Z-Wave Long Range MAC Layer Test Specification

Release 2.9.0

Z-Wave Alliance

May 30, 2025

Table of contents

1 Preamble

1.1 Description

Test specification for testing the MAC layer of the Z-Wave Long Range protocol

Reviewed by the Z-Wave Alliance Core Stack Working Group (CSWG) and approved by the Z-Wave Alliance Board of Directors.

1.2 Disclaimer

THIS SPECIFICATION IS BEING OFFERED WITHOUT ANY WARRANTY WHATSOEVER, AND IN PARTICULAR, ANY WARRANTY OF NON-INFRINGEMENT IS EXPRESSLY DISCLAIMED. ANY USE OF THIS SPECIFICATION SHALL BE MADE ENTIRELY AT THE IMPLEMENTER'S OWN RISK, AND NEITHER THE ALLIANCE, NOR ANY OF ITS MEMBERS OR SUBMITTERS, SHALL HAVE ANY LIABILITY WHATSOEVER TO ANY IMPLEMENTER OR THIRD PARTY FOR ANY DAMAGES OF ANY NATURE WHATSOEVER, DIRECTLY OR INDIRECTLY, ARISING FROM THE USE OF THIS SPECIFICATION.

1.3 Revision Record

Table 1.1: Revision History

Doc. Rev	Date	By	Pages Affected	Brief Description of Changes
0.1	2020/09/29	CSWG	Initial Version	
0.5	2021/02/12	CSWG	ALL	Complete set of test cases added to document
0.8	2021/03/01	CSWG	None	Updated to revision 0.8 after WG review. Ready for TC review
0.9	2021/03/29	CSWG	Frontpage	Cleanup for IPR review.
1.0	2021/08/27	ZWA Board		Approved for Publication
1.1	2022/06/28	CSWG	Multiple	3.1 (disclaimer on negative testing). Inclusion of 4 Test Cases from the PHY Test Specification 3.2, 3.3, 3.4, thus adding +3 in TC number from there on with respect to version 1.0. Removed TC 3.11, thus shifting upwards the following Test Cases. Various typos corrections & clarifications: Test Cases: 3.14, 3.15, 3.17, 3.18, 3.19, 3.20, 3.26, 3.27. New test Cases From 3.28 to 3.34
1.5	2023/04/14	CSWG	Multiple	3.6 – Overhaul of how CCA is performed Various typos, corrections & Clarifications: Test Cases: 3.8, 3.10, 3.11, 3.12, 3.13, 3.14, 3.15, 3.16, 3.17, 3.18, 3.19, 3.20, 3.21, 3.22, 3.23, .24, 3.25, 3.27, 3.28, 3.29, 3.30, 3.31, 3.33, 3.34
1.5.1	2023/11/28	CSWG	3.2.3	Removed Explorer frame
			3.3.3	Removed Explorer frame
			3.6.4 3.6.5	Reduces scope of Pass and Fail criteria's
			3.14.1	Fixed test prerequisites
			3.16.1	Added frame generator to prerequisites
			3.18.2	Clarified test setup
			3.19.1	Fixed test prerequisites
			3.20	Removed spectrum analyzer from test setup and fixed pass criteria's
			3.23	Updated test to not include invalid RSSI values
			3.24.1	Fixed test prerequisites
1.5.2	2024/01/15	CSWG	All	Corrected erroneous references and footers
			3.4	Fixed description, pass and fail criteria for Rx-To-Tx turnaround time
			3.6	Fixed Prerequisites and Test setup in network robustness
1.5.3	2024/01/23	CSWG	3.6.2	Fixed figure 5 to align with test setup description
2.5.0	2024/10/16	CSWG	all	Document is migrated from Word to Github
2.5.1	2024/12/20	CSWG	3.2 3.3. 3.4	No use to check all LRF profile if the behavior is the same.
			3.2	No use to check preamble on both controller and end device.
2.7.0	2025/03/21	CSWG	n/a	Ready for the TC review.
2.9.0	2025/05/30	TC	n/a	Approved for IPR review.

1.4 Abbreviations

Table 1.2: Abbreviations

Abbreviation	Explanation
ACK	Acknowledgement
AL	Always Listening
API	Application Program Interface
ERTT	Encoder Receiver Transmitter Test
FCS	Frame Check Sequence
FL	Frequently Listening
LR	Long Range
MAC	Media Access Control
MFR	MAC Footer
MHR	MAC Header
MPDU	MAC Protocol Data Unit
MSDU	MAC Service Data Unit
NL	Not Listening
NOP	No Operation
PHY	Physical (Layer)
RF	Radio Frequency
RSSI	Received Signal Strength Indicator
S2	Security 2 (Command Class)
TX	Transmission
US_LR	United States Long Range
VSA	Vector Signal Analysis (software)

2 INTRODUCTION

2.1 Purpose

The purpose of this specification is to provide a set of tests that verifies compliance with the MAC layer of the Z-Wave Long Range protocol.

2.2 Audience and Prerequisites

Developers and testers of the Z-Wave Long Range protocol.

An RF Sniffer hardware and analyzer software that can be tuned in on the valid Long Range frequency for Z-Wave or a Sniffer module and PC Application. Z-Wave Controller or equivalent to execute communication between the different nodes.

3 MAC-LAYER TEST CASE DESCRIPTIONS

3.1 General assumptions

For performing this test it is assumed that the MAC layer is running on a PHY layer that is compliant with the "Z-Wave Long Range PHY Layer specification" [1] and is verified by the set of tests in the "Z-Wave Long Range PHY Layer Test Specification."

All components are defined in "Z-Wave Long Range PHY and MAC Specification" and that document is the sole reference for the present Test Plan.

All **times** and **time-out periods** must be compliant with the values described in tables 6-32 & 6-33 from "Z-Wave Long Range PHY and MAC Specification." [1]

Inclusion in Z-Wave Long Range refers to Bootstrapping in "Z-Wave Long Range PHY and MAC Specification" – 6.1.2, each device has the same HomeID and different NodeID. The controller needs to have each End Node registered in its Node List with a unique NodeID. It is performed by Smart Start inclusion for Long Range Devices.

From here on all requirement numbers refer to sections in "Z-Wave Long Range PHY and MAC Specification"[1]. It is also referred to as [LRMAC].

Test Cases towards the end of the spec are the Negative Testing complement to Test Cases described earlier, they show the number and title of the Test Case they relate to for identification. These Negative Testing Test Cases, at the current time of issuing of this spec, are not mandatory.

3.2 Preamble field, LR

Data frames transmitted by a Z-Wave device must be formatted as described in [LRMAC] section 5.3.1: With a preamble field, a Start of Frame delimiter, payload and an End of Frame delimiter. The requirements for the number of preamble bytes to transmit are stated in [LRMAC] table 5-10.

The preambles are coded according to [LRMAC] tables 5-2, 5-4, 5-5 and 5-6.

The number of preamble types transmitted must be tested for each type of Z-Wave frame according to [LRMAC] table 5-10 for at least one LRF Profile from [LRMAC] table 5-1 in one LR region.

3.2.1 Prerequisites

1. All Z-Wave devices are capable of transmitting Z-Wave packages correctly formatted according to [LRMAC] section 5.
2. The Z-Wave devices must be able to transmit each of the various types of Z-Wave frames described in [LRMAC] table 5-10.
3. The Z-Wave devices must be mounted on a PCB enabling a cabled RF connection between a RF measurement device and a 50 Ohms matched output of the Z-Wave device.
4. A spectrum analyzer with better or identical specifications to a Rhode & Schwartz FSV3007, 7.5GHz
5. A digital VSA installed on the spectrum analyzer with the capabilities of at least Rhode & Schwartz option FSV3-K70.

The spectrum analyzer should be initialized to:

Table 3.1: Preamble Spectrum Analyzer settings

Spectrum analyzer parameter	Setting
f_{center}	$f_{center\ frequency}$ according to [LRMAC] table 5-1
Span	1MHz
Resolution Bandwidth	1kHz
Video Bandwidth	Auto
Amplitude reference level	30dBm
Detector type	Average
Analog demodulation type	OQFSK
Demodulation time (time across screen)	t_{demod_time}
IF filter	400 ks/s
Trigger option	IF power

6. The number of preambles transmitted for each type of transmission-type must be measured using the demodulation feature of the VSA FSV3-K70 option of the spectrum analyzer. Below is shown an example of how a demodulated data stream could appear on the screen of the spectrum analyzer:

Example of Demodulation
view of FSV3-K70 Option

Row/ Line	1	2	3	4	5	6	7	8
1	0	2	1	3	3	0	2	1
2	1	0	1	3	0	0	1	3
3	2	0	1	1	3	1	3	2
4	3	3	0	1	3	2	1	0
5	2	2	3	0	1	0	2	1
6	1	2	3	2	1	0	3	2

Figure 3.1: Example of demodulated symbols from OQPSK data stream

7. Devices all flashed to work in LR frequency with rates as per table 5-10:
   - 1 x Serial API Controller
   - 1 x Always Listening End Node
   - 1 x Frequently Listening End Node

3.2.2 Test Setup

1. Include all End Nodes to the Controller's Network
2. Connect the Controller's Antenna to the Spectrum Analyzer with a coaxial cable and turn off the AL & FL End nodes.
3. From the PC Controller application, try sending frames to the End Nodes with Data Payload (MSDU) = 0x00 (NOP), as follows:
   a. Sending a Broadcast sending the Controller's NIF.
   b. Sending a Singlecast frame to one AL End Node.
   c. Sending a Singlecast frame to the disconnected AL End Node.
   d. A Singlecast to the FL End Node.

3.2.3 Test Result

1. All End Nodes are included to the Controller's Network.
2. The antenna of the Controller device is connected to the Spectrum Analyzer and the AL & FL End Nodes are turned off.
3. Each frame type is captured correctly in the Spectrum Analyzer:
   a. The Broadcast with the Controller's NIF.
   b. The singlecast to an AL End Node.
   c. The singlecast frame to the FL End Node
   d. The Wake Up Beam frames when the Controller can't reach the FL End Node.

The measurement result is an analysis of the preamble pattern for each type of Z-Wave frame type transmitted at each data rate as stated in [LRMAC] table 5-10. If any irregularities are found within the time period $t_{preamble}$, the Z-Wave device has failed the test.

3.2.4 Pass Criteria

The Z-Wave device shall pass the test if:

1. The demodulated data pattern must match with expected pre-amble pattern as described in ([LRMAC] Table 5-10).

3.2.5 Fail Criteria

The Z-Wave device shall fail the test if:

1. The demodulated data pattern does not match with expected pre-amble pattern as described ([LRMAC] Table 5-10)

3.3 Start of Frame field

The transceiver of a Z-Wave must be able to correctly transmit and correctly receive Z-Wave start of frame information as described in [LRMAC] section 5.3.3. The data content of the Start of Frame field is described in [LRMAC] table 5-11. The handling of Start of Frame field in Z-Wave frames must be tested for all data rates, i.e. at least one LRF Profile from [LRMAC] table 5-1 in one LR region.

