ThingPark Wireless

Device Interoperability Test Plan

REVISION: 1.6.1

DATE: 20/05/2025

REVISION HISTORY

Version	Date	Author	
0.1	03/02/2014	PCa, OHe	Initial version
0.2	15/06/2015	GRE, DSO	Document review Added class C Added JOIN test
0.3	20/09/2015	OHe	Merged with CORE / RF optimization testing plan
0.4	30/10/2015	GDG	Update testing plan
1.0	04/11/2015	GRE	Renamed PMLT to TPIT Full review of the test plan
1.1	06/11/2015	GRE, GDG	Added TPIT 1.2.2.5, 1.2.2.6 Document review
1.2	24/11/2015	GDG	Reorganization TPIT order
1.3	13/01/2016	GDG	Added LoRa-Alliance End Device Test Software commands
1.3.1	20/01/2016	GDG	Rx2 clarification
1.3.2	12/02/2016	GDG	• Increase to 60 messages uplink/downlink test (20 by SF) • Logical Channel start with LC1 • Add test change Rx2 Frequency • Some frame counter up/down clarification • Remove 0x82 MAC command in test mode
1.3.3	21/04/2016	GDG	Add switch mode ABP/OTAA in test mode
1.4.0	19/05/2016	GDG	Add ETSI and FCC regulation specifications
1.4.1	08/09/2016	GDG	Add TPIT 4.4.2 Delete Channels Update TPIT 3.2.1/3.2.2 Update TPIT 7.1.1
1.5	04/05/2017	RG GDG	Update new branding & LoRaWAN specifications 1.0.2 Review
1.5.1	23/05/2017	GDG	Merge TPIT 4.4.3/4.4.4
1.5.2	08/2017	GDG RG	Add ISM Band KR920, TPIT 2.2.5 description update, TPIT 4.4.3 is also required for EU868 band

1.5.3	11/2018	GDG	Add TPIT 3.2.4 redundancy Add TPIT 4.4.4 Remove default channels Add TPIT 8.1.1 MAC Command LinkCheck Add TPIT 9.1.x MAC Command incorrect
1.6.0	02/2019	GDG	Add Class B tests

LEGAL

This document contains proprietary and confidential material of ACTILITY SA. This document is provided under and governed by either a license or confidentiality agreement. Any unauthorized reproduction, use, or disclosure of this material, or any part thereof, is strictly prohibited.

The material provided in this document is believed to be accurate and reliable. However, no responsibility is assumed by Actility SA for the use of this material. Actility SA reserves the right to make changes to the material at any time and without notice. This document is intended for information and operational purposes only. No part of this document shall constitute any contractual commitment by Actility SA.

Headquarters

Actility Lannion,

Actility S.A 4 rue Ampère

BP 30225

22300 Lannion France

www.actility.com

TABLE OF CONTENTS

1 DEFINITIONS AND ABBREVIATIONS

Term	Definition
ADR	Adaptive Data Rate
AppKey	Device root key used to derive AppSkey and NwkSKey during OTA procedure
AppSKey	Application Session Key
DevEUI	Device EUI, globally unique 64-bit identifier assigned according to the IEEE EUI-64 guidelines
DUT	Device Under Test
LoRa	Long range and low energy radio RF technology developed by Semtech
LC	Logical Channel
LRC	Long Range Concentrator : LoRa network server
LRR	Long Range Relay: LoRa Basestation
NwkSKey	Network Session Key
OSS	Operations Support System
RSSI	Received Signal Strength Indicator
SF	Spreading Factor
SNR	Signal to Noise Ratio

2 REFERENCE DOCUMENTS

Term	Definition
LoRaWAN Specification 1R0	LoRaWAN Specification 1.0
LoRaWAN1.0.1final05Apr2016_1099_1	LoRaWAN Specification 1.0.1
LoRaWAN102-20161012_1398_1	LoRaWAN Specification 1.0.2
LoRaEndDeviceCertificationEU 1.0	LoRa-Alliance End Device Test Software EU
LoRaEndDeviceCertificationNAV	LoRa-Alliance End Device Test Software US

3 INTRODUCTION

This document presents the basic tests to be performed by Actility and device manufacturers using LoRaWAN to roll out their connected objects on the ThingPark platform.

Once the device has passed the interoperability successfully it can use the label “ThingPark Connected” and use this logo:

This document covers the LoRaWAN 1.0.2 specification.

The ThingPark Wireless interoperability test does not replace the LoRa Alliance certification. It is at the responsibility of the device partner to respect the LoRaWAN certification. The “LoRaWAN Certified Device” is not mandatory to pass the ThingPark Wireless Interoperability Test but the ThingPark Wireless Interoperability Test could be a perfect pre-certification test to prepare the LoRa Alliance certification.

