



Instruction

Z-Wave Network Installation and maintenance Procedures User Guide

Document No.:	INS12712
Version:	6
Description:	-
Written By:	NTJ;JFR;JSI;PSH;BBR
Date:	2018-03-02
Reviewed By:	JFR;PSH;JSI;BBR;CRASMUSSEN;SRO
Restrictions:	Public

Approved by:

Date	CET	Initials	Name	Justification
2018-03-02	15:51:35	JFR	Jørgen Franck	on behalf of NTJ

This document is the property of Silicon Labs. The data contained herein, in whole or in part, may not be duplicated, used or disclosed outside the recipient for any purpose. This restriction does not limit the recipient's right to use information contained in the data if it is obtained from another source without restriction.



REVISION RECORD

Doc. Ver.	Date	By	Pages affected	Brief description of changes
1	20130926	NTJ	All	Initial version
2	20131216	JSI	Appendix E	Updated UI description
3	20150312	JSI	Appendix A Appendix A.4 Appendix E Appendix A.6 Appendix B.2	Added sub chapters Added description of LWR RSSI measurement Updated console UI description with new commands Updated Table with LWR_RSSI Updated SerialAPI description with DevKit 6.6x specifics
4	20161026	PSH	4.3 & Appendix A.3	Clarified the handling of FLiRS nodes during PowerLevel test
5	20161110	PSH	Appendix C	Updated test timing to fit 6.xx developer's kits
6	20171206	JFR	Appendix A.6 Section 3.2	Added green traffic light to NHV = 6 and 7 to improve verification when adding repeater nodes. Clarified verification during installation for yellow and red traffic light.
6	20180302	BBR	All	Added Silicon Labs template

Table of Contents

1	ABBREVIATIONS.....	1
2	INTRODUCTION.....	1
2.1	Purpose	1
2.2	Audience and prerequisites	1
3	DESCRIPTION.....	2
3.1	Z-Wave communication.....	2
3.1.1	Acknowledge.....	2
3.1.2	Transmit power reduction.....	2
3.1.3	Routing.....	2
3.1.4	Explorer Search	3
3.1.5	Network-Wide Inclusion	3
3.2	Installation and maintenance application	3
3.3	On-site installation and maintenance	5
3.4	Central service center monitoring and support	8
4	INSTALLATION AND MAINTENANCE PROCEDURES.....	9
4.1	Installation	9
4.1.1	Prepare gateway for new device or new repeater	10
4.1.2	Add new device.....	10
4.1.3	Was the new device added?	10
4.1.4	More devices to add?	11
4.1.5	Was all devices added successfully?.....	11
4.1.6	Try adding failed devices again	11
4.1.7	Move failing device or place repeater	11
4.1.8	Was any devices moved after inclusion?.....	11
4.1.9	Create associations and assign return routes.....	12
4.1.10	Installation complete	12
4.2	Remove devices from the network.....	12
4.2.1	Removal of devices in direct range.....	12
4.2.2	Local reset of devices	12
4.2.3	Remove using an exclusion tool	12
4.3	Preventive maintenance.....	14
4.4	Remotely maintenance.....	16
4.5	On-site maintenance	16
	REFERENCES	36
	INDEX.....	37

Table of Figures

Figure 1, Diagnosing a network.....	5
Figure 2, Installation of a repeater.....	6
Figure 3, Remote diagnostics.....	8
Figure 4, Installation flowchart.....	9
Figure 5, Maintenance flowchart	15
Figure 6, Determining margin of all links between source and destination device.	18

Table of Tables

Table 1, Network health levels 4

Table 2, Network health scale 22

1 ABBREVIATIONS

Abbreviation	Explanation
IMA	Installation and Maintenance Application that resides on the gateway
LAN	Local Area Network
LWR	Last Working Route
NHM	Network Health Measurement
NP	Normal Power
NWI	Network-Wide Inclusion
NVM	Non-Volatile Memory

2 INTRODUCTION

2.1 Purpose

The purpose of this document is to define a service provider installation and maintenance procedure, which can ensure an easy installation and provide an operational qualification of the installation. In addition, this document defines an installation and maintenance application running on the gateway that supports the abovementioned procedure.

2.2 Audience and prerequisites

The audience is Z-Wave partners and Silicon Labs.

3 DESCRIPTION

The following sections give an overview of the installation and maintenance scenarios. The built-in features in the protocol to ensure reliable communication are also addressed.

3.1 Z-Wave communication

The Z-Wave protocol uses a number of mechanisms to ensure that communication in the network is as reliable as possible. The protocol is designed to always try to get a frame to its destination even if the network topology has changed, nodes are unavailable or something has become unstable. With respect to details about the routing algorithm for a given library refer to [1].

The following sections give a short description of the mechanisms used by the protocol to ensure reliable communication.

3.1.1 Acknowledge

All single destination frames (singlecast) in Z-Wave are acknowledged so the sender of the frame knows if the destination received the frame. In case acknowledgement is not received the transmission will be retried two additional times before the protocol decides what to do next.

Multi destination (multicast and broadcast) frames in Z-Wave does not have acknowledgement handshake so if an application wants to make sure that a multi destination frame is received it should follow up with a singlecast frame to each destination in question.

3.1.2 Transmit power reduction

Normally Z-Wave frames will be sent at the highest possible transmit power. But there are situations where the Z-Wave protocol will reduce the transmit power to ensure that there is a good signal reception margin on the communication link. When Z-Wave reduces the transmit power to test a link, the reduced power level will be indicated in the frame and the resulting acknowledgement will also be transmitted at reduced power. This is done to assure that a link has the desired signal reception margin in both directions so problems with asymmetrical links are avoided. The Z-Wave protocol will reduce the transmit power when finding neighbors during inclusion, when finding a route via explorer search and when using the Powerlevel Command Class.

3.1.3 Routing

The Z-Wave protocol uses routing to communicate with nodes that are out of direct range. Routing in Z-Wave is based on a routing table in the controllers that is built during installation of the network. The routing table can be rebuilt by the controller at any time.

When a controller needs to send a routed frame to a destination it will calculate the best route to the destination and send the frame using that route. Routed frames are retransmitted and acknowledged as described in section 3.1.1. If a routed frame fails all retransmissions the Z-Wave protocol will calculate alternative routes and send the frame again using these routes until the frame gets to the destination or the max number of routing attempts is reached.

When a frame is sent, the route that was used successfully will always be saved in NVM as a Last Working Route (LWR) so the routing algorithm knows what route to try first the next time.