3.3.1 Prerequisites

1. A Z-Wave device capable of transmitting and receiving, decoding and error handling Z-Wave frames formatted according to [LRMAC] section 5.3.1. The Z-Wave device must be able to decode and data process at transmissions rates stated in [LRMAC] table 5-2. The Z-Wave device must be able to indicate when a frame is not correctly received, and all incoming Z-Wave frames must be acknowledged. The Z-Wave receiver device is here after called DUT.
2. The Z-Wave device must be mounted on a PCB enabling a cabled RF connection between a RF measurement device and a 50 Ohms matched output of the Z-Wave device.
3. A 700s Z-Wave Controller device which can transmit and receive Z-Wave messages. Data must be transmitted according to [LRMAC] tables 5-2 to 5-6 and formatted as described in [LRMAC] section 5.1.3. The test pattern generator must acknowledge all incoming Z-Wave traffic. The Z-Wave transmitter is here after called test pattern generator.
4. A means to control the transmission of a Z-Wave frame from the pattern generator.
5. A suggestion of devices to use:
   - 1 x Serial API Controller
   - 2 x Always Listening End Node, at least one with the ability to activate associations or Notifications
   - 1 x Frequently Listening End Node

3.3.2 Test Setup

The first possible method is to use the VSA option of a spectrum analyzer, demodulate the data stream and observe the SOF byte right after the preamble bits.

The Z-Wave receive device, the DUT, is connected to the Z-Wave pattern generator with a coax cable. The pattern generator transmits Z-Wave test packages to the DUT and the DUT must acknowledge the incoming Z-Wave frame. The number of correctly received packages and wrongly received packages must be recorded for both the Z-Wave pattern generator and the Z-Wave DUT and for both.

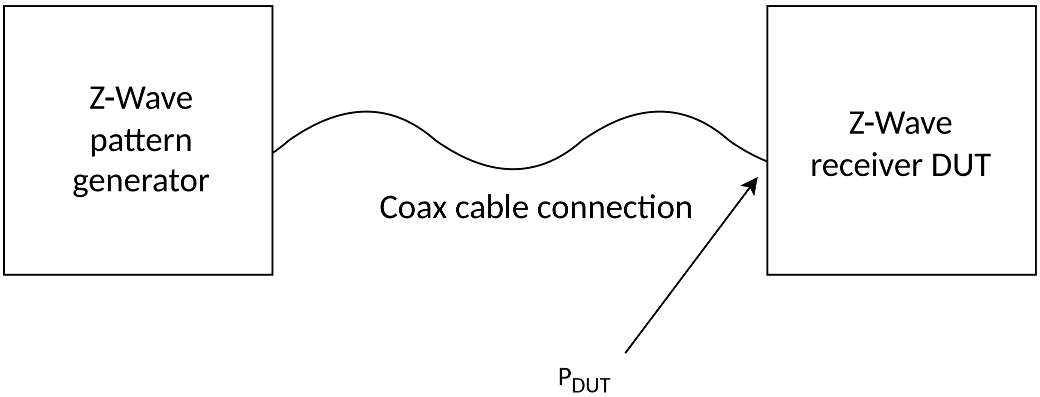

Figure 3.2: Start of frame measurement setup

Attenuation must be added to ensure that the received power at the DUT, P_{DUT}, is between -60 dBm and -50 dBm.

To achieve that:

1. Include all End Nodes to the Controller's Network
2. Ensure Associations or Notifications can be triggered.
3. Connect the DUT's Antenna to the Spectrum Analyzer with a coaxial cable and turn off the AL & FL End nodes.
4. From the PC Controller application, try sending frames to the End Nodes with Data Payload (MSDU) = 0x00 (NOP), as follows:
   a. Sending a Broadcast sending the Controller's NIF.
   b. Sending a Singlecast frame to the AL End Node.
   c. Sending a Singlecast frame to one turned off AL End Node.
   d. A Singlecast to the FL End Node.
5. Change the connection of the Coaxial cable to the other AL End Nodes with associations or notifications enabled. Turn off the Controller and the other AL & FL End nodes.
6. Send the NIF of the End Node and enable the Association or Notification and observe the frames captures by the Spectrum Analyzer.

3.3.3 Test Result

1. All End Nodes are included to the Controller's Network.
2. Associations/Notifications work as expected.
3. The antenna of the Controller device is connected to the Spectrum Analyzer and the AL & FL End Nodes are turned off.
4. Each frame type is captured correctly in the Spectrum Analyzer:
   a. The Broadcast with the Controller's NIF.
   b. The singlecast to an AL End Node.
   c. The singlecast frame to the FL End Node
   d. The Wake Up Beam frames when the Controller can't reach the FL End Node.
5. The Coaxial Cable is now on the other AL End Node. The Controller and the other AL & FL End nodes are turned off.
6. Verifications in 4.a – 4.c apply now for the End Node after the association/notifications are triggered.

For each of the LRF profile, at least 1000 frames must be transmitted by the test pattern generator and received by the DUT.

The measurement result is the number of correctly and wrongly received Z-Wave frames for each LRF profile by both the Z-Wave pattern generator and the DUT.

3.3.4 Pass Criteria

The Z-Wave device shall pass the test if:

1. For each LRF profile in [LRMAC] table 5-1, the frame error rate (FER) is < 0.002 for both the DUT and for the Z-Wave pattern generator; the Frame Error Rate can be calculated:

   FER = 1 – (Number of correctly received frames) / (Number of frames transmitted)

3.3.5 Fail Criteria

The Z-Wave device shall fail the test if:

1. Any LRF profile given in [LRMAC] table 5-1, the frame error rate (FER) is > 0.002 for either the Z-Wave DUT or the Z-Wave pattern generator:

   FER = (Number of frames with errors received) / (Number of frames transmitted)

3.4 RX-to-TX Turnaround time

The response time of the transceiver of a Z-Wave device should not be too fast, giving the transmitting device time to switch back from Tx to Rx, the so-called RX-to-TX turnaround time. The RX-to-TX turnaround time must be measured under the test conditions given in [LRMAC] section 5.2.5.9. The RX-to-TX turnaround time measurements must be tested for all data rates, i.e. at least one LRF Profile from [LRMAC] table 5-1 in one LR region.

3.4.1 Prerequisites

1. A Z-Wave device capable of transmitting and receiving, decoding and error handling Z-Wave frames formatted according to [LRMAC] section 5.3.1. The Z-Wave device must be able to decode and data process at transmissions rates stated in [LRMAC] table 5-2. The Z-Wave device must be able to indicate when a frame is not correctly received, and all incoming Z-Wave frames must be acknowledged. The Z-Wave receiver device is here after called DUT.
2. The Z-Wave device must be mounted on a PCB enabling a cabled RF connection between a RF measurement device and a 50 Ohms matched output of the Z-Wave device.
3. A 700s Z-Wave device which can transmit and receive Z-Wave coded data messages. Data must be transmitted according to [LRMAC] tables 5-2 to 5-6 and formatted as described in [LRMAC] section 5.3.1. The test pattern generator must acknowledge all incoming Z-Wave traffic. The Z-Wave transmitter is here after called test pattern generator. The output power of the Z-Wave pattern generator must be 20dB below the output power of the DUT.
4. A means to control the transmission of a Z-Wave frame from the pattern generator.
5. A spectrum analyzer with better or identical specifications to a Keysight CXA N9000A, 7.5GHz, with the capability to measure zero-span.
6. A 3 port RF resistive power combiner.
7. Recommended setup:
   - 1 x Serial API Controller as the frame generator
   - 1 x Always Listening End Node (AL)

3.4.2 Test setup

The DUT, RF generator and spectrum analyzer are all connected through the 3 port RF power combiner:

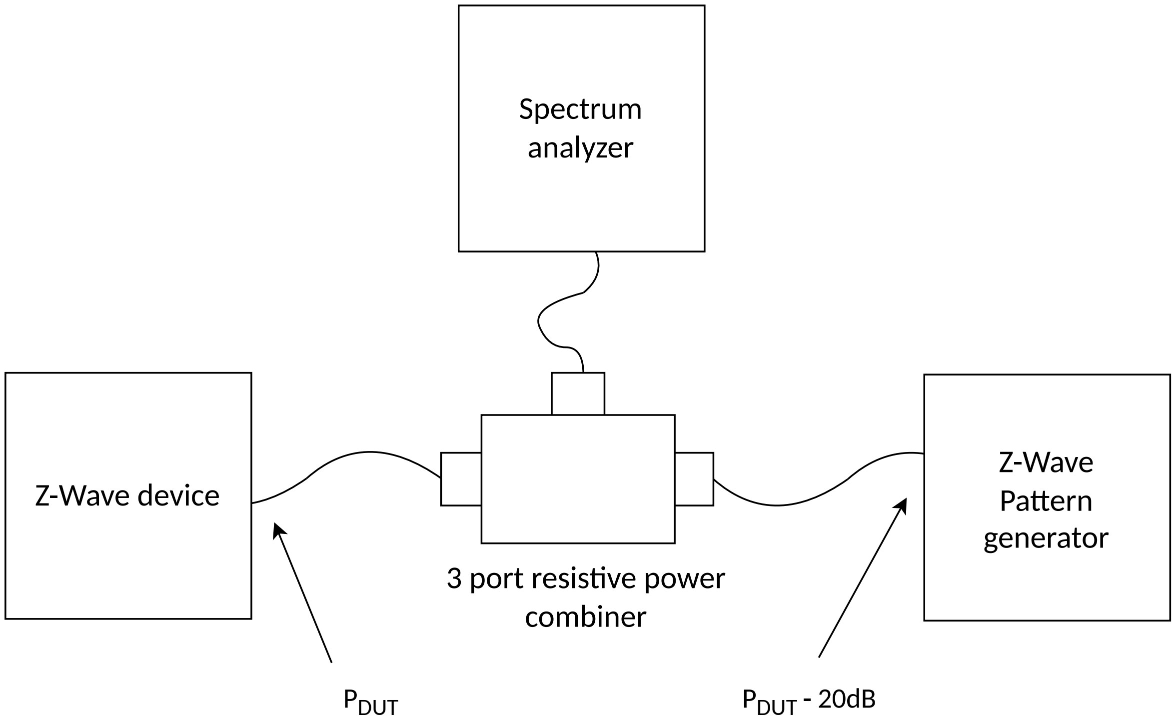

Figure 3.3: RX-to-TX turnaround time measurement setup

The spectrum analyzer must be initialized to:

Table 3.2: RX-to-TX turnaround time Spectrum Analyzer Settings

Spectrum analyzer parameter	**Setting**
f_{center}	$f_{center\ frequency}$ according to [LRMAC] table 5-1
Span	Zero span
Resolution Bandwidth	0
Video Bandwidth	Auto
Amplitude reference level	Depending on $P_{nominal}$
Detector type	Clear/write
Trigger	RF burst
RF trigger level	P_{DUT} – 10 and accounting for the loss of the 3 port resistive power combiner.
Sweep Time	10ms second

The Z-Wave pattern generator must be initialized to transmit Z-Wave data packets. A received Z-Wave packet at the DUT will prompt the DUT to transmit a acknowledge packet. Since the trigger threshold of the spectrum analyzer is set to trigger when the DUT transmits, the following can be observed on the spectrum analyzer:

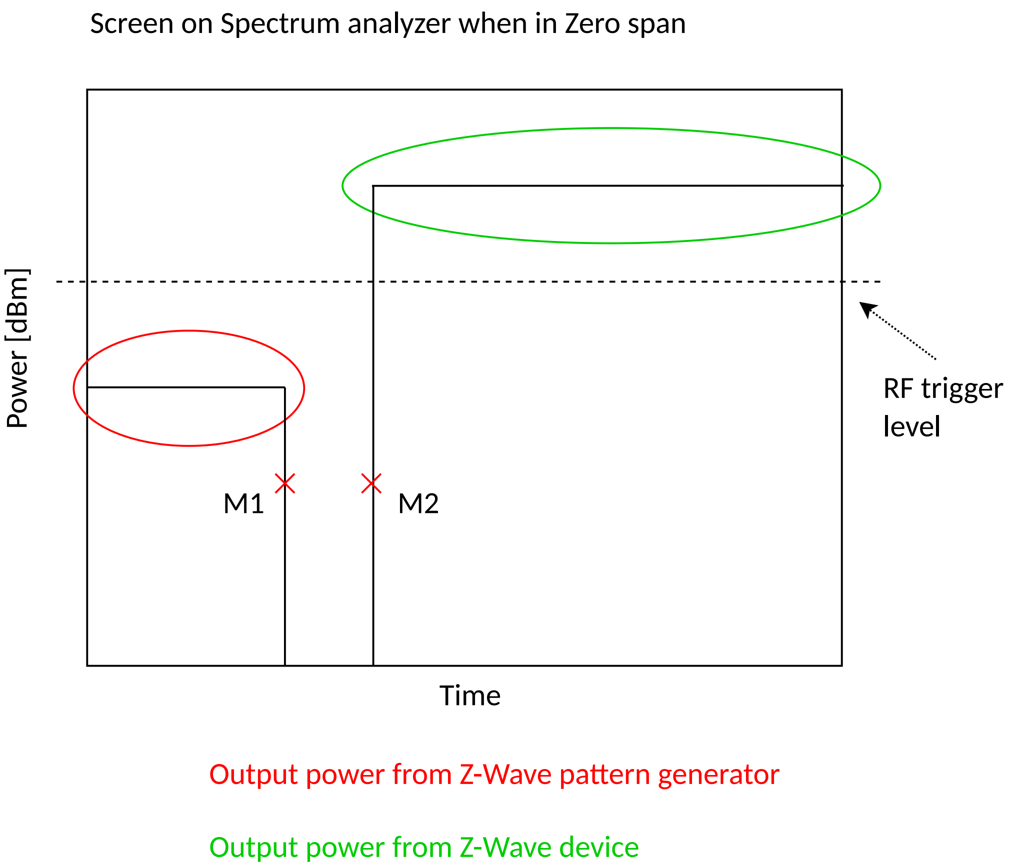

Figure 3.4: RX-to-TX turnaround measurement

The reply from the DUT will trigger the spectrum analyzer. Using the marker functionality of the spectrum analyzer, the RX-to-TX turnaround time can be calculated as $t_{rx\_to\_tx} = t_{M2} - t_{M1}$.

Further, the number of transmitted and received frames by the DUT must be recorded.

3.4.3 Test results

The measurement result is the time difference between the two markers in Figure 3.4, measured for at least 10 transmissions.

The DUT must have received and acknowledged all the frames transmitted by the Z-Wave pattern generator.

3.4.4 Pass criteria

The Z-Wave device shall pass the test if:

1. The RX-to-TX turnaround time, $t_{rx\_to\_tx}$ for at least 10 samples are greater than *aPhyTurnaroundTimeRXTX* in [LRMAC] table 5-27 and all transmitted frames by the Z-Wave generator were received and acknowledged by the Z-Wave device.

3.4.5 Fail criteria

The Z-Wave device shall fail the test if:

1. Any of 10 sampled RX-to-TX turnaround times, $t_{rx\_to\_tx}$ for at least 10 samples were less than *aPhyTurnaroundTimeRXTX* stated in [LRMAC] table 5-27 or not all transmitted frames by the Z-Wave generator were received and acknowledged by the Z-Wave device.

3.4.6 Exception

The output power transmitted by the parts in this test must be adjusted in order to ensure, that no receivers are overstressed or saturated. To avoid this, RF attenuators may be required to be inserted in the measurement setup as shown in Figure 3.4.

3.5 Format of MPDU, Singlecast in Long Range

A device must be able to produce the 2 types of frames: Single Cast & Acknowledge in Long Range [LRMAC] 6.1.3.1.

3.5.1 Prerequisites

- 1 x Sniffer.
- 1 x Controller
- 1 x LR End node

3.5.2 Test Setup

1. Include End Node to the Controller network.
2. Controller sends Singlecast frame to End Node with Data Payload (MSDU) = 0x00 (NOP).

3.5.3 Test Result

2. Verify on Sniffer that End Node responds with an Acknowledgement frame to the Controller (Header type: 0x03).

3.5.4 Pass Criteria

If the frames are displayed in the Sniffer, that means the PHY-layer header and EOF Delimiter are structured correctly (6.3.2).

1. The singlecast frame sent to the End Node has the format from [LRMAC] figure 6-4 (6.3.2).
2. The singlecast frame sent to the End Node has the frame type set to: 0x01 ([LRMAC]6.2.2.3.1 – Table 6-5).
3. The singlecast frame sent to the End Node has the ACK bit set to 0x01 ([LRMAC] 6.3.1.5)
4. The End Node responds with an Acknowledgement frame ([LRMAC] 6.1.3.2.2).
5. This Acknowledgement frame matches the description ([LRMAC] 6.1.3.1.2).
6. This Acknowledgement singlecast responded has the frame type set to: 0x03 ([LRMAC] 6.2.2.3.1 – Table 6-5).
7. The Ack Req bit (byte 8, bit 7) in the Acknowledgement frame is set to 0 ([LRMAC] 6.3.1.5.1 – Table 6-19).
8. This singlecast acknowledgement responded has the same HomeID as the sent singlecast ([LRMAC] 6.1.2).
9. This singlecast acknowledgement responded has the Destination ID set to the Node ID of the Controller ([LRMAC] 6.1.2).

3.5.5 Fail Criteria

1. The single cast does Not have the format described by figure 6-4 ([LRMAC] 6.3.2).
2. The singlecast frame sent to the End Node doesn't have the frame type set to: 0x01 ([LRMAC] 6.2.2.3.1 – Table 6-5).
3. The singlecast frame sent to the End Node doesn't have the ACK bit set to 0x01 ([LRMAC] 6.3.1.5).
4. The End Node did not respond using an Acknowledgement frame ([LRMAC] 6.1.3.2.2).
5. The Acknowledgement singlecast frame does not match the description ([LRMAC] 6.1.3.1.2).
6. This Acknowledgement singlecast responded does not have the frame type set to: 0x03 ([LRMAC] 6.2.2.3.1 – Table 6-5).
7. The ACK bit (byte 8, bit 7) in the Acknowledgement frame is NOT set to 0 ([LRMAC] 6.3.1.5.1 – Table 6-19).
8. This singlecast acknowledgement responded has a Different HomeID than the singlecast ([LRMAC] 6.1.2).
9. This singlecast acknowledgement responded has a different destination ID than the node ID of the Controller ([LRMAC] 6.1.2).

3.6 Network Robustness, Clear channel assessment in Long Range

A device must ensure robustness in data transmission. This is achieved by the mechanisms: Backoff Algorithm, Frame Acknowledgement, Data Verification and Frame Retransmission ([LRMAC] 6.1.3.2).

3.6.1 Prerequisites

- 1 x Sniffer
- 1 x Controller
- 1 x RF combiner
- 1 x Noise Generator (can either be: RF noise generator at the Z-Wave Long Range frequencies or Z-Wave modules loaded with RailTest configured to constant carrier transmission at the Z-Wave Long Range frequencies)

3.6.2 Test Setup

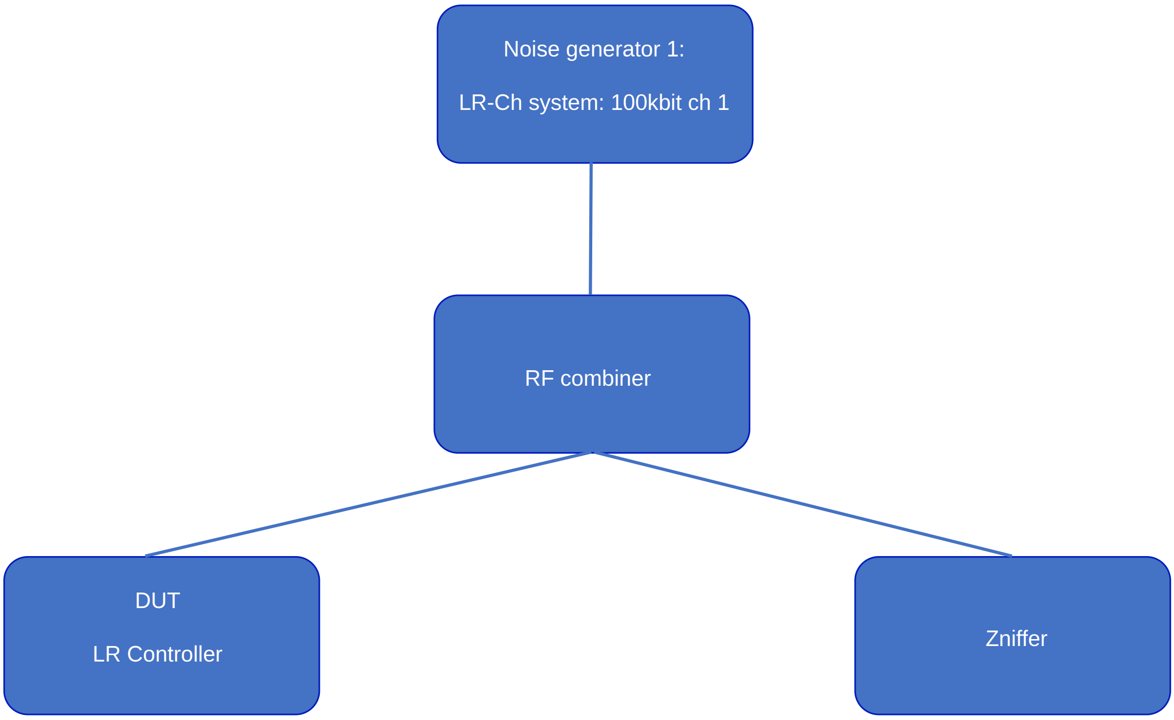

Figure 3.5: Connection structure for LR

1. Configure Noise Generator to operate 100kbit (RF noise = Constant carrier signal at the channel frequency, 0dBm).
2. Configure PC controller to send One Broadcast frame (HEX [00 00] – as a broadcast we don't expect answer). Don't send it yet.
3. Start the Noise Generator generating noise with 0dBm RF power.
4. On the Controller send the Broadcast. This is the reference start time for the next point.
5. After 0.1 second stop the Noise Generator. Wait 3 additional seconds to verify the frame is sent.
6. After 3 seconds more start the Noise Generator.