There is no exhaustive RF test during the ThingPark Wireless Interoperability Test so the device partner must perform its RF test in accordance with the local regulatory. The RF test is a comparison with a reference device.

3.1 Scope of the ThingPark Wireless Interoperability test

The ThingPark Wireless Interoperability Test will check all the requirements to be compliant with the ThingPark Wireless platform as a Network Server, and then in accordance with all Network Operator deploying over the ThingPark Wireless platform.

The following regions are currently supported:

- EU868
- US915
- AU915
- AS923
- JP920
- KR920
- CN470
- RU864
- IN865

3.2 Test procedure

The test consists as a functional test with the ThingPark LoRaWAN platform. Actility offers a dedicated setup so the device can communicate in a closed environment. This process requires the device to be sent to Actility or one of its approved partners.

A report will be generated at the end of the test with a pass/fail sanction.

4 DEVICE PROFILE

The Device Partner must provide Actility with the application behavior of the sensor. This allows to perform the corresponding test relevant to the device. This is done by filling out a questionnaire:

- Online from the Partner website (https://partners.thingpark.com) using the left side menu under the 'Tests and Approval' section.

5 THINGPARK INTEROPERABILITY TESTS

Not every test is required for all devices to successfully pass the 'ThingPark Connected'. It depends on:

- The LoRaWAN specification the device follows
- The LoRaWAN Region where the device will be operating
- The LoRaWAN class the device is intended to be used for

Prerequisites

- ☐ Testing is performed on devices that have been provisioned on the ThingPark Community platform.
- ☐ DUT should already have been through the 'Auto Testing Tool' successfully.
- ☐ Device partners should have the ability through their own debug tools to view data messages received by DUT.
- ☐ DUT must have the ability to respond to MAC commands at boot.

The following list allows to know which tests are required and will be run as part of the Interoperability tests.

Throughout the test plan, "ETSI" style refers to the LoRaWAN regional profiles that use a limited set of default LoRaWAN channels (for boot) and rely on NewChannelReq/Ans commands to configure additional channels; this is the case for EU868, RU864, AS923, KR920 and IN865 ISM bands. "FCC" style refers to regional profiles having predefined channel lineup, which is the case for US915, AU915 and CN470 profiles.

ID	Description	LoRaWan Specs	Class	Region
1.0.1	Specify the range of DevEUI allocated for the device part number.	1.0+	ALL	ALL
1.1.1	ABP devAddr conformity	1.0+	ALL	ALL
1.1.2	ABP NwkSKey conformity	1.0+	ALL	ALL
1.1.3	ABP MIC validation on uplink	1.0+	ALL	ALL
1.2.1	OTAA AppKey conformity	1.0+	ALL	ALL
1.2.2.1	OTAA Join Request / Join Accept	1.0+	ALL	ALL
1.2.2.4	OTAA resent of Join Request message on non-reception of Join Accept	1.0+	ALL	ALL
1.2.2.6	OTAA Channel List	1.0+	ALL	ETSI
1.2.3	Aggregated Duty Cycle	1.0+	ALL	EU868 / RU864
2.1.1	Uplink Unconfirmed message	1.0+	ALL	ALL
2.1.2	Uplink confirmed message	1.0+	ALL	ALL
2.1.3	Uplink Spreading factor range	1.0+	ALL	ALL
2.1.4	Uplink Packet loss rate	1.0+	ALL	ALL
~~**2.1.5**~~	~~Spreading factor automatic increase~~	~~1.0+~~	~~ALL~~	~~ALL~~
2.2.1	Downlink unconfirmed message	1.0+	ALL	ALL
2.2.2	Downlink confirmed message	1.0+	ALL	ALL
2.2.3	Downlink Spreading factor range	1.0+	ALL	ALL
2.2.4	Downlink Packet loss rate	1.0+	ALL	ALL
2.2.5	RX2 window	1.0+	ALL	ALL
3.1.1	Boot channel conformity	1.0+	ALL	ETSI
3.1.2	Uplink channel usage	1.0+	ALL	ALL
3.2.1	ADR on unconfirmed uplink	1.0+	ALL	ALL
3.2.2	ADR on confirmed uplink	1.0+	ALL	ALL
3.2.3	ADR TxPower change	1.0+	ALL	ALL
3.2.4	ADR Redundancy	1.0+	ALL	ALL
3.3.1	ADR Disabling (for mobile devices)	1.0+	ALL	ALL

