and all subsequent transmissions are sent only on the enabled 125kHz channel plan at the commanded DR.

2.5.8.e.ii.1.1. Test Procedure Frame Sequence Chart



Step	Procedure	Frame Sequence		Test
		End Device - TCL	Frame	Purpose
1	DUT sends Unconfirmed frame FCntUp = n	→		
	The TCL sends Unconfirmed frame	←	MAC-CMD1 LinkADRReq TXPower = Maximum, refer [2] DataRate = Any DR allowed for 500 kHz uplink channels, refer [2] ChMaskCntl = 7 ChMask = [0x]0001 NbTrans = 1 MAC-CMD2 LinkADRReq TXPower = Any allowed TXPower other than max TXPower, refer [2] DataRate = Any DR allowed for 125 kHz uplink channels, refer [2] ChMaskCntl = 0 ChMask = [0x]00FF NbTrans = 1 Payload =	
	DUT and Union from all forms		[0x]03XXXXXXXX[0x]03XXX XXXXX	Hallaha
2	DUT sends Unconfirmed frame FCntUp = n + 1	→	DataRate and TXPower = same as set in CMD2 above Channel Plan = Channel 0-7 MAC-CMD1 LinkADRAns MAC-CMD2 LinkADRAns Payload = [0x]0307[0x]0307	Uplinks as configured
	The TCL sends Unconfirmed frame	←	MAC-CMD LinkADRReq TXPower = Maximum, refer [2] DataRate = Max125kHzDR, refer [2] ChMaskCntl = 6 ChMask = [0x]00FF NbTrans = 1 Payload = [0x]03XXXXXXXX	
3	DUT sends Unconfirmed frame FCntUp = n + 2	→	MAC-CMD LinkADRAns Payload = [0x]0307	DUT reverted to default settings





1083	
1084	2.5.9.DutyCycleReq
1085 1086 1087	The test validates that the DUT correctly updates its Duty Cycle – maximum aggregated transmit duty time.
1088 1089 1090 1091 1092 1093	TCL must then set the DUT to the maximum DataRate, refer [2]. DUT sends uplink frames and TCL stores the time of arrival of 2 consecutive uplink frames. TCL sends DutyCycleReq MAC command with a MaxDutyCycle value of 7 (Duty Cycle is smaller than 1% duty-cycle used for default channels) DUT again sends uplink frames and TCL stores the time of arrival of the next 2 consecutive uplink frames.
1094	Verify
1095 1096	• DUT responds with a <i>DutyCycleAns</i> command, adjusts its duty cycle as requested by the TCL and transmits the uplink frames less frequently after the <i>DutyCycleReq</i> is sent.
1097	



Step	Procedure		Frame Sequence	Test
				Purpose
		End Device – TCL	Frame	
1	DUT sends Unconfirmed frame	→		
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq DataRate = Max125KHzDR, refer [2] Payload = [0x]03XXXXXXXX	Set DataRate to Max125kHzD R
			ChMaskCntl: DC = 0, FC = 6	
			ChMask: DC - Enable only default channels FC = [0x]00FF	
2	DUT sends Unconfirmed frame	→	MAC-CMD LinkADRAns Payload = [0x]0307 Time of Arrival = A	
3	DUT sends Unconfirmed frame	→	Time of Arrival = B	
	The TCL sends Unconfirmed frame	+	MAC-CMD DutyCycleReq Payload = [0x]0407	
4	DUT sends Unconfirmed frame)	MAC-CMD DutyCycleAns Payload = [0x]04	Max Duty Cycle set to 7
5	DUT sends Unconfirmed frame	→	Time of Arrival = C Time of Arrival = D Verify that ((B - A) + 5 sec) < (D - C)	Uplink frames are sent less frequently after the duty cycle is changed.
6	DUT sends Unconfirmed frame	→		
	The TCL sends Unconfirmed frame	+	MAC-CMD DutyCycleReq Payload = [0x]0400	
7	DUT sends Unconfirmed frame	→	MAC-CMD DutyCycleAns Payload = [0x]04	Revert to default Duty cycle

2.5.10. DeviceTimeReq

- The **DUT** is triggered to request the **TCL** for the current network time and the **TCL** must correctly send the network time for the **DUT** to synchronize its time.
- TCL must trigger the **DUT** to send a *DeviceTimeReq* to the TCL. The TCL must reply with the *DeviceTimeAns* providing the current network time to the **DUT**. The **DUT** must return the value received on *DeviceTimeAns* in the Payload of the next uplink.
- 1106 Verify

- **DUT** sends a *DeviceTimeReq* to the **TCL**.
- **DUT** resumes normal operation after **TCL** sends *DeviceTimeAns*.



1110

1111

1112

1113

1114

1115

1116

1117

1118

1119

1120

1121

1122

1123

1124

1125

1126

1127

1128

1129

1130

11311132

1133

11341135

2.5.10.a. Test Procedure Frame Sequence Chart

Step	Procedure		Test Purpose	
		End Device - TCL	Frame	
1	DUT sends Unconfirmed frame	\rightarrow		
2	TCL sends Unconfirmed frame	+	CP-CMD DeviceTimeReq FPort = 224 Payload = [0x]21	
3	DUT sends Unconfirmed frame	→	MAC-CMD DeviceTimeReq Payload = [0x]0D	
4	The TCL sends Unconfirmed frame	+	MAC-CMD DeviceTimeAns Payload = [0x]0DXXXXXXXXX	DeviceTimeA ns sent
5	DUT sends Unconfirmed frame	\rightarrow		

2.5.11. RX Window test

2.5.11.a. RX1 Receive Window Test

This test validates the **DUT**'s capability to receive data on RX1 for data rates as specified in RX1DRoffset table in [2].

For each combination of Operating Uplink DR to RX1 offset:

- The TCL commands the DUT to implement the target configuration and subsequently validates corresponding LinkADRAns and RXParamSetupAns from the DUT indicating it has implemented the target configuration.
- The TCL then sends 5 consecutive downlinks to the DUT on the RX1 window where:
 - At least one of the frames is the maximum allowed payload for the current RX1
 DR (with respect to the current offset).
 - At least one of the frames is confirmed, requiring the **DUT** to set the ACK bit in the subsequent uplink.
- The **TCL** validates reception by verifying for each downlink that the subsequent uplink contains a payload indicating the **DUT**'s downlink counter has incremented by one. The uplink following the confirmed downlink must have its ACK bit set to true as well.

For the given RX1DRoffset, the **TCL** then transitions the **DUT** through the remaining Uplink DR settings, verifying the respective *LinkADRAns* and repeating the above procedure for each.