3.6.3 Test Result

1. Noise Generator is configured.
2. PC Controller is prepared to send a broadcast frame.
3. Noise Generator is generating noise.
4. Observe on Zniffer there is no traffic.
5. Observe on Zniffer the broadcast frame at 100kbit.
6. The Zniffer shows no new communication.

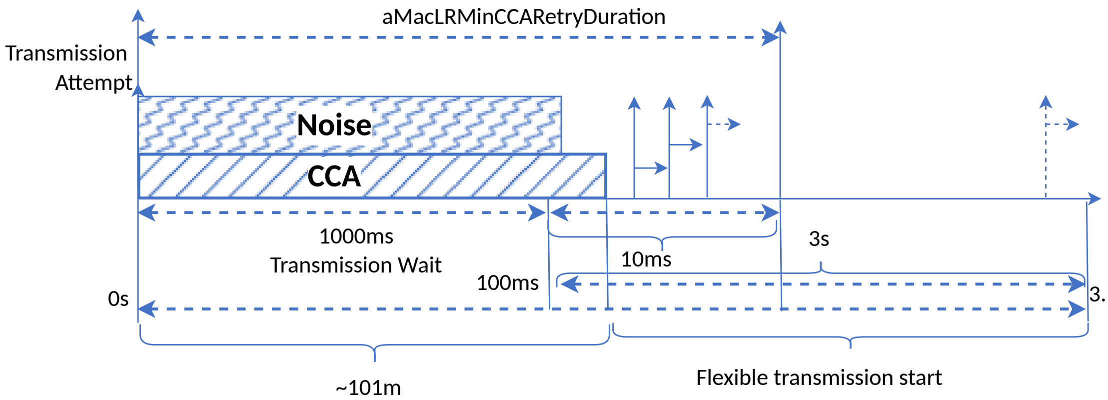

Figure 3.6: Clear channel Assessment time chart

3.6.4 Pass criteria

1. The Broadcast frames are sent while the corresponding channel is available and after the 100ms, use Figure 3.6 for reference. ([LRMAC] 6.1.3.2.1)
2. The Broadcast uses the channel with noise generator Switched off in steps 5 ([LRMAC] 6.1.3.2.1)

3.6.5 Fail criteria

1. The Broadcast frames appear before the noise is removed ([LRMAC] 6.1.3.2.1).
2. The Broadcast fails to be transmitted after step 5 ([LRMAC] 6.1.3.2.1).

3.7 Network Robustness, Acknowledgement in Long Range

A device must ensure robustness in data transmission. This is achieved by the mechanisms: Backoff Algorithm, Frame Acknowledgement, Data Verification and Frame Retransmission ([LRMAC] 6.1.3.2).

3.7.1 Prerequisites

- 1 x Sniffer
- 1 x Controller
- 1 x LR Not-Listening (NL) End node

3.7.2 Test Setup

1. Include End Node to the Controller network.
2. On the Sniffer observe that communication between Controller and End Node is possible.
3. Set on Controller a queue of 30 Version Get commands to send to the NL End Node. On the Command Classes view, enable "Expect Command" select Version Report for each command queued. Wait or wake the End Node up.
4. Once the commands start being sent after a couple of seconds, remove power, or do a power reset on the End Node, in order to prevent it from responding to the commands.
5. After a few seconds restore power to the End Node and let the Controller finish sending its command queue.

3.7.3 Test Result

2. The End Node answers only when it has awakened, and Controller waits until the End Node wakes up to transmit frames to it.
3. Observe on the Sniffer how the End Node replies to the controller's commands with one acknowledgement frame using the same sequence number present in each singlecast.
4. When the End Node does not answer, the controller retransmits the frame with the same sequence number. It then waits for the End Node to continue responding.
5. When the power returns to the End Node, the Controller waits for the End Node to answer with an acknowledgement frame and followed by responding to the query with the Version Report command as described.

3.7.4 Pass criteria

1. The singlecast frames are answered with an acknowledgement frame ([LRMAC] 6.1.3.2.2).
2. This Acknowledgement frame matches the description ([LRMAC] 6.3.3).
3. This Acknowledgement singlecast responded has the frame type set to: 0x03 ([LRMAC] 6.2.2.3.1– Table 6-5).
4. The ACK bit (byte 8, bit 7) in the Acknowledgement frame is set to 0 ([LRMAC] 6.3.1.5.1– Table 6-19).
5. This singlecast acknowledgement responded has the same HomeID as the sent singlecast ([LRMAC] 6.1.2).
6. This singlecast acknowledgement responded has the destination ID set to the node ID of the Controller that sent it ([LRMAC] 6.1.2).
7. The controller retransmits the frames that weren't answered with an Acknowledgement ([LRMAC] 6.1.3.2.3)
8. When the Controller retransmit the frames, the End Node answers with an acknowledgement using the same Sequence Number as the received frame ([LRMAC] 6.3.1.6).

3.7.5 Fail criteria

1. The End Node did not respond using an Acknowledgement frame ([LRMAC] 6.1.3.2.2).
2. The Acknowledgement singlecast frame does not match the description ([LRMAC] 6.1.3.3).
3. This Acknowledgement singlecast responded does not have the frame type set to: 0x03 ([LRMAC] 6.2.2.1.1– Table 6-3).
4. The Ack bit (byte 8, bit 7) in the Acknowledgement frame is NOT set to 0 ([LRMAC] 6.3.1.5.1– Table 6-19).
5. This singlecast acknowledgement responded has a Different HomeID than the singlecast ([LRMAC] 6.1.2).
6. This singlecast acknowledgement responded has a different destination ID than the node ID of the Controller ([LRMAC] 6.1.2).
7. The controller does not retransmit the frames that weren't answered with an Acknowledgement ([LRMAC] 6.1.3.2.3)
8. When the Controllers retransmit the frames, the End Node does not answer or answers with an Acknowledgement using a different Sequence Number as the received frame ([LRMAC] 6.3.1.6).

3.8 Network Robustness, Acknowledgement OFF in Long Range

A device must ensure robustness in data transmission. This is achieved by the mechanisms: Backoff Algorithm, Frame Acknowledgement, Data Verification and Frame Retransmission ([LRMAC] 6.1.3.2).

3.8.1 Prerequisites

- 1 x Sniffer
- 1 x Controller
- 1 x LR End node
- 1 x Frame Generator

3.8.2 Test Setup

1. Include End Node to the Controller network.
2. On the Sniffer observe that communication between the Controller and End Node is possible.
3. Generate a frame that sends to the End Node a singlecast with MDSU = 0x00 (NOP) making sure ACK bit (byte 8, bit 7) is set to 0x00.
4. Send this frame as singlecast to the End Node.

3.8.3 Test Result

4. Observe on the Sniffer how the End Node ignores the singlecast.

3.8.4 Pass criteria

1. The ACK bit (byte 8, bit 7) in the singlecast frame is set to 0 ([LRMAC] 6.3.1.5.1 – Table 6.19).
2. The singlecast frames are not answered with an acknowledgement frame ([LRMAC] 6.1.3.2.2).

3.8.5 Fail Criteria

1. The End Node did respond using an Acknowledgement frame ([LRMAC] 6.1.3.2.2).

3.9 Network Robustness, Retransmission

A device must ensure robustness in data transmission. This is achieved by the mechanisms: Backoff Algorithm, Frame Acknowledgement, Data Verification and Frame Retransmission ([LRMAC] 6.1.3.2).

3.9.1 Prerequisites

- 1 x Sniffer
- 1 x Controller
- 1 x LR End node

3.9.2 Test Setup

1. Include End Node to the Controller network.
2. Disable the End Node device by removing power or removing the antenna from it.
3. Send a singlecast from the controller to the End Node. With MSDU = 0x00 (NOP).
4. On the Sniffer observe that communication between both Devices is not possible and the Controller sending the frame re-tries sending it.

3.9.3 Test Result

4. Observe on the Sniffer the transmission is attempted up to 2 times more (the maximum number of frame transmission retries "aMacLRMaxFrameRetries") before increasing the Sequence Number and each re-transmission waits a random period to prevent collisions with other frames that may be being sent at the same time.

3.9.4 Pass criteria

1. The Controller sends only 2 retransmissions ("aMacLRMaxFrameRetries") with the same Sequence Number waiting a random period of time after each attempt ([LRMAC] 6.1.3.2.3).
2. The Controller issues a new frame with the same contents but with its Sequence Number value increased by one and sent also only up to the value of "aMacLRMaxFrameRetries" waiting a random period of time after each attempt ([LRMAC] 6.5.1.5.5).

3.9.5 Fail criteria

1. When the Controllers retransmit the frames, the sequence number changes each time and does it a different amount of times than the one defined by "aMacLRMaxFrameRetries" ([LRMAC] 6.1.3.2.3).
2. The Controller does not issue any new frame, or it issues them with a Sequence Number entirely unrelated to the previously used one ([LRMAC] 6.5.1.5.5).

3.10 Network Robustness, Data Validation Corrupt FCS, Long Range

A device must ensure robustness in data transmission. This is achieved by the mechanisms: Backoff Algorithm, Frame Acknowledgement, Data Verification and Frame Retransmission ([LRMAC] 6.1.3.2).

3.10.1 Prerequisites

- 1 x Sniffer
- 1 x Controller
- 1 x LR End node
- 1 x Frame Generator

3.10.2 Test Setup

1. Include End Node to the Controller network.
2. On the Sniffer observe that communication between both Controllers and End Node is possible.
3. Generate a frame that sends to the End Node a singlecast with MDSU = 0x00 (NOP) making sure the 8 bits for FCS are random and not generated automatically.
4. Send this frame as singlecast to the End Node.

3.10.3 Test Result

4. Observe on the Sniffer how the End Node ignores the singlecast

3.10.4 Pass criteria

1. The singlecast frame is not answered with an acknowledgement frame ([LRMAC] 6.1.3.2.4).

3.10.5 Fail criteria

1. The End Node does respond using an Acknowledgement frame ([LRMAC] 6.1.3.2.4).

3.11 General MPDU Format, Long Range

The MAC Protocol Data Unit (MPDU), consists of three basic components: A MAC Header (MHR), a MAC data payload (MAC Service Data Unit (MSDU)) and a MAC Footer (MFR) ([LRMAC] 6.3.1).

3.11.1 Prerequisites

- 1 x Sniffer
- 1 x Controller
- 1 x LR End node

3.11.2 Test Setup

1. Include the End Node to the Controller's Network.
2. Send a singlecast with MPDU = 0x00 (NOP) to one End Node.
3. Observe the structure of the singlecast sent.