3.1.4 Explorer Search

Explorer Search is a reactive route discovery method in Z-Wave that is used when direct or routed communication does not work anymore because of major changes to the network topology or the RF environment. Explorer Search is a broadcast-based search algorithm that will propagate an explorer frame through the network so it is received by all nodes in the network. When the explorer frame reaches the destination, the frame carries a list of the traversed repeater nodes. The target node now has a route it can return to the requesting node. The Last Working Route (LWR) in the source and destination nodes will be updated with the route found with Explorer Search.

Explorer Search provides a working route within a typical time of 300ms (worst case 3.5 seconds). Explorer Search generates a significant network load when no node answers the search request, so it is used as a last resort when calculated routes fail.

3.1.5 Network-Wide Inclusion

Network-Wide Inclusion (NWI) allows a Z-Wave node to be installed anywhere in a Z-Wave network; even when the new node is not within direct range of the gateway. NWI also uses explorer frames to propagate an inclusion request from the new node to the gateway.

3.2 Installation and maintenance application

The service provider Installation and Maintenance Application (IMA) resides on the gateway in the Z-Wave network and is accessed via the gateway LAN interface. The application is intended to be used for both on-site visits and remotely by a central service center. The installer will connect to the IMA application through a standard device e.g. tablet computer and gain access to the current network health and tools useful for diagnostics. The central support center can also access the IMA application via the Internet to provide installer or customer support etc. Furthermore, the IMA application will make it possible to set network health criteria's so critical conditions are reported to a central service center.

The application determines the network health (Appendix A) based on the following measurements:

- Jitter (variation in the time packets arriving)
- Number of repeating neighbors
- Packet error rate
- RF communication link margin

The network health is measured between the gateway and the Z-Wave node in question during normal operation. In addition, all network health measurements are stored for making network health available at any time. An operator can also initiate network health measurements for diagnostic purposes. Z-Wave protocols v4.5x and 6.x support network health measurements. However, the Z-Wave node application must also support the Z-Wave Powerlevel Command Class. Network health measurements are aggregated into a simple indicator for each Z-Wave node.

Instructions are provided for nodes where the network health is not sufficient and action is needed to ensure optimal network health in the Z-Wave network.





Network Health	Interpretation and actions
	<p>Network health is good.</p> <p>ACTIONS: No actions are needed. The installation leverages on route resolution mechanisms to assure a robust and reliable Z-Wave network. This mechanism is handled solely by the Z-Wave protocol when using the TRANSMIT_OPTION_EXPLORE flag.</p>
	<p>Network health is acceptable but latency can be observed occasionally.</p> <p>ACTIONS:</p> <p><u>Installation:</u> The installer MUST move nodes or install repeater nodes to achieve green traffic lights on all nodes. Verify changes by activating node in question several times to trigger a new route calculation and perform a new traffic light calculation to get network health updated.</p> <p><u>Maintenance:</u> No immediate actions are needed and the installation leverages on route resolution mechanisms to assure a robust and reliable Z-Wave network. Nodes with potential problems are flagged for rediscovery. Rediscovery of flagged nodes MUST be performed once a day to rebuild the routing information and resolve the potential problems. Assignment of return routes MUST be done after a rediscovery.</p>
	<p>Network health is insufficient because frames are dropped.</p> <p>ACTIONS:</p> <p><u>Installation:</u> The installer MUST move nodes or install the necessary repeaters to achieve green traffic lights on all nodes. Verify changes by activating node in question to trigger a new route calculation and perform a new traffic light calculation to get network health updated.</p> <p><u>Maintenance:</u> No immediate actions are needed and the installation relies on route resolution mechanism to assure a robust and reliable Z-Wave network. Nodes with potential problems are flagged for needing rediscovery. Rediscovery of flagged nodes MUST be performed once a day to rebuild the routing information and resolve the potential problems. Assignment of return routes MUST be done after a rediscovery.</p> <p>REMOTE ACTIONS: Customer interaction can be conducted to understand what potential change has caused the problem to find a solution. An additional node rediscovery can be initiated. Assignment of return routes MUST follow rediscovery.</p> <p>ON-SITE ACTIONS: If the problem couldn't be solved remotely, a technician may have to be dispatched. The Z-Wave node in question must either be moved closer to the gateway or a repeater node must be inserted between the Z-Wave node and the gateway.</p>
	<p>Network health is critical because Z-Wave node is not responding at all.</p> <p>ON-SITE ACTIONS:</p> <ol style="list-style-type: none"> 1. Alert the user about the problem and instruct him to check that all Z-Wave nodes are powered. 2. In case the Z-Wave node is powered, instruct the user to check local operation of the node. 3. Perform a new network health check after the user has checked Z-Wave nodes. 4. In case network health is not green, perform a full rediscovery. 5. If network health is still critical notify call center about the problem, and initiate replacement of the node or installation of a repeater node.

Table 1, Network health levels

3.3 On-site installation and maintenance

When service personnel is dispatched for an on-site installation/maintenance task it is important that an installation problem can be resolved both quickly and correctly. It cannot be expected that the service personnel has an in-depth knowledge of Z-Wave so any issues in a network must be presented in a simple and intuitive way.

When service personnel arrives on-site, a tablet computer is connected to the gateway to get an overview of the current network's health. All Z-Wave nodes in the network will be displayed with one of the states described in the previous section.

The figure below illustrates the status of a network being diagnosed.