Once the combinations for the current RX1DRoffset are exhausted, the **TCL** commands the **DUT** to the next RX1DRoffset, verifies the *RXParamSetupAns* and repeats the same series of downlink tests. The **TCL** proceeds in this fashion exhausting all combinations.

For each unique RX1 DR in the RX1DRoffset table, the **TCL** will perform the Sufficient Reception test. For these tests RX1DRoffset will be set to default. The **TCL** sends 60 downlink packets in the RX1 window and verifies that at least 57 frames were received by the **DUT**.





1136 **2.5.11.a.i.** Test Procedure Frame Sequence Chart

LoRaWAN 1.0.4 End Device Certification Page 130 of 162
Requirements for All Regions Version 1.0
Copyright © 2020 LoRa Alliance, Inc. All rights reserved.



Step	Procedure		Frame Sequence	Test Purpose
		End Device - TCL	Frame	i uiposo
1	DUT sends Unconfirmed frame	\rightarrow		
	For DC only – If the DUT supports optional data rate(s), then The TCL sends Unconfirmed frame	←	CMD NewChannelReq ChannelIndex = any unused optional channel Frequency = any allowed frequency for the channel, refer [2] Payload =	
2	If the NewChannelReq was sent by	→	Ox07XXXXXXXXXX Note: If the DUT supports more than one optional data rate, a NewChannelReq must be sent for each of them. CMD NewChannelAns	Added new
_	the TCL, then DUT sends Unconfirmed frame		Payload = [0x]0703[Repeat for all channel added]	channel(s) for the optional data rate(s) supported
	For each RX1DRoffset combination	R		
	in the RX1DRoffset table in [2], the	Steps 3-9 for		
	TCL transitions the DUT to each	[All RX1DR		
	Uplink Data Rate by repeating below Steps 3-9.	in RX1DRoffse t table]		
3	DUT sends Unconfirmed frame	→ →		
J	The TCL sends Unconfirmed frame	*	MAC-CMD LinkADRReq DataRate = X (where X = DR as defined in the RX1DRoffset table in [2], if DR is supported) ChMaskCntl: DC = 0, FC = 6 ChMask: DC - Enable only default channels FC = [0x]00FF	
			MAC-CMD RxParamSetupReq RX1DRoffset = Offset value as defined in the RX1DRoffset table in [2] RX2DataRate = default DR, refer [2] RX2Frequency = default frequency, refer [2] Payload = [0x]03XXXXXXXXXXX[0x]05XXX XXXXX	



Step	Procedure		Frame Sequence	Test Purpose
		End Device - TCL	Frame	•
4	DUT sends Unconfirmed frame	\rightarrow	MAC-CMD LinkADRAns	
			1440 0140	
			MAC-CMD RxParamSetupAns	
			TXI aramoetupAns	
			Payload = [0x]0307[0x]0507	
	The TCL sends Unconfirmed frames	←	CP-CMD RxAppCntReq	
			FPort = 224	
		_	Payload = [0x]09	
5	DUT sends Unconfirmed frame	\rightarrow	CP-CMD RxAppCntAns	
			FPort = 224 Payload = [0x]09XXXX	
			RxAppCnt = x	
	The TCL sends Unconfirmed frames	+	CP-CMD	
			RegionalDutyCycleCtrlReq-	
			OFF	
			FPort = 224 Payload = [0x]0500	
6	DUT sends Unconfirmed frame	→ R [4]	Fayload = [0x]0500	
	De l'editae ellectimine manie) IV[1]		
	Repeat 4 times			
	The TCL sends Unconfirmed frames	← R [4]	CP-CMD TxFramesCtrlReq	
	on RX1 window		FPort = 224	
	Repeat 4 times		Frame type = No change Payload = [0x]0700	
	Nepeat 4 times			
		_		
7	DUT sends Unconfirmed frame	→ ←		
	The TCL sends a Confirmed frame on RX1 window	_	Payload = Max allowed payload as defined in	
	OITTOXT WINGOW		Maximum Payload size table	
			in [2]	
8	DUT sends Unconfirmed frame	\rightarrow	ACK Bit = True	
	The TCL sends Unconfirmed frames	+	CP-CMD RxAppCntReq	
			FPort = 224	
9	DUT sends Unconfirmed frame	→	Payload = [0x]09 CP-CMD RxAppCntAns	
3	201 301103 Officerimined frame	_	FPort = 224	
			Payload = [0x]09XXXX	
			RxAppCnt >= x + 4	
	Repeat steps 3-9 for each			
	RX1DRoffset combination in the RX1DRoffset table in [2].			
	TOTALISE LADIE III [2].	R		
		Steps 10-15		
	For each unique RX1 Data Rate in	for		
	the RX1DRoffset table in [2], the	[All RX1DR		
	TCL will perform the Sufficient	in RX1DRoffse		
	Reception test by repeating Steps 10-15.	t table]		
10	DUT sends Unconfirmed frame	→ ·		
		l	1	



Step	Procedure		Frame Sequence	Test Purpose
		End Device - TCL	Frame	
	The TCL sends Unconfirmed frame	<u>+</u>	MAC-CMD LinkADRReq DataRate = X (where X = DR as defined in the RX1DRoffset table in [2], if DR is supported) ChMaskCntl: DC = 0, FC = 6 ChMask: DC - Enable only default channels FC = [0x]00FF MAC-CMD RxParamSetupReq RX1DRoffset = 0 Payload =	
11	DUT sends Unconfirmed frame	→	[0x]03XXXXXXXX[0x]05XXX XXXXX MAC-CMD LinkADRAns	
			MAC-CMD RxParamSetupAns Payload = [0x]0307[0x]0507	
	The TCL sends Unconfirmed frames	+	CP-CMD RxAppCntReq FPort = 224 Payload = [0x]09	
12	DUT sends Unconfirmed frame	→	CP-CMD RxAppCntAns FPort = 224 Payload = [0x]09XXXX RxAppCnt = y	
13	DUT sends Unconfirmed frame	→ R [60]		
	Repeat 60 times The TCL sends Unconfirmed frame on RX1 window Repeat 60 times	← R [60]	CP-CMD TxFramesCtrlReq DataRate = X FPort = 224 Frame type = No change Payload = [0x]0700 (MaxLen for RX1 DR)	
14	DUT sends Unconfirmed frame The TCL sends Unconfirmed frames	<i>→</i> ←	CP-CMD RxAppCntReq FPort = 224 Payload = [0x]09	
15	DUT sends Unconfirmed frame)	CP-CMD RxAppCntAns FPort = 224 Payload = [0x]09XXXX RxAppCnt <= y + 60 + 1 RxAppCnt >= y + 57 + 1	Reception failure rate <=5%