3.11.3 Test Result

3. The singlecast is displalyed correctly on the Sniffer.

3.11.4 Pass Criteria

1. The singlecast shows:
    a. MHR: ([LRMAC] 6.1.3 – Figure 6-4)
        i. Home ID: 4 bytes ([LRMAC] 6.3.1.1)
        ii. Source Node ID: 12 bits ([LRMAC] 6.3.1.2)
        iii. Destination Node ID: 12 bits ([LRMAC] 6.3.1.3)
        iv. Length: 1 byte ([LRMAC] 6.3.1.4)
        v. Frame Control (8 bits): ([LRMAC] 6.3.1.5 – Table 6-19)
            1. Ack Req: 1 bit ([LRMAC] 6.3.1.5.1)
            2. Extended: 1 bit ([LRMAC] 6.3.1.5.2)
            3. Header type: 3 bits ([LRMAC] 6.3.1.5.3)
            4. Reserved: 3 bits ([LRMAC] 6.3.1.5.4)
        vi. Sequence Number: 8 bits ([LRMAC] 6.3.1.6)
        vii. Noise Floor: 8 bits ([LRMAC] 6.3.1.7)
        viii. Tx Power: 8 bits ([LRMAC] 6.3.1.8)
    b. MSDU: Payload: ([LRMAC] 6.3.1.9)
        i. 1 Byte = 0x00 (NOP)
    c. MFR: FCS (not described in the structure in the Sniffer): ([LRMAC] 6.3.1.10)
        i. FCS: 2 bytes

3.11.5 Fail Criteria

At Least one of the components of the format of the MPDU for singlecast has a different length or values ([LRMAC] 6.3.1).

3.12 MPDU Format, Home ID

The MAC Protocol Data Unit (MPDU), consists of three basic components: A MAC Header (MHR), a MAC data payload (MAC Service Data Unit (MSDU)) and a MAC Footer (MFR). Home ID are 4 bytes that identify all nodes in the same domain ([LRMAC] 6.3.1.1).

3.12.1 Prerequisites

- 1 x Sniffer
- 1 x Controller
- 2 x LR End node
- 1 x Frame Generator

3.12.2 Test Setup

1. Include both End Node devices to the Network of the Controller.
2. Verify the Controller can communicate with both End Nodes by sending Singlecast to each of them.
3. Send a singlecast to one of the End Nodes, modifying the Home ID to be different from the original value.
4. Send an S2 multicast (Broadcast) to both End Nodes with modified Home ID.

3.12.3 Test Result

2. Both End Nodes answer with an Acknowledgement frame as expected to the singlecast.
3. The End Node does not answer, since the Home ID is not the Home ID it has been included to.
4. Neither End Node answers since the Home ID is different from the one, they have been included to.

3.12.4 Pass Criteria

1. On the singlecast the Home ID occupies only 4 bytes ([LRMAC] 6.3.1.1)
2. No node responds to any frame that holds a modified Home Id in any way, because of mis-matching Home ID value ([LRMAC] 6.3.1.1)

3.12.5 Fail Criteria

1. Any of the methods for altering the Home ID Component is accepted by the receiving node and answered with an acknowledgement frame.

3.13 MPDU Format, Source NodeID

The MAC Protocol Data Unit (MPDU), consists of three basic components: A MAC Header (MHR), a MAC data payload (MAC Service Data Unit (MSDU)) and a MAC Footer (MFR). Source Node ID are 12 bits that identify the node within one domain that have transmitted the frame ([LRMAC] 6.3.1.2).

3.13.1 Prerequisites

- 1 x Sniffer
- 1 x Controller
- 2 x LR End node
- 1 x Frame Generator

3.13.2 Test Setup

1. Include End Node to Controller's network.
2. Verify the Controller can communicate with both End Nodes by sending Singlecast to each of them and S2 Multicast (Broadcast) to both.
3. Send a singlecast to one of the End Nodes, modifying the Source Node ID to be different from the original value.
4. Send an S2 Multicast (broadcast addressed to 0xFFF) to both End Nodes, modifying the Source Node Id to be different from the original value, the multicast is followed by singlecast follow-up frames (each addressed to the individual End Nodes).
5. Send a singlecast to one of the End Nodes, modifying the Source Node ID to be 0x000.
6. Send a singlecast to one of the End Nodes, modifying the Source Node ID to be a value between 0xFA6 & 0xFFF (reserved values, Table 6-15).

3.13.3 Test Result

3. End Node received the singlecast and responds to the modified Source Node ID with an Acknowledgement frame.
4. The End Nodes do not answer the S2 multicast (Broadcast) frame but answer to the singlecast follow-up frames originated after the S2 Multicast.
5. The End Node receiving the singlecast with Source Node ID set to 0x000, will answer to it with an Acknowledgement frame.
6. The End Node receiving the singlecast with Source Node ID set to a reserved value will not answer, since a network is limited to that number of nodes.

3.13.4 Pass Criteria

1. Each time the receiving End Node node answers to the modified Source Node ID. This happens for single cast, singlecast follow-up or frames addressed to Node 0. ([LRMAC] 6.3.1.2,)
2. The End Nodes won't answer to a frame with Source Node Id set to a reserved value (outside the valid values defined by Table 6-15),. ([LRMAC] 6.3.1.2)

3.13.5 Fail Criteria

1. The End Node answers to the controller with an Acknowledgement frame directly, ignoring the field Source Node ID ([LRMAC] 6.3.2.1)

3.14 MPDU Format, Destination Node ID

The destination Node ID specifies a destination node in the same domain identified by the HomeID. It shall comply with table 6-17 ([LRMAC] 6.3.1.3).

3.14.1 Prerequisites

- 1 x Sniffer
- 1 x Controller
- 3 x LR End Node

3.14.2 Test Setup

1. Include End Nodes to Controler's network.
2. Send a singlecast with MPDU = 0x00 (NOP) to each End Node.
3. Look for the Destination NodeID field in the frames on the Sniffer.

3.14.3 Test Result

2. Communication is correct. End Nodes answer with an Acknowledgement frame.
3. Destination NodeID is byte 12 in the frames. It holds the value of the End Node's NodeID.

3.14.4 Pass Criteria

1. The Destination Node ID is 12 bits in length. ([LRMAC] 6.3.1.3)
2. The Destination Node ID can be any value up to 0xFA6 (4006) ([LRMAC] 6.3.1.3)
3. The Destination Node ID is in Bytes 5 & 6 of the frame ([LRMAC] 6.3.1.3 – Table 6-16)

3.14.5 Fail Criteria

1. The Destination Node ID can be more than 12 bits in length ([LRMAC] 6.3.1.3).
2. The Destination Node ID can be any value beyond 0xFA6 (4006) ([LRMAC] 6.3.1.3).
3. Always Listening devices respond or try to route singlecast addressed to 0xFFF (Broadcast) ([LRMAC] 6.3.1.3).

3.15 MPDU Format, Length

The length field is 1 byte that indicates the length of the MPDU in bytes. It's limited by "aMacLR-MaxMSDUSize" defined on table 6-33. A receiving node shall not read more than the maximum length allowed ([LRMAC] 6.3.1.4).

3.15.1 Prerequisites

- 1 x Sniffer
- 1 x Controller
- 1 x LR End node

3.15.2 Test Setup

1. Include the End Node to the Controller's Network.
2. Send a singlecast with MPDU = 0x00 (NOP) to the End Node.
3. Look for the Length field.
4. Populate the MPDU with a large amount of random data (more than just 0x00 in the field "Send Data"), less than "aMacLRMaxMSDUSize" and send it to the End Node.

3.15.3 Test Result

2. Communication is possible and End Node answers with an Acknowledgement frame. Check on the Singlecast the Length field.
3. The Length field should be in Byte 7 of the frame and be show value 16 (0x10) for a NOP MPDU.
4. The singlecast should show the corresponding size in length.

3.15.4 Pass Criteria

1. The Length field is only one byte in length. ([LRMAC] 6.3.1.4)
2. The Length field is in byte 7 of the Frame ([LRMAC] 6.3.1.4– Figure 6-18)
3. The value of the Length field is always less or equal than "aMacLRMaxMSDUSize". ([LRMAC] 6.3.1.6)

3.15.5 Fail Criteria

1. The Length field is different from one byte in length. ([LRMAC] 6.3.1.6)
2. The Length field is located outside byte 7 of the Frame ([LRMAC] 6.3.1.6 – Figure 6-18)
3. The value of the length field can be more than "aMacLRMaxMSDUSize". ([LRMAC] 6.3.1.6)

3.16 MPDU Format, Frame Control, Header Type, Singlecast

The Frame Control field is 8 bits (1 byte) in length. It defines the frame type and other control flags. The header type defines the frame Header type. A broadcast MPDU is a singlecast MPDU (type 0x01) carrying destination Node ID = 0xFFF ([LRMAC] 6.3.1.5.3).

3.16.1 Prerequisites

- 1 x Sniffer
- 1 x Controller
- 1 x LR End node
- 1 x Frame Generator

3.16.2 Test Setup

1. Include the End Node to Controller's Network.
2. Send one single cast to the End Node with MPDU 0x00 (NOP).
3. Generate a frame with Header going from 0x02 to 0x0F according to table 6-20 and send it to the End Node.
4. Generate a frame with Header going from 0x02 to 0x0F according to table 6-20 and send it to Node ID 0xFFF (4095).

3.16.3 Test Results

2. Singlecast is sent correctly and it's answered with an Acknowledgement frame.
   a. Its Header type is set to 0x01.
3. Each frame sent to the node is displayed as the corresponding type on the Sniffer, it's ignored by the End Node and no Acknowledgement frame is responded.
4. Each frame sent to Node ID 4095 is displayed as the corresponding type on the Sniffer.

3.16.4 Pass Criteria

1. Each frame sent by the Controller in 3. & 4. Is displayed as the corresponding type on the Sniffer, making each frame correctly defined. ([LRMAC] 6.3.1.5.3)
2. None of the frames sent in 3. Are answered by definition.
3. None of the frames sent in 4. Are answered by definition nor by being addressed to a reserved Destination Node ID.

3.16.5 Fail Criteria

1. Any frame is displayed as singlecast on the Sniffer regardless of the different Header. ([LRMAC] 6.3.1.5.3)
2. Any frame sent in 3. Received an Acknowledgement frame.
3. Any frame sent in 4. Received an Acknowledgement frame.

3.17 MPDU Format, Frame Control, Header Type, Acknowledgement

The Frame Control field is 8 bits (1 byte) in length. It defines the frame type and other control flags. The header type defines the frame Header type. A broadcast MPDU is a singlecast MPDU (type 0x01) carrying destination Node ID = 0xFF ([LRMAC] 6.3.1.5.3).

3.17.1 Prerequisites

- 1 x Sniffer
- 1 x Controller
- 1 x LR Z-Wave Serial API End Node

3.17.2 Test Setup

1. Include the End Node to Primary Controller's network.
2. Send one singlecast to the End Node with MPDU 0x00 (NOP).

3.17.3 Test Result

2. The singlecast is sent correctly and it's answered with an Acknowledgement frame from the End Node.

3.17.4 Pass Criteria

1. The frame sent by the Controller in 2. Is answered with the corresponding type on the Sniffer,. ([LRMAC] 6.3.1.5.3, 6.3.3.2.2)

3.17.5 Fail Criteria

1. The frame is not displayed as Acknowledgement on the Sniffer ([LRMAC] 6.3.1.5.3, 6.3.3.2.2)

3.18 MPDU Format, Sequence Number

The sequence number is an 8-bit field provided by higher layers when transmitting. The same Sequence Number shall be used for all retransmissions of a given MPDU that first fails being delivered. A receiving node shall return the same value in an Acknowledgement frame if the Ack bit is present in the received frame ([LRMAC] 6.3.1.6).