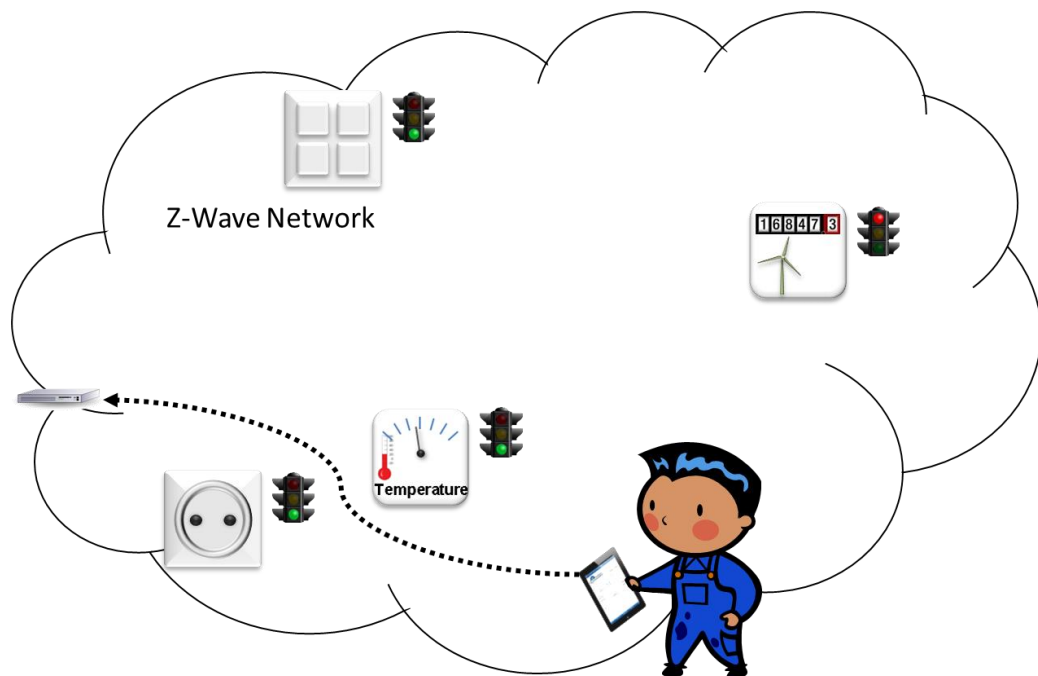


Figure 1, Diagnosing a network

The service personnel can see on the tablet computer that the network health of a node is insufficient. Instructions are given to either move the node closer to the gateway or insert a repeater node. In this example the node cannot be moved since it is used for meter reading. The installer will therefore have to install a repeater node to improve the overall network health to green level.

The installer will now follow this instruction for improving connectivity:

1. Move the troublesome node closer to the gateway
2. If (1.) is not an option, insert a repeater node 30 feet away from the troublesome node in the direction towards the gateway.
 - a. Convenience may dictate a location less than 30 feet from the troublesome node.
 - b. If the repeater node gets closer to the gateway than 30 feet, the node may be placed less than 30 feet from the troublesome node.
3. If a repeater node 30 feet away from the troublesome node cannot be included, insert a repeater node 60 feet away from the troublesome node in the direction towards the gateway.
 - a. In case of continued inclusion trouble, continue moving closer in 30 feet steps until the repeater can be included
4. Try to include the troublesome node
5. If the troublesome node still cannot be included, one more repeater node is needed. Repeat the process; starting from (2.)

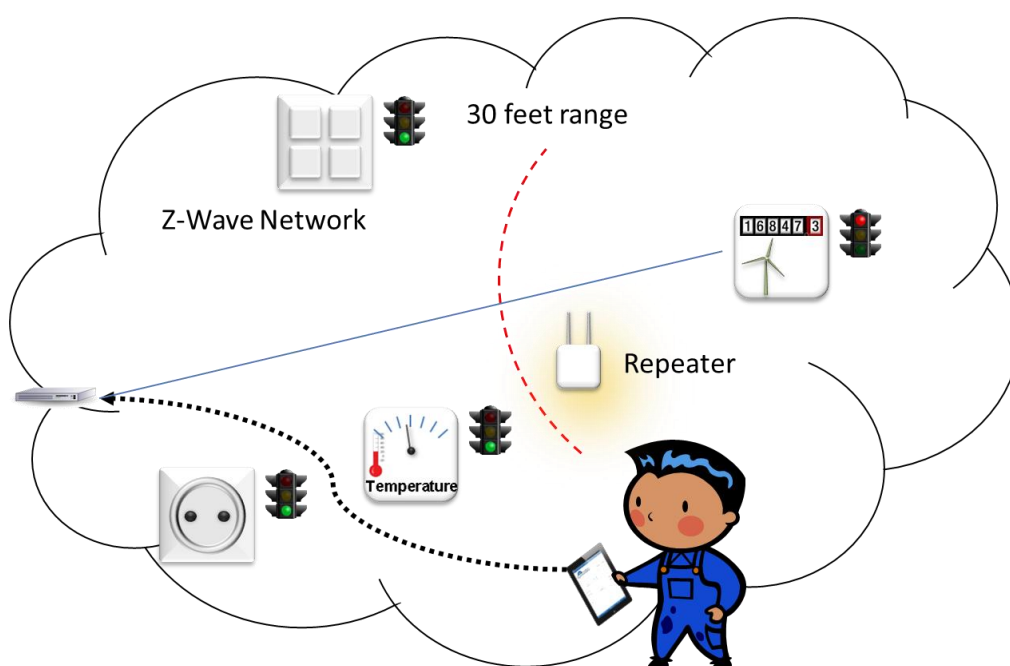


Figure 2, Installation of a repeater

When the repeater node has been installed the installer will launch IMA application and get an updated status of the network health where the improved network health is displayed. The verification steps of the added repeater node are as follows:

1. Check that troublesome node now has a green traffic light after insertion of a repeater node.
2. Check that the repeater node has a green traffic light.
3. Check that the repeater node now has the former troublesome node as a neighbor.
4. Check that the repeater node has at least one additional neighbor having green traffic light.

3.4 Central service center monitoring and support

The IMA application can be accessed remotely over the Internet making it accessible to a service providers call center. The service center can also be informed if network health drops below a predefined threshold for a given installation. This could e.g. be due to a failing node or a link with intermittent network health properties.

The call center will be able to resolve issues by interacting with the customer and utilizing the diagnostic tools provided by the application. The call center will also be able to judge if a node must be replaced and whether this can be done by dispatching a new unit to the customer for “self install” or dispatching a service technician.

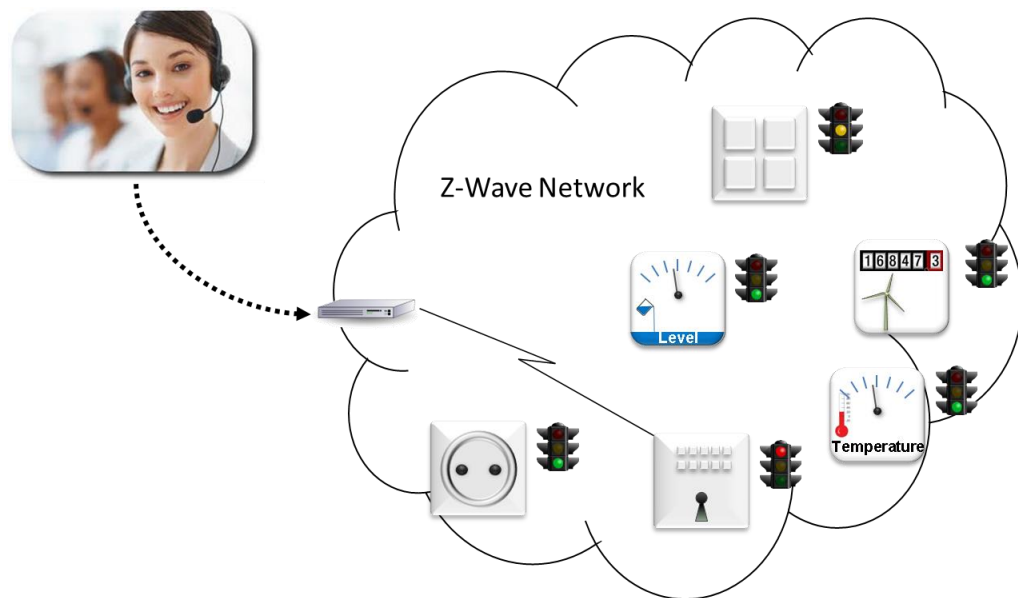


Figure 3, Remote diagnostics

4 INSTALLATION AND MAINTENANCE PROCEDURES

The following sections describe in details the complete installation and maintenance procedures.