Step	Procedure		Frame Sequence	Test Purpose
		End Device - TCL	Frame	
	Repeat steps 10-15 for each unique RX1 Data Rate in the RX1DRoffset table in [2]			
16	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame	←	MAC-CMD LinkADRReq DataRate = Max125kHzDR, refer [2] ChMaskCntl: DC = 0, FC = 6 ChMask: DC - Enable only default channels FC = [0x]00FF MAC-CMD	
			RxParamSetupReq RX1DRoffset = 0 RX2DataRate = default DR, refer [2] Freq = default frequency, refer [2] Payload = [0x]03XXXXXXXX	
17	DUT sends Unconfirmed frame	→	[0x]05XXXXXXXX MAC-CMD LinkADRAns	DUT
"	DOT Serius Officoniimieu frame		MAC-CMD RxParamSetupAns Payload = [0x]0307[0x]0507	reverted to default settings
	The TCL sends Unconfirmed frames	←	CP-CMD RegionalDutyCycleCtrlReq- ON FPort = 224 Payload = [0x]0501	
18	DUT sends Unconfirmed frame	\rightarrow		
	For DC only – If the DUT supports optional data rate(s), then The TCL sends Unconfirmed frame	←	CMD NewChannelReq ChannelIndex = as added in Step 1 Frequency = 0 Payload = 0x07XXXXXXXXX Repeat the command for all channels added in Step 1 of this table	
19	If the NewChannelReq was sent by the TCL, then DUT sends Unconfirmed frame	→	CMD NewChannelAns Payload = 0x0703[Repeat for all optional channels added]	Removed channel(s) added





1139	2.5.11.b. RX2	Receive Window Test
1140	This test valid	ates the DUT 's capability to receive data on RX2 for all data rates.
1141	For each uniq	ue RX2DataRate, the TCL will perform the Sufficient Reception. For these tests
1142	RX1DROffset v	will be set to default. The TCL sends 60 downlink packets in the RX2 window and
1143	verifies that at	t least 57 messages were received by the DUT .
1144	2.5.11.b.i.	Test Procedure Message Sequence Chart
1145		



Step	Procedure	Me	essage Sequence	Test Purpose
		End Device	Message	T di poco
		- TCL		
1	DUT sends Unconfirmed frame	→		
	For each RX2DataRate supported	R		
	by the DUT, the TCL will perform the	Steps 2-7 for		
	Sufficient Reception test by	[All RX2DR]		
	repeating Steps 2-7.	,		
2	DUT sends Unconfirmed frame	→ ←	MAG OMB	
	The TCL sends Unconfirmed frame	_	MAC-CMD	
			RxParamSetupReq RX1DROffset = default,	
			refer [2]	
			RX2DataRate = X (where X	
			= each supported data rate	
			for RX2)	
			RX2 Frequency = default,	
			refer [2]	
			Payload =	
			[0x]05XXXXXXX	
3	DUT sends Unconfirmed frame	\rightarrow	MAC-CMD	
			RxParamSetupAns	
			Payload = [0x]0507	
	The TCL sends Unconfirmed frames	+	CP-CMD RxAppCntReq	
			FPort = 224	
			Payload = [0x]09	
4	DUT sends Unconfirmed frame	\rightarrow	CP-CMD RxAppCntAns	
			FPort = 224	
			Payload = [0x]09XXXX	
-	BUT III (II) D (00)	RxAppCnt = y	
5	DUT sends Unconfirmed frame	→ R [60]		
	The TCL sends Unconfirmed frame	← R [60]	CP-CMD TxFramesCtrlReq	
	on RX2 window		FrameType = No change	
			DataRate = X	
	Repeat 60 times		FPort = 224	
			Payload = [0x]0700 (MaxLen	
	For each unique RX2 Data Rate, the		for RX2 DR)	
	TCL will perform the Sufficient			
6	Reception test. DUT sends Unconfirmed frame	-2		
U	The TCL sends Unconfirmed frames	→ ←	CP-CMD RxAppCntReq	
	The TOE senus oncommitted harnes		FPort = 224	
			Payload = [0x]09	
7	DUT sends Unconfirmed frame	→	CP-CMD RxAppCntAns	Reception
'	20. conde oncommina name		FPort = 224	failure rate
			Payload = [0x]09XXXX	<=5%
			RxAppCnt \leq y + 60 +1	
			RxAppCnt $>= y + 57 + 1$	
	Repeat above steps 2-7 until all RX2		,	
	DataRates are tested			
8	DUT sends Unconfirmed frame	\rightarrow		



Step	Procedure	Me	Message Sequence		t
				Purpo	se
		End Device	Message		
		- TCL			
	The TCL sends Unconfirmed frame	←	MAC-CMD	Revert	to
			RxParamSetupReq	default	RX2
			RX1DROffset = default,	DR	
			refer [2]		
			RX2DataRate = default,		
			refer [2]		
			RX2 Frequency = default,		
			refer [2]		
			Payload =		
			[0x]05XXXXXXX		
9	DUT sends Unconfirmed frame	\rightarrow	MAC-CMD		
			RxParamSetupAns		
			Payload = [0x]0507		

1147

11481149

2.5.11.c. RX1 and RX2 simultaneous frames

This test validates that when **TCL** sends frames on both RX1 and RX2 windows simultaneously, **DUT** responds to the frame on the RX1 window and rejects the frame on the RX2 window.



1152

11531154

1155

1156

1157

1158

11591160

1161 1162

2.5.11.c.i. Test Procedure Message Sequence Chart

Step	Procedure	M	essage Sequence	Test Purpose
		End Device - TCL	Message	1 0.000
1	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame	+	CP-CMD RxAppCntReq FPort = 224 Payload = [0x]09	
2	DUT sends Unconfirmed frame)	CP-CMD RxAppCntAns FPort = 224 Payload = [0x]09XXXX RxAppCnt = x	
	The TCL sends Unconfirmed frame on RX1 window	+	CP-CMD TxFramesCtrlReq FrameType = No change FPort = 224 Payload = [0x]0700	
	The TCL sends Unconfirmed frame on RX2 window	+	CP-CMD TxFramesCtrlReq FrameType = No change FPort = 224 Payload = [0x]0700	
3	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame	+	CP-CMD RxAppCntReq FPort = 224 Payload = [0x]09	
4	DUT sends Unconfirmed frame)	CP-CMD RxAppCntAns FPort = 224 Payload = [0x]09XXXX RxAppCnt = x + 2	RX2 frame is ignored if DUT received downlink in RX1

1151 2.5.11.d. **RX Oversized Payload**

A follow-up negative test must be performed for each oversized scenario. The Max Payload size for each region is defined in [2]. After commanding the **DUT** such that the target RX DR is achieved, the TCL sends a downlink whose payload is one byte greater than the scenario's maximum. Note the payload content is random (i.e. not the echo command).