3.18.1 Prerequisites

- 1 x Sniffer
- 1 x Controller
- 2 x LR End node

3.18.2 Test Setup

1. Include both End Nodes to the Primary Controller's network.
2. Disable one of the End Nodes and send a singlecast with MPDU = 0x00 (NOP) to it.
3. Enable the End Node again and try sending a singlecast to it again.
4. Select both End nodes and send an S2 multicast (Broadcast to 0xFFF) from the controller.

3.18.3 Test Result

2. Observe that the Controller re-tries sending the command to the disabled End Node and all frames have the same sequence number.
3. The Controller transmits directly and the frame correctly reaches the destination Node.
   a. The End Node Answers with an Acknowledgement frame using the same sequence number as the singlecast.
4. The S2 multicast frame and its respective single cast follow-up frames have their own sequence numbers

3.18.4 Pass Criteria

1. The re-transitted frames when the Controller doesn't reach the End Node have the same Sequence Number. ([LRMAC] 6.3.1.6)
2. The Ack frames have the same Sequence number as the original singlecast frame sent from the Controller. ([LRMAC] 6.3.1.6, 6.3.3.3)
3. The S2 Multicast and its follow-up singlecast have successive Sequence numbers. ([LRMAC] 6.3.1.6)
4. Sequence Number only has 8 bits going from 0x0 to 0xFF. ([LRMAC] 6.3.1.6)

3.18.5 Fail Criteria

1. The retransmitted frames have their own Sequence Number value. ([LRMAC] 6.3.1.6)
2. The Ack frames from the Controller to the repeater have different Sequence Number value. ([LRMAC] 6.3.1.6)
3. The S2 Multicast and its successive Follow-up singelcast frames have the same Sequence Number value. ([LRMAC] 6.3.1.6)
4. Sequence Number can have more than 8 bits of length. ([LRMAC] 6.3.1.6)

3.19 MPDU Format, Noise Floor

The Noise Floor field is an 8-bit signed field that indicates the radio noise level present on the channel the frame is being transmitted. Its format follows Table 6-22 ([LRMAC] 6.3.1.7).

3.19.1 Prerequisites

- 1 x Sniffer
- 1 x Controller
- 2 x LR End node

3.19.2 Test Setup

1. Include End Nodes to Controller's Network.
2. Send a regular singlecast to one End Node with MDPU = 0x00 (NOP).
3. Check the Noise Floor field in the Header section in the Sniffer for both the Singlecast and the Acknowledgement frame.
4. Select both End Nodes and send an S2 multicast (Broadcast) to both of them with MDPU = 0x00 (NOP).

3.19.3 Test Results

2. End Node answers with an Acknowledgement Frame to the Controller.
3. The Acknowledgement Frame has similar values as the Singlecast.
4. The End Nodes don't respond to the S2 Multicast (Broadcast) but respond to the singlecast follow-up frames.

3.19.4 Pass Criteria

1. Noise Floor consists of 8 bits at byte 10 of the Frame and only shows valid values ([LRMAC] 6.3.1.7, Table 6-22, Table 6-23).
2. The singlecast and Singlecast follow-up are sent with similar values, within the valid ones. ([LRMAC] 6.3.1.7 Table 6-22)
3. The End Nodes answer with an Acknowledgement frame in similar value to the Noise Floor as the singlecast.

3.19.5 Fail Criteria

1. Noise Floor is not 8 bits at byte 10 of the frame and holds different values from the valid ones ([LRMAC] 6.3.1.7, Table 6-22, Table 6-23).
2. The default singlecast are sent with different values and outside the valid ranges ([LRMAC] 6.3.1.7 Table 6-23).
3. The End Nodes answer to a regular singlecast with an Acknowledgement frame set to a different Noise Floor value than the singlecast.

3.20 MPDU Format, Tx Power

The Tx Power field is an 8 bit signed field specifying the transmit power used to transmit this frame defined by the table 6-25 (6.3.1.8).

3.20.1 Prerequisites

- 1 x Sniffer
- 1 x Controller
- 2 x LR End node

3.20.2 Test Setup

1. Include End Nodes to Controller's Network.
2. Send a regular singlecast to one End Node with MDPU = 0x00 (NOP).
3. Check the Tx Power field in the Header section in the Sniffer for both the Singlecast and the Acknowledgement frame.

3.20.3 Test Results

3. End Node answers with an Acknowledgement Frame to the Controller.
4. The Acknowledgement Frame has the same Tx Power value as the Singlecast.

3.20.4 Pass Criteria

1. The singlecast has a Tx Power field within the valid ones. ([LRMAC] 6.3.1.8 Table 6-25)
2. The End Node nodes answer with an Acknowledgement frame in the same Tx Power as the singlecast.

3.20.5 Fail Criteria

1. The singlecast is sent with a Tx Power value outside the valid ranges ([LRMAC]6.3.1.8 Table 6-25).
2. The End Nodes answer to a regular singlecast with an Acknowledgement frame set to a different Tx Power value.

3.21 MPDU Format, Mac Footer (MFR): FCS

FCS is the 16-bit non-correcting Frame Check Sequence (FCS) used for validating the integrity of a frame. It shall be calculated from the HomeID field to the Data Payload, both included ([LRMAC] 6.3.1.10).

3.21.1 Prerequisites

- 1 x Sniffer
- 1 x Controller
- 2 x LR End Node
- 1 x Frame Generator

3.21.2 Test Setup

1. Include End Nodes to the Controller's Network.
2. Send a singlecast with MPDU = 0x00 (NOP) to each End Node.
3. Look for the FCS field in each frame.
4. Generate a singlecast frame with MPDU = 0x00 (NOP) and modify its FCS value to a random value, send it to each End Node separately.
5. Send an unmodified S2 multicast (Broadcast) frame with MPDU = 0x00 (NOP) to both End Nodes as an S2 Multicast. Look for the FCS in the multicast.
6. Look for the FCS field in each singlecast follow-up frame.
7. Generate an S2 multicast (Broadcast) frame with MPDU = 0x00 (NOP) and modify its FCS value to a random value, send it to both End Nodes. Look for the FCS in the multicast.

3.21.3 Test Result

2. Communication is correct with both End Nodes and they answer with an Acknowledgement frame each.
3. Make sure there are 16 MFR bits corresponding to the FCS section in each frame.
4. Neither End Node answers to the frame since the FCS does not allow verification of its integrity.
    a. Each time the frame is displayed on the Sniffer as a CRC error.
    b. The controller tries retransmitting the frame since it doesn't receive an Acknowledgement frame.
5. The S2 Multicast is not responded with an Acknowledgement frame by either End Node.
    a. The FCS in the multicast frame is 16 bits long.
6. Same as in step 3.
7. The S2 multicast is not responded with an Acknowledgement frame by either End Node.
    a. The multicast frame is displayed on the Sniffer as a CRC error.
    b. The FCS in the multicast frame is 16 bits long.

3.21.4 Pass Criteria

1. The FCS can only be 16 bits long at the end of the frame ([LRMAC] 6.3.1.10)
2. It signalizes the integrity of the frame and when it is modified, the receiver and Sniffer are uncapable of identifying it ([LRMAC] 6.3.1.10)

3.21.5 Fail Criteria

1. The FCS can be different from 16 bits long at the end of the frame ([LRMAC] 6.3.1.10)
2. When it is modified, the receiver and Sniffer are able to identify and respond to the frame ([LRMAC] 6.3.1.10)

3.22 Acknowledgement MPDU Format

The Acknowledgement MPDU format uses the general MPDU format from Test 3.9. It must be returned in the same Long Range channel as the singlecast that triggered it and only when the singlecast has its Ack Req subfield in the Frame Control field set to 1 ([LRMAC] 6.3.3).

3.22.1 Prerequisites

- 1 x Sniffer
- 1 x Controller
- 1 x LR End node

3.22.2 Test Setup

1. Include the End Node to the Controller's Network.
2. Send a singlecast with MPDU = 0x00 (NOP) to the End Node.
3. Observe the structure of the Acknowledgement frame in the Sniffer.

3.22.3 Test Result

2. Communication is correct it answers with an Acknowledgement.
3. The Acknowledgement frame is displayed correctly on the Sniffer.

3.22.4 Pass Criteria

1. The Acknowledgement shows the same structure as the singlecast, with the following differences:
    a. Destination NodeID must be set to the NodeID value of the source NodeID of the singlecast that triggered the Ack frame: 12 bits ([LRMAC] 6.3.3.1).
    b. Frame Control, Ack Req subfield is set to 0: 1 bit ([LRMAC] 6.3.3.2.1).
    c. Frame Control, Header Type subfield is set to Acknowledgement (0x03): 3 bits ([LRMAC] 6.3.3.2.2, 6.3.1.5.3).
    d. Received RSSI is included before the Data Payload, it shows a valid value: 8 bits ([LRMAC] 6.3.3.4, Table 6-27).
    e. Data Payload contains any data ([LRMAC] 6.3.3.5).

3.22.5 Fail Criteria

1. At least one of the components of the format of the MPDU for Acknowledgement has a different length or holds a value different from the mandatory ones ([LRMAC] 6.3.3).

3.23 Acknowledgement MPDU Format, Received RSSI

The Received RSSI is an 8-bit signed field present only in Acknowledgement frames that indicates the signal strength measured while the corresponding singlecast frame is received. It averages at least 1 sample during reception and complies with the values in Table 6-27 ([LRMAC] 6.3.3.4).

3.23.1 Prerequisites

- 1 x Sniffer
- 1 x Controller
- 1 x LR Serial API End node

3.23.2 Test Setup

1. Include the End Node to the Controller's Network.
2. Send a singlecast with MPDU = 0x00 (NOP) to one End Node.
3. Observe the Received RSSI field in the Acknowledgement frame returned.
4. Place the End Node at another distance from the controller with at least 50 cm difference from the previous distance. Send a singlecast to it. Observe the RSSI value in the Acknowledgement frame answered.

3.23.3 Test Result

2. The singlecast is received and answered by the End Node with an Acknowledgement frame.
3. The Received RSSI field in the Acknowledgement holds values within the ones defined as valid in table 6-27.
4. The RSSI value changes in at least 1dB due to the difference in intensity of the received signal.

3.23.4 Pass Criteria

1. The RSSI field consists of 8 bits at Byte 12 of the Acknowledgement frame and its values are within the valid ones ([LRMAC] 6.3.3.4, Table 6-27).
2. The RSSI indicates the strength of the received singlecast depending on the distance between Controller and End Node ([LRMAC] 6.3.3.4).

3.23.5 Fail Criteria

1. The RSSI field is not 8 bits long and its values can be outside the valid ones ([LRMAC] 6.3.3.4, Table 6-27).
2. The RSSI is not corresponding to the strength of the received singlecast ([LRMAC] 6.3.3.4).

3.24 Broadcast MPDU Format

The Broadcast MPDU format uses the general MPDU format from Test 3.9. A broadcast MPDU is a singlecast MPDU (type 0x01) carrying destination Node ID = 0xFFF ([LRMAC] 6.3.4).

3.24.1 Prerequisites

- 1 x Sniffer
- 1 x Controller
- 2 x LR End node

3.24.2 Test Setup

1. Include both End Nodes to the Controller's Network.
2. Send a singlecast with MPDU = 0x00 (NOP) to each End Node.
3. Select both End Nodes and send a frame with MPDU = 0x00 (NOP) to both End Nodes as an S2 Multicast (Broadcast).
4. Observe the structure of the Broadcast (S2 Multicast) in the Sniffer.