4.1 Installation

This section describes how an installer should handle installation and commissioning of a Z-Wave network.

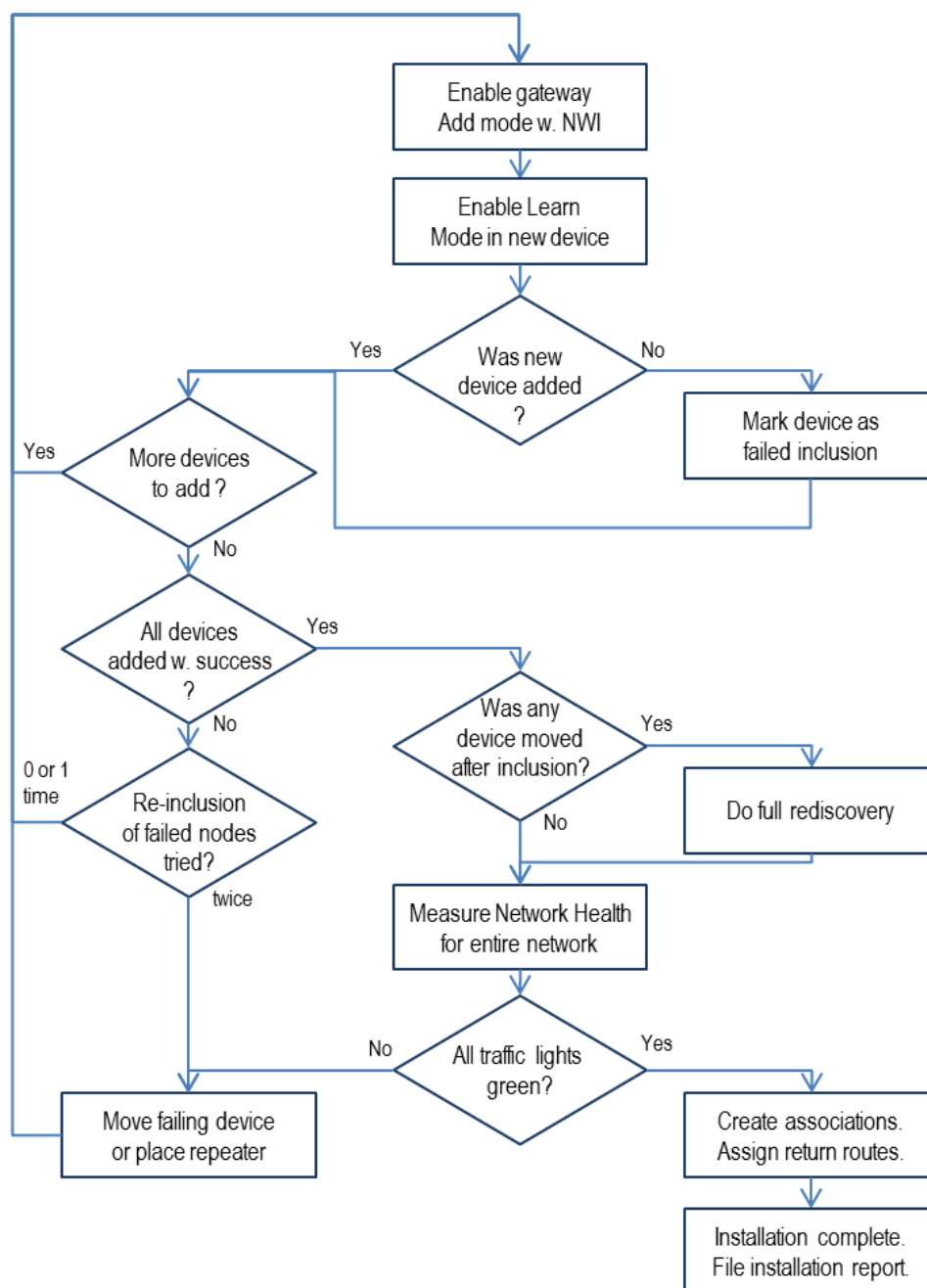


Figure 4, Installation flowchart

The installation flowchart assumes that all Z-Wave devices are already physically installed. The tablet computer is connected to the gateway and ready to start creating the Z-Wave network.

4.1.1 Prepare gateway for new device or new repeater

Installer: Use the tablet computer to enter a name for the new node. Select “Add new device” or “Add new repeater”. The installer MAY check a “Single string network” checkbox next to the “Add new repeater” pushbutton.

The tablet computer enables Add mode with NWI in the gateway. A message box is displayed to the installer: “Place repeater 30 feet from failing device towards gateway – or move it another 30 feet closer to the gateway”.

4.1.2 Add new device

New devices should be added starting with devices closest to the gateway. This way, already included devices may serve as repeaters for new devices.

Installer: Set the device in Learn mode to include it to the Z-Wave network. Learn mode may be enabled by pushing a button or choosing a menu entry.

4.1.3 Was the new device added?

Installer: Use the tablet computer to verify that the new device now appears in the device list with name and address information and that the new device has a green traffic light.

The IMA application checks the following if the installer chose “Add new repeater”:

1. Check that troublesome device now has a green traffic light after insertion of repeater.
2. Check that the repeater has a green traffic light.
3. Check that the repeater now has the former troublesome device as a neighbor.
4. Check that the repeater has at least one additional neighbor having green traffic light.

These operations are handled automatically and the result is presented to the installer as a single red, yellow or green traffic light for the new device. If the IMA application recognizes the network topology to be a single string network, or if the installer checked the “Single string network” checkbox, the repeater installation can be supported by use of Powerlevel Command Class. For details refer to Appendix D. This is handled automatically by the tablet computer.