ID	Description	LoRaWan Specs	Class	Region
4.1.1	MAC RXTimingSetup	1.0+	ALL	ALL
4.1.1.1	MAC RXTimingSetupAns	1.0.1+	ALL	ALL
4.2.1	MAC RXParamSetup	1.0+	ALL	ALL
4.2.1.1	MAC RXParamSetupAns	1.0.1+	ALL	ALL
4.3.1	MAC DevStatus	1.0+	ALL	ALL
4.4.1	MAC NewChannelReq	1.0+	ALL	ETSI
4.4.2	MAC NewChannelReq deletion	1.0+	ALL	ETSI
4.4.3	MAC DlChannelReq	1.0.2+	ALL	ETSI
4.5.1	Large MAC command	1.0+	ALL	ALL
4.6.1	Uplink Dwell time	1.0.2+	ALL	AS923
4.6.2	Downlink Dwell time	1.0.2+	ALL	AS923
5.1.1	Continuous RX2 window check	1.0+	C	ALL
5.1.2	RF Setting non-volatility	1.0+	C	ALL
5.2.1	Multicast downlink	1.0+	C	ALL
6.1.1	Application ADR default	1.0+	ALL	ALL
6.2.1	ABP FCnt reset value	1.0+	ALL	ALL
6.2.2	FCnt 32-bit conformity	1.0+	ALL	ALL
6.2.3	Downlink message dismissal on wrong FCntDn	1.0+	ALL	ALL
7.1.1	Radio Signal Noise Ratio	1.0+	ALL	ALL
8.1.1	LinkCheckReq/Ans	1.0+	ALL	ALL
9.1.1	Test LinkAdrReq with value out of spec	1.0+	ALL	ALL
9.1.2	Incomplete MAC command	1.0+	ALL	ALL
9.1.3	Unknown MAC command ID	1.0+	ALL	ALL
9.1.4	Valid MAC command followed by invalid MAC command	1.0+	ALL	ALL
B.1.1	BeaconFreqReq/Ans	1.0.2+ regional rev B	B	ALL
B.1.2	PingSlotChannelReq/Ans	1.0.2+ regional rev B	B	ALL

ID	Description	LoRaWan Specs	Class	Region
B.1.3	DeviceTimeReq/Ans	1.0.2+ regional rev B	B	ALL
B.2.1	Switch from class A to class B	1.0.2+ regional rev B	B	ALL
B.2.2	Switch from class B to class A	1.0.2+ regional rev B	B	ALL
B.2.3	Beacon less test	1.0.2+ regional rev B	B	ALL
B.4.1	PingSlot downlink frame packet loss	1.0.2+ regional rev B	B	ALL

5.1 TPIT 1.0 – IEEE Addressing Compliance

Each LoRaWAN end device has a globally unique IEEE EUI-64 address, the DevEUI.

These addresses are allocated by manufacturers within address blocks that must be purchased from IEEE, and prefixed by 3 different address blocks:

- MA-S: MAC Address Block Small
- MA-M: MAC Address Block Medium
- MA-L: MAC Address Block Large (previously OUI)

Please check IEEE Standards Association (https://standards.ieee.org/) for more information.

IEEE Addressing compliance		Expected results / Observations
TPIT 1.0.1 **CORE**	Specify the range of DevEUI allocated for the device part number.	DUT DevEUI is in the purchased range.

5.2 TPIT 1.1 – Activation By Personalization

The Activation By Personalization (ABP) is the basic activation of a device on a LoRaWAN Network. A device must know 2 mandatory parameters such as its DevAddr, NwkSKey. Thanks to these 2 parameters, a device can communicate and be authenticated on the ThingPark Wireless platform.

Activation By Personalization		**Expected results / Observations**
TPIT 1.1.1 CORE	On ThingPark networks, the DevAddr must be formed as in **NwkID** [31..25] + **NwkAddr** [24..00] **Test:** DUT should have configurable DevAddr or a defined procedure for producing a defined range of DevAddrs with given NwkID prefix or DUT uses a random DevAddr with experimental netID prefix 0 or 1 Device manufacturers must ensure that each **DevAddr**=**NwkID**+**NwkAddr** corresponds to very few, ideally a single **DevEUI**. If the device partner is to roll out devices on the network of ThingPark-enabled operator platform, the **NwkID** will either: • be specific to that network. • be the Actility NwkID (0x02/7) • be the experimental NwkID (0x00/7 or 0x01/7)	Manufacturer understands the role of DevAddr and NwkID, and has a defined procedure to produce sensors for a given network.
TPIT 1.1.2 CORE	Device partners should use the 128 bit **NwkSKey** for the ThingPark Wireless network to verify the authenticity and integrity of each message. **Test:** NwkSKey must be random or configurable by user Comment: Critical to network operation to resolve DevAddr conflicts and to network security. Device partners are to use a different **NwkSKey** for each device, e.g. allocate it randomly. Eliminate all devices with fixed NwkSKey or random allocation over a subset that has high probability of NwkSKey collision.	NwkSKey can be set by user or is truly random over the whole 128-bit space. When NwkSKey is set, check that device manual explains that same key MUST NOT be used for 2 devices and if randomly allocated must use all 128-bit space.
TPIT 1.1.3 CORE	**Test:** Have DUT send uplink messages	The MIC checksum should be correct