The TCL must use both confirmed and unconfirmed oversized frames; as well as target both receive windows.

Verify

 The **DUT** continues normal operation in the presence of oversized downlinks. DUT must silently discard the oversized downlinks.

2.5.11.d.i. Test Procedure Frame Sequence Chart

LoRaWAN 1.0.4 End Device Certification Page 138 of 162 Requirements for All Regions Version 1.0 Copyright © 2020 LoRa Alliance, Inc. All rights reserved.



Step	Procedure	ı	Test Purpose	
		End Device - TCL	Frame	
1	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frames	+	CP-CMD RegionalDutyCycleCtrlReq- OFF FPort = 224 Payload = [0x]0500	
	Repeat Steps 2-7 'i' times, where i =	R		
	all possible combinations in the Maximum Payload size table in [2]	Steps 2-7 for [All DR in Max Payload table]		
2	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame	←	MAC-CMD LinkADRReq DataRate = X (where X = Each DR as defined in the Maximum Payload size table in [2]) ChMaskCntl: DC = 0, FC = 6 ChMask: DC - Enable only default channels FC = [0x]00FF MAC-CMD RxParamSetupReq RX1DRoffset = 0 RX2DataRate = Y (where Y = Each RX2DataRate as defined in the Maximum Payload size table in [2]) Freq = default frequency [2] Payload = [0x]03XXXXXXXXXX[0x]05XXX	
3	DUT sends Unconfirmed frame	→	XXXXX MAC-CMD LinkADRAns MAC-CMD RxParamSetupAns Payload = [0x]0307[0x]0507	
	The TCL sends Unconfirmed frame on RX1 window	+	CP-CMD TxFramesCtrlReq FPort = 224 FrameType = Confirmed Payload (i) = [0x]0702(MaxLen + 1 for UL DR-X)	



Step	Procedure	[Frame Sequence	Test
				Purpose
		End Device - TCL	Frame	
4	DUT sends Unconfirmed frame	→		DUT discards the oversized frame and sends Unconfirmed frame
	The TCL sends Unconfirmed frame on RX2 window	+	CP-CMD TxFramesCtrlReq FPort = 224 FrameType = No change Payload (i) = [0x]0700 (MaxLen + 1 for UL DR-Y)	
5	DUT sends Unconfirmed frame	→		DUT continues normal operation
	The TCL sends Confirmed frame on RX1 window	←	CP-CMD TxFramesCtrlReq FPort = 224 FrameType = No change Payload (i) = [0x]0700 (MaxLen + 1 for UL DR-X)	
6	DUT sends Unconfirmed frame	→	No ACK	DUT continues normal operation (RX1)
	The TCL sends Confirmed frame on RX2 window	+	CP-CMD TxFramesCtrlReq FPort = 224 FrameType = No change Payload (i) = [0x]0700 (MaxLen + 1 for UL DR-Y)	
7	DUT sends Unconfirmed frame	→	No ACK	DUT continues normal operation (RX2)
	Repeat above steps 2-7 until all Data Rates in the Max Payload table are tested			
8	DUT sends Unconfirmed frame	\rightarrow		



Step	Procedure	Frame Sequence		Test
				Purpose
		End Device	Frame	
		- TCL		
	The TCL sends Unconfirmed frame	←	MAC-CMD LinkADRReq	
			DataRate = Max125kHzDR,	
			refer [2]	
			ChMaskCntl:	
			DC = 0,	
			FC = 6	
			ChMask:	
			DC - Enable only default	
			channels	
			FC = [0x]00FF	
			MAC-CMD	
			RxParamSetupReq	
			RX1DRoffset = 0	
			RX2DataRate = default, refer	
			[2]	
			Freq = default frequency,	
			refer [2]	
			Payload =	
			[0x]03XXXXXXXX[0x]05XXX	
			XXXXX	
9	DUT sends Unconfirmed frame	\rightarrow	MAC-CMD LinkADRAns	DUT
			MAC-CMD	reverted to
			RxParamSetupAns	default
			Payload = $[0x]0307[0x]0507$	settings
	The TCL sends Unconfirmed frames	←	CP-CMD	
			RegionalDutyCycleCtrlReq-	
			ON	
			FPort = 224	
			Payload = [0x]0501	
10	DUT sends Unconfirmed frame	\rightarrow		

1164

1165

1166

11671168

1169

1170

1171

1172

1173

1174

2.5.11.e. **Maximum Allowed Payload**

The purpose of this test is to validate the maximum allowable payload sizes in uplink transmissions. The test depends on properly functioning *LinkADRReq* and *RXParamSetupReq* MAC command control. The first part validates the maximum uplink payload for each of the five uplink data rates against both receive windows, testing both positive and negative scenarios. The second part validates the maximum downlink payload for the various RX1DR offsets as well as various RX2 data rates.

The **TCL** commands the **DUT** with some combination of the following settings and validates the device successfully implements them before proceeding to test max payload handling:

- RX1DRoffset = 0
- RX2DataRate = Maximum default Data Rate, refer [2]
- 1175 The Maximum Payload size for each region is defined in [2].





1176	2.5.11.e.i. Max Payload via Echo
1177	These tests are performed on both RX1 and RX2 Windows. For each 'Uplink DR',
1178	the TCL first commands the DUT to use the target 'Uplink DR' and verifies the
1179	configuration. It then sends an echo command ([0x]08) whose payload size is the
1180	current maximum allowed for the uplink DR as defined in [2].
1181	
1182	Verify
1183	 The DUT responds to each echo command with an echo answer.
1184	 The payload size is in fact the maximum for the DUT's current DR and its
1185	content is correct.
1186	2.5.11.e.i.1. Test Procedure Frame Sequence Chart
1187	