4.1.3.1 New device was not added successfully

Installer: Using the tablet computer, the installer can see that the new device was not added successfully. The device appears in the device list with the text “failing” in place of the address information and a red exclamation mark indicates that this device needs attention. It is recommended to postpone this until all other devices have been added (or an attempt has been made to add).

4.1.3.2 New repeater was not added successfully

Installer: Using the tablet computer, the installer can see that the new repeater was not added successfully. The repeater appears in the device list with the text “failing” in place of the address information and a red exclamation mark indicates that this repeater needs attention.

The IMA application shows a message box “Move repeater 30 feet closer to the gateway and try adding the repeater again”. Refer to 4.1.7.

4.1.4 More devices to add?

Check if all devices are shown on the node list, i.e. installer has at least tried to install all devices once.

Installer: Use the tablet computer to verify that all new devices now appear in the device list. Names are correct but the address information of some devices may say “failing”.

A service provider may provide installers with a preloaded electronic list of devices to install. Devices should be presented in a neutral color with no traffic light and the text “not added” along with an “Add device” pushbutton next to each device in the list.

4.1.5 Was all devices added successfully?

Installer: The tablet computer shows a simple page with one green traffic light and a line of text saying “Network health is good”. The computer shows a pushbutton: “Create device associations”.

Refer to 4.1.9.

The IMA application uses the following criterion to determine that the network health is good:

All devices added successfully have at least one neighbor, which has a RF communication link margin equal to or better than 6dB ($\frac{2}{3}$ of a typical indoor range equal to 100 feet). The routing table used for route calculations therefore has the same built-in 6dB margin.

4.1.6 Try adding failed devices again

Other devices added since the previous attempt may provide the required repeater functionality that caused a device to fail.

Installer: Try to add devices that failed previously one more time. This may prevent the installation of repeaters that are not needed. If there are failing devices left in the device list after the first cycle, another cycle of re-addition attempts should be carried out.

4.1.7 Move failing device or place repeater

In case of sparse networks, large building complexes or difficult RF conditions, one or more devices may remain failing even after two attempts of re-adding failing devices.

Installer: If repeated attempts of adding a device fail, investigate if it is possible to move the device closer to another healthy device. If it is not possible to move the device – or if moving the device within the possible limits does not alleviate the problems, it may be necessary to add a repeater.

Refer to 4.1.1.

4.1.8 Was any devices moved after inclusion?

Moving one or more devices in an existing Z-Wave network may jeopardize the RF communication link margin.

Installer: If any device was moved after it was added to the network, the installer MUST do a full network re-discovery. Following the network re-discovery, the IMA application MUST re-evaluate the network health of the entire network.

Refer to 4.1.5

4.1.9 Create associations and assign return routes

After establishing a robust and reliable Z-Wave network it is time to create the associations. Return routes must be assigned when making slave to slave or slave to controller associations. All associations are created via the tablet computer and return routes are assigned automatically by the gateway when relevant.

Installer: When the tablet computer reports “Network health is good”, the installer may push the “Create device associations” pushbutton. Once the installer has created the required associations, the installer **MUST** test that all configured functionality works as intended.

4.1.10 Installation complete

The installer must complete the network installation by generating a full report containing a device list including network health measurements etc. and store the report on the tablet computer. The report can also be sent to central service center as documentation of installation.

4.2 Remove devices from the network

Z-Wave does currently not support network wide exclusion of devices so when excluding a nodes from the network that are out of range of the gateway another exclusion strategy must be used.

4.2.1 Removal of devices in direct range

When excluding a device in direct range of the gateway it can be done in a similar way as inclusion.

1. Put the gateway in exclusion mode by calling `ZW_RemoveNodeFromNetwork()`
2. Put the device in learn mode by calling `ZW_SetLearnMode()` and send out a node information frame by calling `ZW_SendNodeInformation()`

Removal of the node should now be done by the protocol and the gateway will remove information about the deleted device and the device will be reset to default. If the device is out of direct range the remove process will not start on the controller.

Devices out of direct range will need to be moved into direct range of the gateway or the gateway will need to be moved in order to exclude nodes out of direct range with this method.

4.2.2 Local reset of devices

If the devices in the network supports being reset to factory default (`ZW_SetDefault()`) locally by some kind of physical interaction (button press) the device can be removed from the network by activating the local reset on the device. However this will not remove the information about the device from the gateway so the gateway will need to poll all devices to find out what device it was that was reset and then the gateway needs to do a remove failed node on that device (`ZW_RemoveFailedNode()`) This will ensure that the device is removed from the network in the gateway.

4.2.3 Remove using an exclusion tool

Using a portable Z-Wave controller that is not a part of the network will enable an installer to exclude nodes that are not in direct range of the gateway. The installer can use the exclusion tool to always exclude a device in direct range by moving the portable controller to the device that should be excluded from the network and then perform the exclusion. However this will not remove the information about the

device from the gateway so the gateway will need to poll all devices to find out what device it was that was reset and then the gateway needs to do a remove failed node on that device
(ZW_RemoveFailedNode()) This will ensure that the device is removed from the network in the gateway.

4.3 Preventive maintenance

The maintenance network health algorithm is responsible for monitoring the health of all devices in the network during normal operation. The algorithm will automatically measure the health of devices and make repairs to the network when needed.

The maintenance network health algorithm will poll all mains powered and FLiRS devices and monitor PER and RC for the last 10 transmissions and an accumulated $RC_{Lifetime}$ that is incremented every time there is a RC and only reset when a rediscovery is done on the node or on the network. These monitors will trigger network health checks, rediscovery, user alerts and call center alerts.

The maintenance health algorithm is designed to detect as many potential problems as possible in the network and resolve them without involving the user or the call center.

The algorithm uses the network health algorithm (see Appendix A.6) and rediscovery of devices to detect and resolve problems. But because these mechanisms generates a lot of network traffic and can interfere with normal operation of the network the repair is not done immediately if network functionality is still intact but postponed to a time when the network is idle.

Dynamic route resolution (explorer frame) in the Z-Wave protocol will assures a robust and reliable Z-Wave communication during normal operation of the network. To obtain this the API call `ZW_SendData` MUST use the following transmit option flags when sending application frames:

- `TRANSMIT_OPTION_ACK`
- `TRANSMIT_OPTION_AUTO_ROUTE`
- `TRANSMIT_OPTION_EXPLORER`.