5.3 TPIT 1.2 – Over-The-Air Activation

The Over-The-Air-Activation is a method allowing a device to register to a network through 3 parameters:

- its DevEUI which is its unique MAC address
- an AppEUI which is a unique application identifier
- an AppKey which encrypt the JoinAccept message

It assumes that the DevEUI is already tested in TPIT 1.0.1.

Over-The-Air Activation		Expected results / Observations
TPIT 1.2.1 CORE	Device partners are to use to generate a 128 bits **AppKey** per application.	

5.3.1 TPIT 1.2.2 – Join accept testing

The main purpose of the Join Accept testing is to verify that the JOIN_ACCEPT message is properly formed and the settings downloaded on the device are properly used in subsequent messages from the sensor.

Join Accept Testing		Expected results / Observations
TPIT 1.2.2.1 CORE	Sensor to send Join-request to Network Server with proper **AppEUI, DevEUI.** Network Server to send Join-accept with **NetID, DevAddr, AppNonce, DLSettings, CFList, RXDelay** Following Join_Accept Sensor to send confirmed/unconfirmed data message up to the Network Server with the right DevADDR	Actility and Vendor to test Join-accept MAC Message format Sensor vendor to test join_accept message delivery Actility to test AppSKey and NwkSKey generation Actility to test use of CFList Actility to test DevAddr allocation Actility to test NwkSKey and AppSKey calculation by device
TPIT 1.2.2.4 CORE	Actility to test if no response sent in **JOIN_ACCEPT_DELAY1** by not replying in less than 6s	Vendor to test re-join message
TPIT 1.2.2.6 CORE	ETSI Test of CF List Network Server to send a specific CF List to device FCC Not available	ETSI Test that the device communicates on the new channels with the CF List FCC Test that the CFlist is ignored by the end-device

Join Accept Testing		Expected results / Observations
TPIT 1.2.2.7 CORE	Actility test if DLSettings and RxDelay defined in JoinAccept are properly used by the device	Set RxDrOffset, Rx2Datarate and RxDelay with a different nominal value. Check a downlink through Rx1 and Rx2, the DUT must answer in both case

5.3.2 TPIT 1.2.3 – Aggregated Duty Cycle testing

Aggregated Duty Cycle Testing		Expected results / Observations
TPIT 1.2.3 **CORE**	Aggregated Duty Cycle during the first hour: Duty-cycle < 1% Aggregated Duty-cycle during the next 10 hours <0.1% Aggregated Duty-cycle after 11 hours (measured over a sliding 24h window) <0.01%	Device under test presents the risk of synchronized uplink message storm (yes/no)? DUT satisfies the back-off duty cycle conditions for retransmissions

5.4 TPIT 2.1 – Uplink Message compliance

This test verifies the compliance of confirmed and unconfirmed messages.

Uplink Message compliance		Expected results / Observations
TPIT 2.1.1 **CORE**	Device is sending **Unconfirmed** messages.	Verify correct message type unconfirmed.
TPIT 2.1.2 **CORE**	Device sends periodic messages or human can trigger a message. Device is sending **Confirmed** messages. Device configuration and retransmit (with same Frame Count Up) if an ACK is not received.	Verify correct message type confirmed Device partner to verify that confirmed messages are sent and received by LRC with incrementing **FCntDown** and ACK bit set. Check that DUT stops retransmissions as soon as ACK is received.
TPIT 2.1.3 **CORE**	All SF test The device sends at minimum 2 message on each SF (the range depends on the region profile)	Verify on the Wireless logger that at minimum 2 messages have been received on each SF
TPIT 2.1.4 **CORE**	Check the packet loss rate The device sends 60 uplink messages: (20 at SF7, 20 at SF8, 20 at SF9)	The packet loss rate must be maximum 1 packet lost by SF.

5.5 TPIT 2.2 – Downlink message compliance

This test verifies the downlink path from the ThingPark platform to the device.