Step	Procedure	Frame Sequence		Test Purpose
		End Device - TCL	Frame	, angest
1	DUT sends Unconfirmed frame	\rightarrow		
·	The TCL sends Unconfirmed frames	*	CP-CMD RegionalDutyCycleCtrlReq- OFF FPort = 224 Payload = [0x]0500	
	Repeat Steps 2-5 'i' times, where i =	R	r dylodd - [exjeddo	
	all possible combinations in the Maximum Payload size table in [2]	Steps 2-5 for [All DR in Max payload table]		
2	DUT sends Unconfirmed frame	→		
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq DataRate = X (where X = Each DR as defined in the Maximum Payload size table in [2]) ChMaskCntl: DC = 0, FC = 6 ChMask: DC - Enable only default channels FC = [0x]00FF MAC-CMD RxParamSetupReq RX1DRoffset = 0 RX2DataRate = Maximum default DR, refer [2] Freq = default frequency [2] Payload = [0x]03XXXXXXXXXXX[0x]05XXX XXXXX	
3	DUT sends Unconfirmed frame	→	MAC-CMD LinkADRAns MAC-CMD RxParamSetupAns Payload = [0x]0307[0x]0507 DataRate = X	
	The TCL sends Unconfirmed frame on RX1 window	+	CP-CMD EchoPayloadReq FPort = 224 Payload (i) = [0x]08XX (MaxLen for UL DR-X)	
4	DUT sends Unconfirmed frame	>	DataRate = X CP-CMD EchoPayloadAns FPort = 224 Payload (i)' = [0x]08XX (MaxLen for DR-X)	DUT echos MaxLen PDU for each Uplink DR on RX1



Step	Procedure	Frame Sequence		Test Purpose
		End Device - TCL	Frame	
	The TCL sends Unconfirmed frame on RX2 window	+	CP-CMD EchoPayloadReq FPort = 224 Payload (i) = [0x]08XX (MaxLen for UL DR-X)	
5	DUT sends Unconfirmed frame)	DataRate = X CP-CMD EchoPayloadAns FPort = 224 Payload (i)' = [0x]08XX (MaxLen for DR-X)	DUT echos MaxLen PDU for each Uplink DR on RX2
	Repeat above Steps 2-5 for all DR in Max payload size table			
6	The TCL sends Unconfirmed frame The TCL sends Unconfirmed frame	→	MAC-CMD LinkADRReq DataRate = Max125kHzDR, refer [2] ChMaskCntl: DC = 0, FC = 6 ChMask: DC - Enable only default channels FC = [0x]00FF MAC-CMD RxParamSetupReq RX1DRoffset = 0 RX2DataRate = default DR, refer [2] Freq = default frequency, refer [2] Payload = [0x]03XXXXXXXXXX[0x]05XXX XXXXX	
7	DUT sends Unconfirmed frame	→	MAC-CMD LinkADRAns MAC-CMD RxParamSetupAns Payload = [0x]0307[0x]0507	DUT reverted to default settings
	The TCL sends Unconfirmed frames	←	CP-CMD RegionalDutyCycleCtrlReq- ON FPort = 224 Payload = [0x]0501	
8	DUT sends Unconfirmed frame	\rightarrow		





1190	2.5.11.e.ii. Oversized Payload via Echo
1191	These tests are performed on RX2 Window only. For each 'Uplink DR', the TCL first
1192	commands the DUT to use the target 'Uplink DR' and verifies the configuration. It
1193	then sends an echo command ([0x]08) whose payload size is one byte greater than
1194	the current maximum allowed for the uplink DR. The Maximum Payload size for
1195	each region is defined in [2].
1196	Verify
1197	 The DUT does not uplink an echo response for any of the echo commands
1198	due to uplink size limitation. It may either respond with an uplink that
1199	has no payload, or silently discard the echo-command.
1200	 Subsequent to the first uplink sent after receiving the command, uplinks
1201	must resume containing only the default payload (downlink counter).
1202	2.5.11.e.ii.1. Test Procedure Frame Sequence Chart
1203	
1204	



Step	Procedure	F	Test Purpose	
		End Device - TCL	Frame	•
1	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frames	+	CP-CMD RegionalDutyCycleCtrlReq- OFF FPort = 224 Payload = [0x]0500	
	Repeat Steps 2-4 'i' times, where i =	R		
	all possible combinations in the Maximum Payload size table in [2]	Steps 4-6 [All DR in Max Payload table]		
2	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame	←	MAC-CMD LinkADRReq DataRate = X (where X = Each DR as defined in the Maximum Payload size table in [2]) ChMaskCntl: DC = 0, FC = 6 ChMask: DC - Enable only default channels FC = [0x]00FF MAC-CMD RxParamSetupReq RX1DRoffset = 0 RX2DataRate = Maximum default DR, refer [2] Freq = default frequency, refer [2] Payload = [0x]03XXXXXXXXXX[0x]05XX XXXXX	
3	DUT sends Unconfirmed frame)	MAC-CMD LinkADRAns MAC-CMD RxParamSetupAns Payload = [0x]0307[0x]0507 DataRate = X	
	The TCL sends Unconfirmed frame on RX2 window	+	CP-CMD EchoPayloadReq FPort = 224 Payload (i) = [0x]08XX (MaxLen + 1 for UL DR-X)	



Step	Procedure	F	Test	
				Purpose
		End Device - TCL	Frame	
4	DUT sends Unconfirmed frame)	Payload does not exceed limits	DUT continues normal operation in the presence of oversized downlinks after each Uplink DR on RX2
	Repeat above Steps 2-4 for all DR in Max payload size table			
5	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame	←	MAC-CMD LinkADRReq DataRate = Max125kHzDR, refer [2] ChMaskCntl: DC = 0; FC = 6 ChMask: DC - Enable only default channels FC = [0x]00FF MAC-CMD RxParamSetupReq RX1DRoffset = 0 RX2DataRate = default DR, refer [2] Freq = default frequency, refer [2] Payload = [0x]03XXXXXXXXXX[0x]05XX XXXXXX	
6	DUT sends Unconfirmed frame	→	MAC-CMD LinkADRAns MAC-CMD RxParamSetupAns Payload = [0x]0307[0x]0507	DUT reverted to default settings
	The TCL sends Unconfirmed frames	+	CP-CMD RegionalDutyCycleCtrlReq- ON FPort = 224 Payload = [0x]0501	
7	DUT sends Unconfirmed frame	\rightarrow		





1207 2.5.12. MAC Command(s) in App-Payload and/or Frame Options

These tests ensure a **DUT** appropriately accepts and processes or discards a downlink whose contents include one or more MAC commands in the App-Payload (i.e. FRMPayload) and/or the Frame options (FOpts) portion of the frame.

2.5.12.a. App-Payload Only (FPort = 0)

Perform the test listed below such that all MAC commands are sent to the **DUT** within the App-Payload portion of a single downlink. Successful completion of each test indicates the MAC Command payload is properly decrypted using the Network Session Key and processed by the **DUT**. It is up to the **DUT** to choose how it answers: it can choose either the FOpts field or the FRMPayload. Some devices for example may use the FRMPayload when the MAC command answers are larger than the 15 bytes limit of the FOpts field.