3.24.3 Test Result

2. Communication is correct with both End Nodes and they answer with an Acknowledgement frame each.
3. The Controller sends a Broadcast frame directed to NodeID 0xFFF (4095) followed by singlecast follow-up frames to each End Node.
4. The Broadcast is displayed correctly on the Sniffer.

3.24.4 Pass Criteria

1. The Broadcast shows the same structure as the singlecast, with the following differences:
   a. Destination NodeID must be set to the broadcast NodeID value 0xFFF: 12 bits ([LRMAC] 6.3.4.1, 6.3.1.3).
   b. Frame Control, Ack Req subfield is set to 0: 1 bit ([LRMAC] 6.3.4.2.1).
   c. Frame Control, Header Type subfield is set to Singlecast (0x01): 3 bits ([LRMAC] 6.3.4.2.2, 6.3.1.5.3).
   d. Data Payload shall contain at least 1 byte of data ([LRMAC] 6.3.4.3).

3.24.5 Fail Criteria

1. At least one of the components of the format of the MPDU for broadcast has a different length or holds a value different from the mandatory for Broadcast ([LRMAC] 6.3.4).

3.25 MPDU Header Extension Format

The extended MPDU header format is an extension to the General MPDU frame format as in test 3.9. Follows the Format from Table 6-28 and the values from Table 6-29 ([LRMAC] 6.3.5). This functionality is not currently used in the protocol and therefore it might need special test tools to generate the frames.

3.25.1 Prerequisites

- 1 x Sniffer
- 1 x Controller
- 2 x LR End node
- 1 x Frame Generator

3.25.2 Test Setup

1. Include both End Nodes to the Controller's Network.
2. Send a singlecast with MPDU = 0x00 (NOP) to each End Node.
3. Build a singlecast frame that sets the Extended subfield in the Frame Control field to 1; includes an Extension Control Field and Extension Data Field, in which the Extension Control Field describes correctly the contents of the Extension Data field. Send it to the End Node.
4. Observe the structure of the singlecast in the Sniffer.

3.25.3 Test Result

2. Communication is correct with both End Nodes and they answer with an Acknowledgement frame each.
3. The Frame Generator sends a singlecast with the configured values.
4. The Singlecast contains the Extension Control and Extension Data fields as well as the Extended subfield in the Frame Control field is enabled, as configured.

3.25.4 Pass Criteria

1. The Extended subfield in the Frame Control field is set to 1 ([LRMAC] 6.3.1.5.2).
2. The Extension Control field is located in the Byte 12 of the frame and consists of 8 bits of length ([LRMAC] 6.3.5.1).
3. The extension Type subfield consists of only 3 bits from bits 4 to 6 of the Frame Control field and it follows the values of Table 6-29 ([LRMAC] 6.3.5.1.1)
4. The Discard unknown subfield is only bit number 3 in the Extension Control field and it's set depending on whether the Controller considers possible to discard the Extension Data in case the receiving end doesn't know the Extension Type.
5. The extension Length subfield consist of three bits (0 – 2) in the Extension Control field and holds the value of the number of bytes that the Extension Data field holds ([LRMAC] 6.3.5.1.3).

3.25.5 Fail Criteria

1. At least one of the components of the format of the Extension for singlecast has a different length or holds a value different from the ones configured ([LRMAC] 6.3.5.1).

3.26 Beam Frame MPDU Format

Beam frames are used to awake Frequently Listening (FL) nodes. They are transmitted back to back to ensure an FL node can detect a beam within a short time window. Each beam frame shall carry the Beam Tag and NodeID fields. The NodeID field should be followed by the optional HomeID Hash field. An FL node shall stay awake to receive the MPDU that follows if there is a match with the Hash or NodeID, else it may return to sleep ([LRMAC] 6.3.6).

3.26.1 Prerequisites

- 1 x Sniffer
- 1 x Controller
- 1 x LR FL End node

3.26.2 Test Setup

1. Include FL End Node to the Controller's Network.
2. After inclusion, reset/power cycle FliRS to wake it up and send right away a singlecast with MPDU 0x00 (NOP) to the End node.
3. Wait 10 seconds to make sure FL is asleep. Send the frame again.
4. Observe the Beam frame is sent to the End Node.
5. Observe the Beam Stop frame.
6. Once the beaming reaches the FL in a waking up state, the FL stays awake so that the Controller tries with a transmission of the original singlecast again.

3.26.3 Test Result

2. Communication is correct and FL End Node answers with an Acknowledgement frame.
3. Observe that Controller can't deliver the frame and retransmits it.
    a. As soon as it fails, it starts sending Beam frames to the FL End Node.
4. The Beam is shown correctly in the Sniffer.
5. The Beam Stop frame shows some number of beam count.
6. After the beaming, the Controller tries with the original singlecast sent and the FL End Node answers with an Acknowledgement frame in response.

3.26.4 Pass Criteria

1. The Beam frame consists of: ([LRMAC] 6.3.6 – Figure 6-8)
    a. A Beam tag: 0x55 (1 byte). ([LRMAC] 6.3.6.1 – Table 6-30)
    b. Tx Power (4 bits). ([LRMAC] 6.3.6.2 – Table 6-31)
    c. Destination NodeID: 0x100 .. 0xFA0, 0xFFF (12 bits). ([LRMAC] 6.3.6.3, 6.3.1.3 – Table 6-15)
    d. (Optional) Field "HomeID Hash": (1 byte). ([LRMAC] 6.3.6.4)

3.26.5 Fail Criteria

1. Any of the elements of the Beam frame deviates from the description ([LRMAC] 6.3.6 – Table 6-30 – Table 6-15)

3.27 Fragmented Frame MPDU Format

A Beam Fragment comprises a number of beam frames. The Beam Fragment duration is in the range 110-115 ms. Beam frames shall be sent back to back to ensure the FL node can detect it upon waking up. The next Beam Fragment shall begin 190 – 200 ms after the beginning of the previous one. They shall be sent in different channels. When recognizing a Beam the receiving node shall answer with an Acknowledgement frame, upon receiving it, the Controller shall send the original singlecast but only if the Acknowledgement frame matches the originating HomeID and the NodeID of the destination NodeID on the original singlecast. A Beam Fragment can be addressed to 0xFFF (4095) turning it into a broadcast Wake Up Beam, but it can't be answered directly by the End Node ([LRMAC] 6.3.7).

3.27.1 Prerequisites

- 1 x Sniffer
- 1 x Controller
- 1 x LR FL End node

3.27.2 Test Setup

1. Include FL End Node to the Controller's Network.
2. Wait 10 seconds for FL to sleep. Send a singlecast with MPDU 0x00 (NOP) to the End Node.

3.27.3 Test result

2. Observe there are 2 wake up Beam frames sent to the End Node.
   a. Each Beam Fragment is sent between 80 and 90ms after the Beam Stop from the previous Beam Fragment (as to begin between 190 – 200 ms from the beginning of the previous Beam Fragment).
   b. When the End Node recognizes the Beam, it answers with an Acknowledgement frame.
   c. Controller repeats the original singlecast.

3.27.4 Pass Criteria

1. There are more than one wake up Beam frames sent to the End Node ([LRMAC] 6.3.7).
2. The fragment lasts between 110-115ms ([LRMAC] 6.3.7).
3. There are between 190 – 200ms between two Wake Up Beam Start in the Fragment ([LRMAC] 6.3.7).
4. The Fragments are sent in different Channels (A & B) ([LRMAC] 6.3.7).
5. The Receiving Node can validate an (optional) Hash of the HomeID ([LRMAC] 6.3.7).
6. The End Node answers the Beam with an Acknowledgement frame and the Controller repeats the original singlecast ([LRMAC] 6.3.7, 6.3.7.1).

3.27.5 Fail Criteria

1. Any of the Pass Criteria is not met.

3.28 (3.14 - Negative Testing) MPDU Format, Destination Node ID

The destination Node ID specifies a destination node in the same domain identified by the HomeID. It shall comply with table 6-17 ([LRMAC] 6.3.1.3).

3.28.1 Prerequisites

- 1 x Sniffer
- 1 x Controller
- 2 x LR End node
- 1 x Frame Generator

3.28.2 Test Setup

We assume a Frame Generator is available in order to generate frames with individual bits, bytes or sections modified individually in order to test the behavior of the receiver.

1. Include End Nodes to Controler's network.
2. Generate a singlecast frame with Destination ID value different to either End Node, send it.
3. Generate a singlecast frame with Destination ID value higher than 0xFA6 (4006), send it.
4. Generate a singlecast frame with Destination ID value of 0xFFF (4095), send it.

3.28.3 Test Result

2. The Frame tries to reach this End Node but can't reach it.
3. The Frame tries to reach this End Node but can't reach it.
4. The Controller sends this frame as a Broadcast.
    a. The End Nodes don't respond to this Broadcast frame.

3.28.4 Pass Criteria

1. The Destination Node ID can be any value up to 0xFA6 (4006) ([LRMAC] 6.3.1.3)

3.28.5 Fail Criteria

1. The Destination Node ID can be any value beyond 0xFA6 (4006) ([LRMAC] 6.3.1.3).

3.29 (3.15 – Negative Testing) MPDU Format, Length

The length field is 1 byte that indicates the length of the MPDU in bytes. It's limited by "aMacLR-MaxMSDUSize" defined on table 6-33. A receiving node shall not read more than the maximum length allowed ([LRMAC] 6.3.1.4).

3.29.1 Prerequisites

- 1 x Sniffer
- 1 x Controller
- 1 x LR End node
- 1 x Frame Generator

3.29.2 Test Setup

We assume a Frame Generator is available in order to generate frames with individual bits, bytes or sections modified individually in order to test the behavior of the receiver.

1. Include the End Node to the Controller's Network.
2. Generate a singlecast with MPDU = 0x00 (NOP) and modify the Length field to be more than 16 and less than 59. Send it to the End Node.
3. Generate a singlecast with MPDU = 0x00 (NOP) and modify the Length field to be more than 59 Bytes. Send it to the End Node.
4. Generate a singlecast populating the MPDU with a long amount of random data, less than "aMacLRMaxMSDUSize" and modify the value of the Length field to be 16. Send it to the End Node.

3.29.3 Test Result

2. When the End Node receives it, the stated size and the actual size as well as the FCS values do not correspond, and the End Node ignores the singlecast.
   a. The Controller tries re-transmitting the same singlecast because of not receiving an Acknowledgement frame.
3. When the End Node receives it, the stated size is larger than "aMacLRMaxMSDUSize" and the End Node ignores the frame.
   a. The Controller tries re-transmitting the same singlecast because of not receiving an Acknowledgement frame.
4. When the End Node receives it, the stated size of the frame is smaller than it actually is and the End Node ignores the frame.
   a. The Controller tries re-transmitting the same singlecast because of not receiving an Acknowledgement frame.