Downlink Message compliance		Expected results / Observations
TPIT 2.2.1 CORE	Device sends periodic messages or human can trigger a message to open a RX window. Actility posts an **unconfirmed** message to the device with the proper **FCntDown**.	Verify on the device side that the message has been received.
TPIT 2.2.2 CORE	Device sends periodic messages or human can trigger a message to open a RX window. Actility posts a **confirmed** message to the device with the proper **FCntDown**.	Verify on the device side that the message has been received. Verify that the LRC has received the ACK from the device.
TPIT 2.2.3 CORE	All SF test The device sends at minimum 2 message on each SF (SF7, SF8, SF9, SF10, SF11, SF12) Actility posts a **confirmed** message to the device with the proper **FCntDown**.	Verify on the device side that the message has been received. Verify that the LRC has received the ACK from the device.
TPIT 2.2.4 CORE	Check the packet loss rate Device sends periodic messages or human can trigger a message to open a RX window. Actility posts 60 **confirmed** messages to the device with the proper **FCntDown**. (20 at SF7, 20 at SF8, 20 at SF9)	The packet loss rate must be maximum 1 packet lost by SF. Verify that the LRC has received the ACK from the device.
TPIT 2.2.5 CORE	Test: Send 60 downlink command on RX2 window to DUT (any command that has a result that can be checked).	Command should be received by DUT. That verifies proper function of Rx2. The packet loss rate must be at most 3 packets lost for 60 downlinks (5%).
TPIT 2.2.6 CORE	Downlink Window Timing The test server sends downlink packets at both -20 and +20 microsecond offsets from the nominal receive window location and verifies that the packets are properly received by End-Device.	Verify that the LRC received **DevStatusAns** on both windows and timing

5.6 TPIT 3.1 – Default Uplink RF channel usage

Default Uplink RF channel usage		Expected results
TPIT 3.1.1 **CORE**	Test: ETSI DUT uses only default channels at boot. This channels depends on the Regional Parameters for the DUT. FCC Not applicable	ETSI Detect DUTs with hardcoded channels, which will have high PER in case of channel mismatch and may retransmit packets more than needed. FCC Not applicable
TPIT 3.1.2 **CORE**	Test: DUT should alternate between channels Comment: DUTs should not concentrate on a single channel	Check that uplinks cycle through available frequencies allocating approximately the same number of uplinks in each channel.

5.7 TPIT 3.2 – Adaptive Data Rate

ADR testing can be performed in any ISM Band supported by the LRR.

The device must support ***LinkADRReq*** and ***LinkADRAns*** MAC commands to test the Actility Adaptive mechanism

Adaptive Data Rate		Expected results
TPIT 3.2.1 **CORE**	SF Fallback uplink unconfirmed message scenario Precondition: • Switch on device and place the device under coverage ideally with sufficient path loss to reach SF7 or force DUT to use SF7 Test scenario: • The device is sending uplink unconfirmed messages • Block the downlink (MAC commands) from the LRC (ex: Actility can access the LRC to block the downlink)	Use Wireless Logger to observe ADRAckReq flag and SF switch to higher rates SF. After ~64 uplink the DUT must enable the ADRAckReq flag. Then after ~32 uplink with ADRAckReq flag enabled, the DUT must increase the SF Use Wireless Logger to observe: • SF of DUT uplink messages should progressively use increasing SF, until SF12 • Use of alternating channels • No aggressive repeat, respect of duty cycle limits
TPIT 3.2.2 **CORE**	SF Fallback uplink confirmed message scenario Precondition: • Switch on device and place the device under coverage ideally with sufficient path loss to reach SF7 or force DUT to use SF7 Test: • The device is sending uplink confirmed messages • Block the ACK from the LRC (ex: Actility can access the LRC to block the downlink) Comment: DUT must be capable of lowering SF when downlink communication is lost.	Use Wireless Logger to observe: • SF of DUT uplink messages should progressively use increasing SF, until SF12 • Use of alternating channels • No aggressive repeat, respect of duty cycle limits
TPIT 3.2.3 **RF opt.**	Prerequisite: Put device in good RF conditions (SNR >0, but SNR < 4 to avoid SX1301 non-linear RSSI/SNR region) Test • Send ADR command with TxPower parameter set to 4 (5dBm)	Check in WLogger that uplink RSSI decreases by about 9dB, assuming default is 14dBm
TPIT 3.2.4	Test • Send ADR command with Redundancy parameter set to 3 • Try to post a downlink to check if the device stops the repetition	Check that devices effectively retransmits uplinks 3 times, with same uplink counter and same packet. A new downlink must stop the uplink repetition, frame counter must be increased.