The maintenance health algorithm does the following:

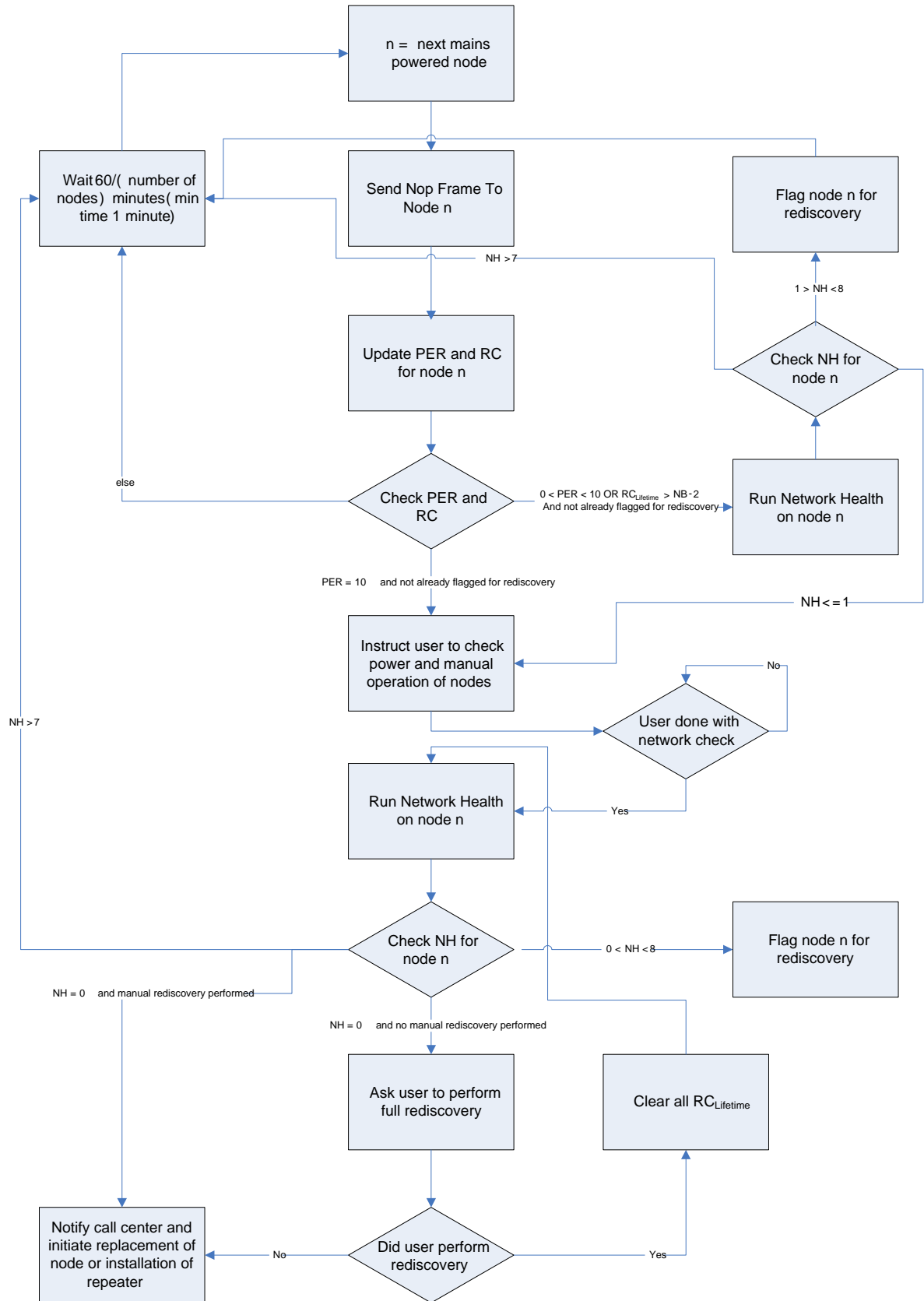


Figure 5, Maintenance flowchart

The IMA software that resides on the gateway will automatically monitor network health on a scheduled basis. All mains powered nodes in the network will be checked at least once every hour. However, battery operated FLiRS devices need special attention to preserve battery lifetime. In this case the network health measurement is done once a day and also on application oriented communication to the FLiRS devices. See section Appendix A.3 for further details on FLiRS handling and power level test.

In case the maintenance network health gives yellow or red traffic lights the node(s) with the problem will be flagged for later rediscovery. And if communication to the device is completely failing a user alert and a rediscovery will be performed.

The rediscovery of devices will generate a lot of traffic in the network and can disturb normal operation of the network. So the rediscovery should be done at a time where it does the least disturbance to the user of the network. Determining the best time for rediscovery depends of the type of devices and the use of the network.

In case the maintenance network health is critical the user must be alerted about the failure and instructed to check the power and local operation of devices. The most common cause of a critical failure in a running Z-Wave network is that power has been removed from the device or that the device no longer functions at all. When the user has checked the devices a new network health check is performed and if the health check still shows problems a full rediscovery of the network is performed. If the problem is not solved the call center must be informed about the problem and a replacement of a device or installation of a repeater must be initiated.

A network rediscovery is conducted at a RF communication link margin better than 6dB so after the rediscovery all links in the routing table will have a margin of 6dB.

4.4 Remotely maintenance

This section describes remotely maintenance performed by the service providers call center and if appropriate in conjunction with the end-user.

The central service center must have remote access to the following IMA software features:

- Perform rediscovery of individual devices in the Z-Wave network.
- Perform full network rediscovery of the whole Z-Wave network.
- Perform network health of individual devices in the Z-Wave network.
- Perform network health of the whole Z-Wave network.
- Perform assignment of return routes

4.5 On-site maintenance

This section describes on-site maintenance performed by the installer.

The installer repairs defect devices etc. Repair is finished when all devices again obtain a green traffic light with respect to network health.

As a final step, installer may use the IMA software to report a full node list containing network health parameters to the central service center.

APPENDIX A NETWORK HEALTH ALGORITHM

The network health is based on a number of measurements. In combination the measurements give a value between 0 and 10 as a network health estimate for the device in question. The measurements are repeated 10 times for each device and the network health is then calculated. The process is repeated 6 times and an average of the 6 network health calculations will be the final network health.

The network health is a combination of measurements that include the performance of the RF, the route diversity and the network topology of the tested network. The network health gives a number for how well the network will perform in real use when the routing protocol uses all its features to find alternative routes from the routing table and new paths via dynamic route resolution. The network Health also gives an indication of how stable the operation of the network will be over time.

Appendix A.1 Timing, Packet Error Rate and Route Change Measurements

T_{app} = Time the application uses to process a serial API command. T_{app} must be measured in a resolution of minimum 1 millisecond.