- DevStatusReq
- RxParamSetupReq
- RxTimingSetupReq
- 1221 LinkADRReq

1211

12121213

1214

1215

1216 1217

1218

1219



1222 2.5.12.a.i. Test Procedure Frame Sequence Chart

Step	Procedure		Frame Sequence		
		End Device - TCL	Frame	Purpose	
1	DUT sends Unconfirmed frame FCntUp = n	→			
	The TCL sends Unconfirmed frame	←	App-Payload MAC-CMD1 DevStatusReq MAC-CMD2 RxParamSetupReq MAC-CMD3 RxTimingSetupReq MAC-CMD4 LinkADRReq Payload = [0x]06[0x]05XXXXXXXXX[0x]0 8XX[0x]03XXXXXXXX		
			FPort = 0		
2	DUT sends Unconfirmed frame FCntUp = n + 1	→	MAC-CMD1 DevStatusAns MAC-CMD2 RxParamSetupAns MAC-CMD3 RxTimingSetupAns MAC-CMD4 LinkADRAns Payload = [0x]06XXXX[0x]0507[0x]08[0 x]0307	Successful completion of each command	
3	The TCL sends Unconfirmed frame	+	CP-CMD TxFramesCtrlReq FrameType = No change FPort = 224 Payload = [0x]0700		
	DUT sends Unconfirmed frame FCntUp = n + 2	→			

1223

1224

12251226

1227

1228

1229

1230

1231

1232

1233

2.5.12.b. Frame Options Only (FPort NOT = 0)

Perform the test listed below such that all MAC commands are sent to the **DUT** within the Frame Options field (i.e. FOpts) portion of a single downlink. Successful completion of each test indicates the MAC Command payload is properly decrypted using the Application Session Key and processed by the **DUT**. It is up to the **DUT** to choose how it answers: it can choose either the FOpts field or the FRMPayload. Some devices for example may use the FRMPayload when the MAC command answers are larger than the 15 bytes limit of the FOpts field.

- DevStatusReq
- RxParamSetupReq
- RxTimingSetupReq
- 1234 LinkADRReq



1235 **2.5.12.b.i. Test Procedure Frame Sequence Chart**

Step	Procedure Frame Sequence			Test
		End Device - TCL	Frame	Purpose
1	DUT sends Unconfirmed frame FCntUp = n	→		
	The TCL sends Unconfirmed frame	+	Frame Options MAC-CMD1 DevStatusReq MAC-CMD2 RxParamSetupReq MAC-CMD3 RxTimingSetupReq MAC-CMD4 LinkADRReq Payload = [0x]06[0x]05XXXXXXXXX[0x]0 8XX[0x]03XXXXXXXX	
2	DUT sends Unconfirmed frame		FPort NOT = 0 MAC-CMD1 DevStatusAns	Successful
2	FCntUp = n + 1	→	MAC-CMD1 DevStatusAns MAC-CMD2 RxParamSetupAns MAC-CMD3 RxTimingSetupAns MAC-CMD4 LinkADRAns Payload = [0x]06XXXX[0x]0507[0x]08[0 x]0307	completion of each command
3	The TCL sends Unconfirmed frame	←	CP-CMD TxFramesCtrlReq FrameType = No change FPort = 224 Payload = [0x]0700	
	DUT sends Unconfirmed frame FCntUp = n + 2	→		

2.5.12.c. App-Payload and Frame Options
 Verify the DUT ignores a downlink when MAC commands are simultaneously present in the App Payload and Frame options (FOpts) fields.

1240 **2.5.12.c.i. Test Procedure Frame Sequence Chart** 1241

1239

LoRaWAN 1.0.4 End Device Certification Page 150 of 162 Requirements for All Regions Version 1.0 Copyright © 2020 LoRa Alliance, Inc. All rights reserved.



Step	Procedure		Frame Sequence	Test Purpose
		End Device - TCL	Frame	
1	DUT sends Unconfirmed frame FCntUp = n	→		
	The TCL sends Unconfirmed frames	+	CP-CMD RxAppCntReq FPort = 224 Payload = [0x]09	
2	DUT sends Unconfirmed frame FCntUp = n + 1)	CP-CMD RxAppCntAns FPort = 224 Payload = [0x]09XXXX RxAppCnt = x	
	The TCL sends Unconfirmed frame	+	Frame Options MAC-CMD1 LinkADRReq MAC-CMD2 LinkADRReq MAC-CMD3 RxParamSetupReq Payload = [0x]03XXXXXXXXXXX App-Payload MAC-CMD1' LinkADRReq MAC-CMD2' RxTimingSetupReq MAC-CMD3' DevStatusReq Payload = [0x]03XXXXXXXXXX [0x]08XX[0x]08XX[0x]08XX[0x]06	
3	DUT sends Unconfirmed frame FCntUp = n + 2	→		Downlink ignored when MAC commands are present simultaneou sly in App- Payload and FOpts field
	The TCL sends Confirmed frame	+	Frame Options MAC-CMD1 LinkADRReq MAC-CMD2 LinkADRReq MAC-CMD3 RxParamSetupReq Payload = [0x]03XXXXXXXXX[0x]03XXX XXXXX[0x]05XXXXXXX App-Payload MAC-CMD1' LinkADRReq MAC-CMD2' RxTimingSetupReq MAC-CMD3' DevStatusReq Payload = [0x]03XXXXXXXXXX[0x]08XX[0 x]06	



4	DUT sends Unconfirmed frame	\rightarrow		Downlink
	FCntUp = n + 3			ignored
				when MAC
				commands
				are present
				simulaneousl
				y in App-
				Payload and
				FOpts field
	The TCL sends Unconfirmed frames	←	CP-CMD RxAppCntReq	
			FPort = 224	
			Payload = [0x]09	
5	DUT sends Unconfirmed frame	\rightarrow	CP-CMD RxAppCntAns	Downlink
	FCntUp = n + 4		FPort = 224	counter not
			Payload = [0x]09XXXX	incremented
			RxAppCnt = x + 1	with ignored
				downlinks

1243

2.5.13. Incorrect MAC Commands

- 1244 These tests ensure that a **DUT** behaves normally after it receives incorrect MAC commands from the
- 1245 **TCL.**
- 1246 The invalid commands are:
- LinkADRReq with value out of spec Payload: [0x]0380000000
- Incomplete MAC command Payload: [0x]03010000
- Post Unknown MAC command ID Payload: [0x]7F
- Valid MAC command followed by invalid MAC commands Payload: [0x]0603010000
- 1251 Verify:
- The **DUT** continues normal operation after receiving the invalid commands