5.8 TPIT 3.3 – Deactivated ADR (mobility)

Deactivated ADR tests are key for all mobile devices for which the ADR mechanism is not applicable.

Deactivated ADR		Expected results
TPIT 3.3.1 **RF-opt.**	Device should be configured to send periodic uplink or human is able to trigger a message. Device should be configured to start at a fixed SF with no ADR bit set (ADR not supported)	Observe on wireless logger no SF change when device is placed next to the LRR

5.9 TPIT 4.1 – MAC Command RXTimingSetup

The ***RXTimingSetupReq*** command allows configuring the delay between the end of the TX uplink and the opening of the first reception slot. The second reception slot opens one second after the first reception slot.

MAC Command RXTimingSetup		Expected results
TPIT 4.1.1 **CORE**	Precondition: Put DUT in good RF conditions SNR >0 Test: LRC sends **RXTimingSetupReq** with non-default timing (e.g. 3 seconds) Wait for **RXTimingSetupAns** Send any command with an effect that can be checked	Command action should be executed. Checks that DUT can use non-default Rx window.
TPIT 4.1.1.1 **CORE**	**LoRaWAN 1.0.1 update:** Check that ***RXTimingSetupAns*** is repeated until first downlink Send command on DUT: check action (tests RX1 offset) Move DUT to edge of cell condition (LRC should use Rx2), or force Rx2 usage. Send command to DUT (Tests RX2 new data rate).	

5.10 TPIT 4.2 – MAC Command RXParamSetup

The ***RXParamSetupReq*** command allows a change to the frequency and the data rate set for the 2 windows (RX1 and RX2) following each uplink.

MAC Command RXParamSetup		Expected results
TPIT 4.2.1 **CORE**	Precondition: Put DUT in good RF conditions SNR >0 Test: LRC sends ***RXParamSetupReq*** with **RX1DRoffset** value set to 1, **RX2Datarate** value set to SF10 and Rx2 **Frequency** value set in frequency plan ISM Wait for **RXParamSetupAns**	Device partner to verify that the RX1 offset, RX2 data rate and RX2 frequency have been properly set. DUT should execute downlink commands on Rx1 and Rx2
TPIT 4.2.1.1 **CORE**	**LoRaWAN 1.0.1 specifics:** Check that ***RXParamSetupAns*** is repeated until first downlink Send command on DUT: check action (tests RX1 offset) Move DUT to edge of cell condition (LRC should use Rx2), or force Rx2 usage. Send command to DUT (Tests RX2 new data rate).	

5.11 TPIT 4.3 – MAC Command DevStatus

With the ***DevStatusReq*** command the LRC requests status information from the device end-device and expects a ***DevStatusAns***. The command has no payload.

MAC Command DevStatus		Expected results
TPIT 4.3.1	LRC to send a ***DevStatusReq*** to the device	LRC to obtain: • battery level information • signal-to-noise margin

5.12 TPIT 4.4 – Creation / Modification of a Channel

The ***NewChannelReq*** command can be used to either modify the parameters of an existing channel or to create a new one. The command sets the center frequency of the new channel and the range of data rates usable on this channel.

Note: *The* ***NewChannelReq*** *command is not available in US FCC regulation. All channels should be implemented then the* ***ChMaskCntl*** *and the* ***ChMask*** *in* ***LinkADRReq*** *command enable/disable the rights channels.*

The ***DlChannelReq*** command can be used to modify the downlink frequency of the RX1 slot for a given uplink channel, thus creating an asymmetric Uplink/Downlink scheme.

MAC Command NewChannel		Expected results
TPIT 4.4.1 CORE	Add new channels Device should have the 2/3 default logical channels configured. Device should send uplink messages on all 3 channels. The LRC will send a ***NewChannelReq*** with any of new frequency in the ISM range (optional channels configured on LRR based on local regulation)	Wireless logger should show new logical channels being used for uplink messages.
TPIT 4.4.2 CORE	Remove channels Device should have more than 3 channels enabled Device should send uplink messages on all channels. The LRC will send a ***NewChannelReq*** with any of new frequency set to 0 MHz	Wireless logger should not show new logical channels being used for uplink messages.
TPIT 4.4.3 CORE	Update downlink channels Device should have all the channels configured using the ***NewChannelReq*** MAC command. Device should send uplink messages, the LRC will then send ***DlChannelReq*** with any new frequency in the ISM range for the downlink, allowing for symmetric/asymmetric UL/DL. LRC will then send downlink on the new downlink channels.	Wireless logger should show ***DlChannelReq*** MAC commands with the device acknowledging them. Downlinks sent by LRC should be on new downlink frequencies.
TPIT 4.4.4 CORE	Remove default channels The device must reject a ***NewChannelReq*** on default channels	Wireless logger should show ***NewChannelAns*** MAC commands with a rejected frequency