T_{app} can be measured by starting a timer in the host application when the serial API command ZW_GetSUCNodeID has been sent and stopped again when the result is received by the host application.

T = Time from ZW_SendData() [1] returns to callback is received.

T is measured via the serial API from the host application and must be measured at a resolution of minimum 1 millisecond. A timer is started when getting the return value TRUE from the API call ZW_SendData until a callback equal to TRANSMIT_COMPLETE_OK is received by the host application. The API call ZW_SendData MUST use the following transmit option flags:

- TRANSMIT_OPTION_ACK
- TRANSMIT_OPTION_AUTO_ROUTE
- TRANSMIT_OPTION_EXPLORER.

PER = Number of failed transmissions

PER is measured via the serial API from the host application. PER is incremented each time the application receives a callback from ZW_SendData with the status value TRANSMIT_COMPLETE_NO_ACK

RC = Number of times the protocol needed additional routes

RC is the number of times the protocol needed additional routes to reach a destination device because of a transmit failure. The number is a combination of Last Working Route (LWR) changes and jitter measurements during transmission attempts between the gateway and the Z-Wave device.

RC is incremented when either of the conditions below is true:

1. Last Working Route is different after call n to ZW_SendData() than it was after call $n-1$ to ZW_SendData()
2. $T_{(n)} - T_{(n-1)} > 150\text{ms} + T_{app}$

Appendix A.2 Repeater Neighbor measurements

NB = Number of repeater neighbors

The number of repeaters that a given device has as neighbors in the network. The number of repeater neighbors can be found using the serial API command ZW_GetRoutingInfo(NodeID, FALSE, TRUE)

Appendix A.3 Last Working Route Powerlevel test

LWRdB = The maximum reduction in transmit power where the LWR still works

The node IDs of the LWR (LWR or direct route) used for the destination device is retrieved from the Z-Wave protocol. This node IDs are used to check that the 6dB margin obtained during installation of the Z-Wave network is still valid. However, all devices used as hops and destination device **MUST** support the Powerlevel Command Class to determine margin of all links between a source and a destination device. The highest power level is obtained when all links in route have a 6dB or better margin as assured during the installation process. The highest power level cannot be obtained in case a link in a route has less than 6dB margin or a device listed in a route does not support the Powerlevel Command Class.

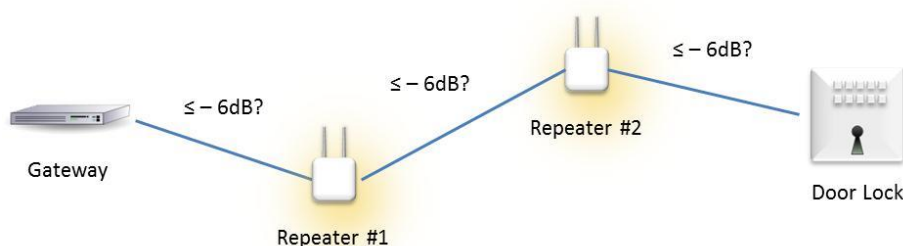


Figure 6, Determining margin of all links between source and destination device.

The Gateway executes the following steps to determine the highest power level reduction:

- Instruct Repeater #1 to determine highest link margin between Repeater #1 and gateway where it is possible to send NOP frames successfully.
- Instruct Repeater #2 to determine highest link margin between Repeater #2 and Repeater #1 where it is possible to send NOP frames successfully.
- Instruct Door Lock (or any other FLiRS based node) to determine highest link margin between Door Lock and Repeater #2 where it is possible to send NOP frames successfully.¹
- In case all devices support Powerlevel Command Class and the test was successful the LWRdB will be set to the highest reduction in power level where all links worked without failures.

Appendix A.4 Last Working Route RSSI measurement

LWR_RSSI = The difference between LWR RSSI and background RSSI

The LWR_RSSI is an estimation of the signal-to-noise ratio for each LWR in the network. LWR_RSSI is measured in dB.

Note: This measurement can only be obtained for routes comprised of repeaters and destination nodes based on SDK 6.60 or newer. For other routes, this measurement should be skipped. If the Gateway is based on an older SDK then RSSI measurements cannot be performed in the network.

¹ Note that power level test should always be done from the FLiRS node to the repeater to avoid that the FLiRS node powers down during the test.

The background RSSI, `RSSI_BACKGR` is an array of 2 or 3 values, describing the background RSSI for each channel. `RSSI_BACKGR` is measured at the Gateway. Use the following procedure to measure it:

1. Get 10 measurements of the background RSSI from the serialapi command `ZW_GetBackgroundRSSI`. At least 50 ms must pass between each measurement. Each call to `ZW_GetBackgroundRSSI` returns measurements for all channels. The following steps must be performed separately for each channel:
 - a. Discard the 5 highest measurements.
 - b. Calculate `RSSI_BACKGR` as the average of the remaining 5 measurements. For the purpose of averaging, replace `RSSI_BELOW_SENSITIVITY` with a value of -100 dBm. Replace `RSSI_MAX_POWER_SATURATED` with a value of 1 dBm.

To calculate the `LWR_RSSI` for a particular LWR follow this procedure:

1. Determine the destination node of the LWR to test
2. From the Gateway, issue a `ZW_SendData` with NOP payload and the `TRANSMIT_OPTION_ACK` (no other transmit options) to the destination. This transmit option will ensure that the LWR is used.
3. The `ZW_SendData` callback contains the channel `BACKChannelNo` used for the RSSI measurements and array of RSSI values called `RSSI_ARRAY` with one element per hop. The calculations are different for 2CH and 3CH systems. `ZW_GetBackgroundRSSI` will return 2 values for 2CH systems and 3 values for 3CH systems. Count the number of values returned by `ZW_GetBackgroundRSSI` to choose the right calculation method.

a. 2CH calculation

- i. Choose the `RSSI_BACKGR` corresponding to CH and call it `RSSI_BACKGR[CH]`.

- ii. Calculate `LWR_RSSI` for the i'th hop as

$$\text{LWR_RSSI}(i) = \text{RSSI_ARRAY}[i] - \text{RSSI_BACKGR}[CH]$$

Replace `RSSI_ARRAY[i]` values of `RSSI_MAX_POWER_SATURATED` with 1 dBm for the purpose of subtraction. Replace `RSSI_ARRAY[i]` values of `RSSI_BELOW_THRESHOLD` with -100dBm.

- iii. The `LWR_RSSI` for the entire route is the minimum of all `LWR_RSSI` values.

b. 3CH calculation.