1253

1254 2.5.13.a. **Test Procedure Frame Sequence Chart**



Step	Procedure		Frame Sequence	Test Purpose
		End Device - TCL	Frame	
1	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkADRReq Payload = [0x]0380000000 FPort = 0	
2	DUT sends Unconfirmed frame	→	LinkADRAns NOT = OK	DUT continues normal operation after receiving LinkADRReq with 'out of spec' Payload
	The TCL sends Unconfirmed frame	←	MAC-CMD LinkADRReq Payload = [0x]03010000 FPort = 0	
3	DUT sends Unconfirmed frame	→	No response	DUT continues normal operation after incomplete MAC command Payload
	The TCL sends Unconfirmed frame	←	MAC-CMD Payload = [0x]7F FPort = 0	
4	DUT sends Unconfirmed frame	→	No response	DUT continues normal operation after unknown MAC command Payload
	The TCL sends Unconfirmed frame	←	MAC-CMD1 DevStatusReq MAC-CMD2 incomplete LinkADRReq	
			Payload = [0x]0603010000 FPort = 0	
5	DUT sends Unconfirmed frame	→	MAC-CMD1 DevStatusAns Payload = [0x]06XXXX	DUT continues normal operation after incomplete MAC command Payload
	The TCL sends Unconfirmed frame	+	MAC-CMD1 LinkADRReq	
			MAC-CMD2 [0x]7F	
			CMD3 DevStatusReq	
			FPort = 0	
6	DUT sends Unconfirmed frame	→	MAC-CMD1 LinkADRAns Payload = [0x]03XXXXXXXX	DUT answers only the first MAC command and continues normal operation if the second MAC command is invalid
7	Wait for a <u>Dynamic channel</u> : maximum of 5 * (number of channels configured) <u>Fixed channel</u> : maximum of 2 * (number of channels configured) uplink packets to be sent.	→ R [5*NbCh DC] or [2*NbChF C] or [AllCh used]	For DC plan devices: Only default channels must be used. The additional channel must not be added. For FC plan devices: The DUT must send an uplink on any of the channels enabled.	



1257 **2.5.14.** Multiple MAC commands prioritization

This test verifies that when a combination of application payload and MAC answers, or new MAC commands are sent by the DUT, the priority for including information in the frame is as shown below.

Priority Level	Information type
Highest	MAC answers
	New MAC commands
Lowest	Application payload

1261 Table 4: Transmit data insertion prioritization1262

1263 1264

1260

Additionally, if the commands cannot fit in the same frame due to size restrictions, the message must be truncated.

12651266

1267 Verification summary:

- Within a single frame, the DUT must send all higher-priority information before sending any
 lower- priority information.
- If the MAC command buffer is too large to fit in the frame, the DUT must truncate the buffer at the end of the last MAC command that is able to fit within the frame.
 - The DUT must execute the full list of MAC commands even if the buffer containing the MAC answers is truncated
- 1274 2.5.14.a. **Test Procedure Frame Sequence Chart**

1275

1272



Step	Procedure		Frame Sequence		
				Purpose	
			End Device - TCL	Frame	
1	DUT sends Unconfirmed frame	→			
	The TCL sends Unconfirmed frame	+	FPort = 224 CP-CMD LinkCheckReq		
			MAC-CMD DevStatusReq		
			MAC-CMD LinkADRReq DataRate = Max125kHzDR		
			Payload = [0x]20 FOpts = [0x]06[0x]03XXXXXXX		
2	DUT sends Unconfirmed frame	→	MAC-CMD DevStatusAns MAC-CMD LinkADRAns MAC-CMD LinkCheckReq Payload = [0x]06XXXX[0x]0307[0x]02	DUT prioritises MAC answers ove new MAC commands and application payload	
	The TCL sends Unconfirmed frame	+	MAC-CMD LinkCheckAns		
			Payload = [0x]02XXXX		
3	DUT sends Unconfirmed frame	→			
	The TCL sends Unconfirmed frame Note: This step is required for only regions with Dwell Time limitation	+	MAC-CMD TXParamSetupReq UplinkDwellTime = 0 Payload = [0x]09XX	Set TXParamSet upReq for regions with Dwell time limitation	
4	DUT sends Unconfirmed frame	→	For regions with Dwell time limitation only		
			MAC-CMD TXParamSetupAns Payload = [0x]09		
			DataRate = Max125kHzDR		



	TOL also condo libraritimas diferent		MAC CMD4 Day Otativa Da	
	TCL also sends Unconfirmed frame	←	MAC-CMD1 DevStatusReq	
			MAC-CMD2	
			RxParamSetupReq	
			Repeat the MAC-CMD	
			DevStatusReq until the MAC	
			command uplink response	
			buffer would be full for	
			MinDR, refer [2]	
			MAC-CMDX LinkADRReq	
			(with DataRate = MinDR)	
			F , F 54000	
			For example: For EU863-	
			870, the Max payload size is	
			51 bytes for MinDR. Hence	
			repeat the DevStatusReq command 15 times to ensure	
			the MAC command response	
			buffer is greater than 51	
			bytes.	
			Payload =	
			[0x]06[0x]05XXXXXXXX[0x]0	
			6[Repeat as	
			required][0x]03XXXXXXXX	
5	DUT sends Unconfirmed frame	\rightarrow	DataRate = MinDR	Successful
				completion of
			MAC-CMD1 DevStatusAns	all MAC
			MAC-CMD2	commands in
			RxParamSetupAns	the correct
			MAC-CMD3 DevStatusAns	sequence.
				Message
			MAC-CMDX DevStatusAns	truncated.
			Doylood -	
			Payload = [0x]06XXXX[0x]0507[0x]06X	
			XXX[0x]06XXXX	
			700[0],0070000	
			DUT truncates the MAC	
			command when max	
			payload size is exceeded.	
			The sequence of the	
			response must be exactly	
			the same as described.	
			T	
			The LinkADRAns is not sent	
			in the response as it must be	
			truncated due to payload	
			size restrictions. However,	
•			the DR must be set to	
			the DR must be set to MinDR.	