5.13 TPIT 4.5 – MAC command over port 0

If **FOptsLen** is greater than 15 bytes, the mac command is send over the port 0:

MAC command over port 0	Expected results

TPIT 4.5.1 RF-opt.	Device should be configured to send periodic uplink or human is able to trigger a message. The LRC will send a command MAC with more 15 bytes	Wireless logger should show new commands received. Device partner checks the device has received the command.

5.14 TPIT 4.6 - MAC TxParamSetup test

The ***TxParamSetup*** command allows to change a device's maximum emission level and dwell time handling. This is a requirement in some specific regions.

Note: *The* ***TxParamSetup*** *is only valid for LoRaWAN 1.0.2 devices, on AS923*

TxParamSetup		**Expected results**
TPIT 4.6.1 CORE	Uplink Dwell time test: 1. LRC sends TxParamSetup command with uplink dwell time set to 400ms 2. LRC sends LinkADRReq with the highest SF available in this region (should be SF12) 3. DUT sends uplinks at a SF less than SF12 4. LRC sends TxParamSetup command with uplink dwell time set to unlimited 5. Device sends uplinks at the requested Spreading Factor	When the Uplink Dwell time bit is set, Spreading Factors above SF10 cannot be used since they generate payloads longer than 400ms just with the minimum LoRaWAN requirements. The DUT should thus never use higher than SF10 value.
TPIT 4.6.2 CORE	Downlink Dwell time test: 1. LRC sends TxParamSetup command with downlink dwell time set to 400ms and uplink dwell time disabled 2. LRC sends LinkADRReq with the highest SF available in this region (should be SF12) 3. DUT sends uplinks at SF12 4. LRC sends short downlinks at SF10 5. LRC sends TxParamSetup command with downlink and uplink dwell time set to unlimited 6. Device sends uplinks at SF12 7. LRC sends short downlinks at SF12	Packets at step 4 should be received properly by device. Packets at step 7 should be received properly by device.

5.15 TPIT 5.1 – Class C receive window

The main test in Class C is to test receive window settings by sending downlink messages randomly.

Class C receive window settings		Expected results
TPIT 5.1.1 CORE	Provision the device in the ThingPark platform as a Class C device. Sensor to send unconfirmed data up message to network server. Actility to send downlink messages randomly after first uplink message	Actility to check delivered message. Sensor vendor to provide a way of displaying properly delivery of downlink messages
TPIT 5.1.2	For all devices that are connected to the mains power, more generally any type of device which presents the possibility of massive synchronized restart, the restart MUST preserve all RF settings (RF channels, frame counters), and therefore not trigger any uplink message storm.	After a device restart, the device should use all channels previous configured

5.16 TPIT 5.2 – Class C Multicast for downlink messages

Class C Multicast for downlink messages		Expected results
TPIT 5.2.1 RF-opt.	Sensor vendor to provide 2 Class C devices. Actility to provide 2 DevAddr for the 2 devices a Class C Multicast DevAddr group which both Class C devices should listen to. Actility to send one downlink message to the DevAddr multicast address.	device is of the type above (yes/no) Actility and Vendor to check proper delivery of downlink message.

5.17 TPIT 6 – Application behavior

The application behavior contains all the specific settings dedicated to the device, such as the default SF at startup, the frame counter management.

5.17.1 TPIT 6.1 – ADR behavior

ADR behavior		Expected results
TPIT 6.1.1	The device can have a specific behavior on its SF at start-up. Test: • Start up the device from reset	Use Wireless Logger to observe the SF of the first message

5.17.2 TPIT 6.2 – FCnt management

Fcnt management		Expected results
TPIT 6.2.1	Until the reset command is implemented: APB mode: The device must start with **FCntUp/FCntDn <= 1** Test: ABP mode: The device can store the frame counter to avoid "replay attack", but the stack must come back in default configuration • Start up the device from reset	The FCntUp of the first message must be 0 or 1 for ABP. Or: The FCntUp of the first message may be the same as before restart if the device saved all the LoRaWAN.
TPIT 6.2.2	Frame counter **FCntUp/FCntDn <= 1** and should be increase over 32-bit counter. Test: • DUT uplink messages should send more than 65536 frames	The device in-memory frame counter should be a 32-bit counter. The MIC computation should reflect in-memory counter values higher than 65536, which do not wrap to 1 when more than 65536 frames have been sent. Supplier declares that frame count has been implemented in 32bit mode in both uplink and downlink directions.
TPIT 6.2.3	For the security DUT must reject frame with a bad frame counter Test: • Send a message from LRC with a frame count down equal or inferior that the current frame count down	The DUT must be reject this frame