3.29.4 Pass Criteria

1. The receiving node ignores all instances where the Length field does not match the actual length of the frame. ([LRMAC] 6.3.1.6)

3.29.5 Fail Criteria

1. The receiving node accepts and answers with an Acknowledgement frame any frame regardless of the size and value of the Length field. ([LRMAC] 6.3.1.6)

3.30 (3.17 – Negative Testing) MPDU Format, Frame Control, Header Type, Acknowledgement

The Frame Control field is 8 bits (1 byte) in length. It defines the frame type and other control flags. The header type defines the frame Header type. A broadcast MPDU is a singlecast MPDU (type 0x01) carrying destination Node ID = 0xFF ([LRMAC] 6.3.1.5.3).

3.30.1 Prerequisites

- 1 x Sniffer
- 1 x Controller
- 1 x LR Z-Wave Serial API End Node
- 1 x Frame Generator

3.30.2 Test Setup

We assume a Frame Generator is available to generate frames with individual bits, bytes or sections modified individually in order to test the behavior of the receiver.

1. Include the End Node to Primary Controller's network.
2. Generate an Acknowledgement frame with Header going from 0x01 to 0x0F (except 0x03) according to table 6-20 to be answered by the End Node to the Primary Controller and send a singlecast to the End Node for each header type.
3. Generate an Acknowledgement frame with Header going from 0x01 to 0x0F (except 0x03) according to table 6-20 to be answered by the End Node to Node ID 0xff (255) and send a singlecast to the End Node for each header type.

3.30.3 Test Result

2. Each frame answered is constructed as an Acknowledgement frame, but with the corresponding header from table 6-20.
    a. Therefore, it's displayed on the Sniffer as the expected type.
3. Each frame answered is constructed as an Acknowledgement frame, but with the corresponding header from table 6-20 and addressed to Node ID 0xFF.
    a. Therefore, it's displayed on the Sniffer as the expected type except for type 0x01, showing as a Broadcast.

3.30.4 Pass Criteria

1. Each frame sent by the Controller in 2. is answered with the corresponding type on the Sniffer, making each frame correctly defined. ([LRMAC] 6.3.1.5.3, 6.3.3.2.2)
2. None of the frames sent in 2. Are answered by the Primary Controller by definition.
3. None of the frames sent in 3. Are answered by the Primary Controller by definition.

3.30.5 Fail Criteria

1. Any frame is displayed as Acknowledgement on the Sniffer regardless of the different Header. ([LRMAC] 6.3.1.5.3, 6.3.3.2.2)
2. Any frame sent in 2. received an Acknowledgement frame.
3. Any frame sent in 3. received an Acknowledgement frame.

3.31 (3.18 – Negative Testing) MPDU Format, Sequence Number

The sequence number is an 8-bit field provided by higher layers when transmitting. The same Sequence Number shall be used for all retransmissions of a given MPDU that first fails being delivered. A receiving node shall return the same value in an Acknowledgement frame if the Ack bit is present in the received frame ([LRMAC] 6.3.1.6).

3.31.1 Prerequisites

- 1 x Sniffer
- 1 x Controller
- 2 x LR End node
- 1 x LR Z-Wave Serial API End Node
- 1 x Frame Generator

3.31.2 Test Setup

We assume a Frame Generator is available in order to generate frames with individual bits, bytes or sections modified individually in order to test the behavior of the receiver.

1. Include both End Nodes and Serial API End Node to the Primary Controller's network. Disable one of the End Nodes.
2. In the Frame Generator: Configure an Acknowledgement frame so that its Sequence Number has a random non-zero static value instead of the End Node when the Primary Controller transmits a singlecast to it and proceed to send a singlecast to the disabled End Node.
3. Configure a singlecast frame with static sequence number on the Frame Generator and send it twice or more to one of the End Nodes instead of the Primary Controller.
4. Configure a singlecast frame with static sequence number set to 0 on the Frame Generator and send it twice or more to one of the End Nodes instead of the Primary Controller.

3.31.3 Test Result

2. When receiving the Singlecast, the End Node (Frame Generator) responds with the generated Acknowledgement frame with the static Sequence Number value.
   a. When the Controller received this Acknowledgement frame with a Sequence Number value different from the singlecast it transmitted, it tries to retransmit the frame again. Since this is equivalent to not having received the proper Ackowledgement frame.
3. The End Node receiving the Singlecast answers correctly to the first one with an Acknowledgement frame using the same Sequence Number.
   a. In the following frames, it ignores the frame, as it holds the same Sequence Number value as previous frames.
4. The End Node receiving the Singlecast with Sequence Number set to 0 answers to it.
   a. The Primary Controller tries transmits the following attempted frames with the same Sequence Number set to 0.
   b. The receiver it ignores those frames.

3.31.4 Pass Criteria

1. Acknowledgement frames with value that do not match the one of the singlecast that originated them are rejected by the Controller and re-transmitted by the receiving node. ([LRMAC] 6.3.1.6)

3.31.5 Fail Criteria

1. All Acknowledgement frames with non-zero value different from the singlecast that originated them are accepted. ([LRMAC] 6.3.1.6)

3.32 (3.19 – Negative Testing) MPDU Format, Noise Floor

The Noise Floor field is an 8-bit signed field that indicates the radio noise level present on the channel the frame is being transmitted. Its format follows Table 6-22 ([LRMAC] 6.3.1.7).

3.32.1 Prerequisites

- 1 x Sniffer
- 1 x Controller
- 2 x LR End node

3.32.2 Test Setup

We assume a Frame Generator is available to generate frames with individual bits, bytes or sections modified individually in order to test the behavior of the receiver.

1. Include End Nodes to Controller's Network.
2. Generate a Frame that has a custom random value set on Noise Floor field, send it to one End Node.
3. Send an S2 Multicast to both End Nodes. Observe the Noise Floor field.

3.32.3 Test Results

2. The End Node answers to the Controller with an Acknowledgement frame with a similar Noise Floor value.
3. The Multicast is sent with the same value as the Singlecast follow-ups.

3.32.4 Pass Criteria

1. The modified singlecast is responded with an Acknowledgement frame set in the same Noise Floor.
2. The Multicast is not answered by any of the End Nodes ([LRMAC] 6.3.3).

3.32.5 Fail Criteria

1. The End Nodes answer to the modified singlecast with an Acknowledgement frame set to a different Noise Floor value.
2. The End Nodes answer to the Multicast frame directly with an Acknowledgement frame ([LRMAC] 6.3.3).

3.33 (3.20 – Negative Testing) MPDU Format, Tx Power

The Tx Power field is an 8 bit signed field specifying the transmit power used to transmit this frame defined by the table 6-25 (6.3.1.8).

3.33.1 Prerequisites

- 1 x Sniffer
- 1 x Controller
- 2 x LR End node
- 1 x Spectrum Analyzer
- 1 x Frame Generator

3.33.2 Test Setup

We assume a Frame Generator is available to generate frames with individual bits, bytes or sections modified individually in order to test the behavior of the receiver.

1. Configure Spectrum Analyzer to detect traffic in the RF determined for LR.
2. Include End Nodes to Controller's Network.
3. From Frame Generator, generate a Frame that has a random value set on Tx Power field, send it to one End Node.
4. From Controller, send an S2 Multicast (Broadcast to 0xFFF) to both End Nodes. Observe the Tx Power field.

3.33.3 Test Results

3. The End Node answers to the Frame Generator with an Acknowledgement frame in the same TX Power.
    a. Observe in the Spectrum Analyzer that the Acknowledgement frame was detected in the corresponding Power Level.
4. The Multicast is sent in higher Power while the Singlecast Follow-up frames are sent in the default value.

3.33.4 Pass Criteria

1. The modified singlecast is responded with an Acknowledgement frame set in the same Tx Power value.
2. The Multicast is not answered by any of the End Nodes ([LRMAC] 6.3.3).

3.33.5 Fail Criteria

1. The End Nodes answer to the modified singlecast with an Acknowledgement frame set to a different Tx Power value.
2. The End Nodes answer to the Multicast frame directly with an Acknowledgement frame ([LR-MAC] 6.3.3).

3.34 (3.27 – Negative Testing) Fragmented Frame MPDU Format

A Beam Fragment comprises a number of beam frames. The Beam Fragment duration is in the range 110-115 ms. Beam frames shall be sent back to back to ensure the FL node can detect it upon waking up. The next Beam Fragment shall begin 190 – 200 ms after the beginning of the previous one. They shall be sent in different channels. When recognizing a Beam the receiving node shall answer with an Acknowledgement frame, upon receiving it, the Controller shall send the original singlecast but only if the Acknowledgement frame matches the originating HomeID and the NodeID of the destination NodeID on the original singlecast. A Beam Fragment can be addressed to 0xFFF (4095) turning it into a broadcast Wake Up Beam, but it can't be answered directly by the End Node ([LRMAC] 6.3.7).

3.34.1 Prerequisites

- 1 x Sniffer
- 1 x Controller
- 1 x LR FL End node
- 1 x Frame Generator

3.34.2 Test Setup

We assume a Frame Generator is available to generate frames with individual bits, bytes or sections modified individually in order to test the behavior of the receiver.

1. Include FL End Node to the Controller's Network.
2. In Frame Generator: Generate a Wake up Beam Frame with Beam Tag different from 0x55, send it to the End Node.
3. In Frame Generator: Generate a Wake Up Beam Frame with a Destination ID different from the End Node, send it to the End Node.
4. In Frame Generator: Generate a Wake Up Beam Frame with a Destination ID 0xFFF "Broadcast", send it to the End Node.
5. In Frame Generator: Generate a Wake Up Beam Frame with a random HomeID Hash hardcoded, send it to the End Node.
6. In Frame Generator: Generate a Wake Up Beam Frame and set it to be delayed more 300ms, send it to the End Node.

3.34.3 Test Result

2. Observe there are 2 wake up Beam frames sent to the End Node.
   a. The End Node never recognizes the beam and the Controller continues sending the Wake Up Fragment until it times out.
3. Observe there are 2 wake up Beam frames sent to the End Node.
   a. The End Node never recognizes the beam and the Controller continues sending the Wake Up Fragment until it times out.
4. Observe there are 2 wake up Beam frames sent to the End Node.
   a. The End Node recognizes the beam but does not respond to the "Broadcast" Wake Up Fragment with an Acknowledgement Frame, the Controller continues sending the Wake Up Fragment until it times out.
5. Observe there are 2 wake up Beam frames sent to the End Node.
   a. The End Node never recognizes the beam and the Controller continues sending the Wake Up Fragment until it times out.
6. Observe there are 2 wake up Beam frames sent to the End Node with 300ms between transmissions.
   a. The End Node never manages to catch the beam when waking up.

3.34.4 Pass Criteria

1. The Beams can be addressed to any node ([LRMAC] 6.3.7).
2. The End Node will answer to a Fragmented Beam addressed to NodeID 0xFFF if macLRenableFLBroadcast is set to 1 and the HomeID Hash corresponds to its own, otherwise it won't answer with an acknowledgement frame ([LRMAC] 6.3.7.1)

3.34.5 Fail Criteria

1. Any of the Pass Criteria is not met.

References

[2] Z-Wave Alliance, ZWA_Z-Wave Long Range PHY Layer Test Specification_SPE_1.x.

[1] Z-Wave Alliance, ZWA_Z-Wave Long Range PHY and MAC Layer Specification_SPE_1.x.