	TCL also sends Unconfirmed frame	←	MAC-CMD1 DevStatusReq	
			MAC-CMD2	
			RxParamSetupReq	
			Repeat the MAC-CMD DevStatusReq until the MAC command uplink response buffer would be full for MinDR, refer [2] (i.e. same number of DevStatusReq commands sent in the previous step)	
			MAC-CMDX LinkADRReq (with DataRate = Max125kHzDR)	
			Payload = [0x]06[0x]05XXXXXXXX[0x]0 6[Repeat as required][0x]03XXXXXXXX	
6	DUT sends Unconfirmed frame	→	DataRate = Max125kHzDR MAC-CMD1 DevStatusAns MAC-CMD2 RxParamSetupAns MAC-CMD3 DevStatusAns MAC-CMDX DevStatusAns MAC-CMDX+1 LinkADRAns Payload = [0x]06XXXX[0x]0507[0x]06X XXX[0x]06XXXX[0x]0307	Successful completion of all commands. Message is not truncated.
			DUT must not truncate the frame.	

3. FPort 224 Deactivation

This test must be performed as the last step ever to be performed on the device. The test lab must ensure that all other tests are completed on the device before performing this test. When the FPort 224 is disabled, it cannot be re-enabled on the device again.

The **TCL** will send a downlink payload message of [0x]07E (*DutFPort224DisableReq*) over port 224, thus disabling FPort 224 for the DUT.

3.1. Test Procedure Message Sequence Chart

1285 1286

1284

1276

1277

12781279

1280 1281

1282 1283







Step	Procedure	Message Sequence		Test Purpose
		End Device - TCL	Message	
1	DUT sends Unconfirmed frame	\rightarrow		
	The TCL sends Unconfirmed frame	←	CP-CMD DutFPort224DisableReq FPort = 224 Payload = [0x]07E	Disable the FPort 224 for the DUT as the last step of the Test cycle
2	If the device is an OTAA device, DUT sends Join-Request frame	→		
	Else, skip to the next step			
	If the device is an OTAA device, TCL sends Join-Accept response	+		Join accepted for OTAA device
3	DUT sends Unconfirmed or Confirmed frame	\rightarrow	FPort = any allowed port except 224	
	The TCL sends Unconfirmed frame	+	CP-CMD TxPeriodicityChangeReq FPort = 224 Periodicity = 5 sec Payload = [0x]0601	Try to set Uplink Periodicity
			If DUT sent Confirmed uplink frame, TCL must Acknowledge	
4	DUT sends Unconfirmed or Confirmed frame	→		
	If this uplink is not received within 2 minutes, then the test can be ended, and the next steps can be skipped.			
	The TCL sends Unconfirmed frame	+	CP-CMD EchoPayloadReq FPort 224 Payload = [0x]08010203 If DUT sent Confirmed uplink frame, TCL must	
			Acknowledge	
5	Wait for a maximum of 3 minutes for the DUT to send an Unconfirmed or Confirmed frame If no uplink is sent, the test can be ended, and the next steps can be skipped.	→	No EchoPayloadAns response received	FPort 224 downlinks are not accepted.
	The TCL sends Unconfirmed frame	+	CP-CMD EchoPayloadReq FPort 224 Payload = [0x]08010203 If DUT sent Confirmed uplink	Repeat the EchoPayloa dReq
			frame, TCL must Acknowledge	





6	Wait for a maximum of 3 minutes.	\rightarrow	No EchoPayloadAns	FPort 224
			response received	downlinks
				are not
				accepted.



4. Test Case Mapping with LoRaWAN Specification [1]

The following table provides the section mapping between the LoRaWAN Specification [1] and this Certification Specification document

LoRaWAN Spec [1] section	This document section	Description
3.3	2.5.11	Receive Windows
4	0 2.4.1 and 2.4.2	MAC Packet Formats
5.1	2.5.7	LinkCheck
5.2	2.5.8	LinkADR
5.3	0	DutyCycle
5.4	2.5.4	RXParamSetup
5.5	2.5.1	DevStatus
5.6	2.5.2 and 2.5.3	NewChannel and DIChannel
5.7	2.5.5	RXTimingSetup
5.8	2.5.6	TXParamSetup
5.9	2.5.10	DeviceTime
6.2	2.2	Over the air Activation
6.3	2.3	Activation by Personalization

Table 5 LoRaWAN Spec vs Certification Spec mapping





5. Notice of Use and Disclosure

Copyright © LoRa Alliance, Inc. (2020). All Rights Reserved.

- The information within this document is the property of the LoRa Alliance (The Alliance) and its use
- and disclosure are subject to LoRa Alliance Corporate Bylaws, Intellectual Property Rights (IPR) Policy
- 1329 and Membership Agreements.
- 1330 Elements of LoRa Alliance specifications may be subject to third party intellectual property rights,
- including without limitation, patent, copyright, or trademark rights (such a third party may or may not
- 1332 be a member of LoRa Alliance). The Alliance is not responsible and shall not be held responsible in any
- manner for identifying or failing to identify any or all such third-party intellectual property rights.
- 1334 This document and the information contained herein are provided on an AS IS basis and THE ALLIANCE
- 1335 DISCLAIMS ALL WARRANTIES EXPRESS OR IMPLIED, INCLUDING BUT NOTLIMITED TO (A) ANY
- 1336 WARRANTY THAT THE USE OF THE INFORMATION HEREINWILL NOT INFRINGE ANY RIGHTS OF THIRD
- 1337 PARTIES (INCLUDING WITHOUT LIMITATION ANY INTELLECTUAL PROPERTY RIGHTS INCLUDING
- 1338 PATENT, COPYRIGHT OR TRADEMARK RIGHTS) OR (B) ANY IMPLIED WARRANTIES OF
- 1339 MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE OR NONINFRINGEMENT.
- 1340 IN NO EVENT WILL THE ALLIANCE BE LIABLE FOR ANY LOSS OF PROFITS, LOSS OF BUSINESS, LOSS OF
- 1341 USE OF DATA, INTERRUPTION OF BUSINESS, OR FOR ANY OTHER DIRECT, INDIRECT, SPECIAL OR
- 1342 EXEMPLARY, INCIDENTIAL, PUNITIVE OR CONSEQUENTIAL DAMAGES OF ANY KIND, IN CONTRACT OR
- 1343 IN TORT, IN CONNECTION WITH THIS DOCUMENT OR THE INFORMATION CONTAINED HEREIN, EVEN
- 1344 IF ADVISED OF THE POSSIBILITY OF SUCH LOSS OR DAMAGE.
- The above notice and this paragraph must be included on all copies of this document that are made.
- 1346 LoRa Alliance, Inc.
- 1347 5177 Brandin Court,
- 1348 Fremont, CA 94538 USA
- 1349

1325

- Note: All Company, brand and product names may be trademarks that are the sole property of their
- 1351 respective owners.
- 1352
- 1353