5.18 TPIT 7 – Radio Frequency testing

RF		Expected results
TPIT 7.1.1	The radio should have a good uplink/downlink SNR. Test: • Compare 60 uplink SNR with the reference device • Send 60 ***DevStatusReq*** from LRC to get the downlink SNR and compare with the reference device • The gateway and the DUT must be separated by 10m	The average SNR uplink/downlink must be greater than SNR(ref)-5 dBm

5.19 TPIT 8 – MAC Command LinkCheck

LinkCheck		Expected results
TPIT 8.1.1 **Optional**	This MAC command is initiated by end-device and cannot be triggered by the Network Server.	Wireless logger should show that DUT send ***LinkCheckReq*** MAC command itself and the Network Server answer by ***LinkCheckAns***

5.20 TPIT 9 – MAC Commands incorrect

MAC Commands incorrect		Expected results

TPIT 9.1.1	Bad MAC command will be sent to the device
	• Test ***LinkAdrReq*** with value out of spec
	  ○ Datarate 8
	  ○ ChMask= 0x0000
		Wireless logger should show the bad MAC command, the device must ignore or reject the MAC command

TPIT 9.1.2	Bad MAC command will be sent to the device • Incomplete MAC command ○ FPort 0, payload: 0x030100	Wireless logger should show the bad MAC command, the device must ignore or reject the MAC command
TPIT 9.1.3	Bad MAC command will be sent to the device • Unknown MAC command ID ○ FPort 0, payload: 0x7F	Wireless logger should show the bad MAC command, the device must ignore or reject the MAC command
TPIT 9.1.4	Bad MAC command will be sent to the device • Valid MAC command followed by invalid MAC command	Wireless logger should show the bad MAC command, the device must ignore or reject the MAC command

5.21 TPIT B – Class B Tests

This section concerned class B test requirements for interoperability.

To be performed, class B must be setup.

- Setup device
  - Device profile 1.0.2 regionals parameters revision B or 1.0.3 with class B enabled
  - Connectivity plan with class B enabled
  - RFRegion with Beacon and PingSlotChannel enabled
- Setup gateway
  - Good GPS coverage
  - Class B enabled

5.21.1 Mac commands

Class B MAC commands		Expected results
TPIT B.1.1 CORE	***BeaconFreqReq/Ans*** • Modify the frequency on which this end-device expects the beacon	Actility sent a ***BeaconFreqReq*** with the requested frequency. The device must answer with a ***BeaconFreqAns*** and listen on the new frequency
TPIT B.1.2 CORE	***PingSlotChannelReq/Ans*** • modify the frequency on which this end-device expects the beacon (Freq only)	Actility sent a ***PingSlotChannelReq*** with the requested frequency and data rate The device must answer with a ***PingSlotChannelAns*** and listen on the new frequency and data rate
TPIT B.1.3 Optional	***DeviceTimeReq/Ans*** • Device able to send ***DeviceTimeReq*** mac command	Use Wireless Logger to observe the ***DeviceTimeReq*** sent by the device, the network server must answer with a ***DeviceTimeAns***

5.21.2 Switch between class A / B

Switch between class A / B		Expected results
TPIT B.2.1 CORE	Switch from class A to class B After received the Beacon the DUT must sent a ***PingSlotInfoReq***	Use Wireless Logger to observe the ***PingSlotInfoReq*** sent by the DUT Then after received the ***PingSlotInfoAns***, the class B bit of the FCTRL field must be set for every uplink frame transmitted
TPIT B.2.2 CORE	Switch from class B to class A Without beacon for a period of two hours, the DUT must revert to class A	The class B bit of the FCTRL field must be unset for every uplink frame transmittedx
TPIT B.2.3 CORE	***Beacon*** less test The DUT must successfully receive the downlink frame through ping slot without ***beacon*** for a period of two hours	Use Wireless Logger to observe the right reception of each downlink frame that must be answered by the device

5.21.3 Packets loss

Packets loss		Expected results
TPIT B.4.1 CORE	PingSlot downlink frame packet loss: Actility posts 60 ***confirmed*** messages to the DUT	Use Wireless Logger to observe the right reception of each downlink frame that must be answered by the device The packet loss rate must be 5% maximum