- i. Choose `RSSI_BACKGR_MIN` as the minimum of `RSSI_BACKGR`

- ii. Calculate `LWR_RSSI` for the i'th hop as

$$\text{LWR_RSSI}(i) = \text{RSSI_ARRAY}[i] - \text{RSSI_BACKGR_MIN}$$

Replace `RSSI_ARRAY[i]` values of `RSSI_MAX_POWER_SATURATED` with 1 dBm for the purpose of subtraction. Replace `RSSI_ARRAY[i]` values of `RSSI_BELOW_THRESHOLD` with -100dBm.

- iii. The LWR_RSSI for the entire route is the minimum of the LWR_RSSI values for each hop in the route. In particular:

$$\text{LWR_RSSI}_{\text{route}} = \min (\text{LWR_RSSI}(0), \text{LWR_RSSI}(1), \dots \text{LWR_RSSI}(N))$$

where N is the number of hops in the route. N=0 for direct range, N=1 for one repeater etc.

Example 2CH calculation: Assume that RSSI_BACKGR is [-70, -66] dBm for ch0 and ch1, and that RSSI_ARRAY is [-40, RSSI_MAX_POWER_SATURATED, -33] and CH=ch1. Perform these calculations:

1. First we substitute RSSI_MAX_POWER_SATURATED to get a modified RSSI_ARRAY = [-40, 1, -33].
2. Then we pick RSSI_BACKGR[CH=ch1]=-66.
3. Calculate LWR_RSSI for hop 0:
 $\text{LWR_RSSI}(0) = -40 - (-66) = 26 \text{ dB}$
4. Calculate LWR_RSSI for hop 1:
 $\text{LWR_RSSI}(1) = 1 - (-66) = 67 \text{ dB}$
5. Calculate LWR_RSSI for hop 2 as
 $\text{LWR_RSSI}(2) = -33 - (-66) = 33 \text{ dB}$
6. The LWR_RSSI for the entire route is 26 dB because that is the minimum LWR_RSSI over all hops.

Example 3CH calculation: Assume that RSSI_BACKGR is [-80, RSSI_BELOW_THRESHOLD, -52] dBm for ch0, ch1 and ch2, and that RSSI_ARRAY is [-50, -30, -43]. Perform these calculations:

1. First we substitute RSSI_BELOW_THRESHOLD to get a modified RSSI_BACKGR = [-80, -100, -52].
2. Choose RSSI_BACKGR_MIN = min(-80, -100, -52) = -100.
3. Calculate LWR_RSSI for hop 0 as
 $\text{LWR_RSSI}(0) = -50 - (-100) = 50 \text{ dB}$
4. Calculate LWR_RSSI for hop 1 as
 $\text{LWR_RSSI}(1) = -30 - (-100) = 70 \text{ dB}$
5. Calculate LWR_RSSI for hop 2 as
 $\text{LWR_RSSI}(2) = -43 - (-100) = 57 \text{ dB}$
6. The LWR_RSSI for the entire route is 50 dB because that is the minimum LWR_RSSI value over all hops.

The LWR_RSSI values obtained for each LWR can be mapped to a network health value using Table 2, Network health scale, below.

Note: If Priority Routes are used, identical routes must be set in the Gateway and the node being tested.

Note: The values to substitute for RSSI_MAX_POWER_SATURATED and RSSI_BELOW_THRESHOLD depend on the RF hardware, in particular the sensitivity and dynamic range of the RSSI detector. The values given here are recommended for the 500-series chip.

Appendix A.5 Network Health Values

NHV = Network health value

The network health value is a number that indicates the performance and stability of a node in the system. The scale goes from 0 – Non working device to 10 – perfect working device.

NHS = Network Health Symbol

NHS is a mapping of NHV into simple symbols that each defines a specific set of actions that should be performed by the installer. The mapping of NHS and the actions taken is described in section 3.

Appendix A.6**Network health calculation**












NHV	NHS	Equation
10		PER = 0 AND RC = 0 AND NB > 2 AND LWRdB ≥ 6dB AND LWR_RSSI ≥ 17dB
9		PER = 0 AND RC = 1 AND NB > 2 AND LWRdB ≥ 6dB AND LWR_RSSI ≥ 17dB
8		PER = 0 AND RC ≤ 1 AND NB ≤ 2 AND LWRdB ≥ 6dB AND LWR_RSSI ≥ 17dB
7		PER = 0 AND RC ≤ 1 AND NB > 2 AND (LWRdB < 6dB OR LWR_RSSI < 17dB)
6		PER = 0 AND RC ≤ 1 AND NB ≤ 2 AND (LWRdB < 6dB OR LWR_RSSI < 17dB)
5		PER = 0 AND 1 < RC ≤ 4
4		PER = 0 AND RC > 4
3		PER = 1
2		PER = 2
1		2 < PER < 10
0		PER = 10

Table 2, Network health scale

APPENDIX B SOFTWARE REQUIREMENTS

In order to support the full functionality of this specification the slave and controller devices in the network must support some specific Z-Wave functionality.

Appendix B.1 Slave devices

The slave devices in the network must support the Powerlevel Command Class

Appendix B.2 Serial API static controller enhancements

The following enhancements have been done to the serialAPI to help the host in implementing the IMA functionality in the SerialAPI_Controller_Static_IMA (DevKit 6.51+). For DevKit 6.6x; all controller based SerialAPI targets supports the IMA features.

SerialAPI startup command

The Started Command is send by the Z-Wave module when it powers up. The command is used by the host processor to see when the Z-Wave module is ready after a hard or soft reset or that it has been reset by the watchdog.

For DevKit 6.51+

7	6	5	4	3	2	1	0
FUNC_ID_SERIALAPI_STARTED = 0x0A							
StartupReason							

For DevKit 6.6x+

7	6	5	4	3	2	1	0
FUNC_ID_SERIALAPI_STARTED = 0x0A							
StartupReason							
bWatchdogStarted							
deviceOptionMask							
nodeType_generic							
nodeType_specific							
cmdClassLength							
cmdClass[cmdClassLength]							

StartupReason (8 bit):

ZW_WAKEUP_RESET – Started up because of reset, power cycle, external interrupt.or watchdog

ZW_WAKEUP_WUT – Started up because the WUT timed out

ZW_WAKEUP_SENSOR – Started up because a wakeup beam was received, See also [1] or [3].

StartupReason define	Value
ZW_WAKEUP_RESET	0x00
ZW_WAKEUP_WUT	0x01
ZW_WAKEUP_SENSOR	0x02
ZW_WAKEUP_WATCHDOG	0x03
ZW_WAKEUP_EXT_INT	0x04

ZW_WAKEUP_POR	0x05
---------------	------

WatchdogStarted (8 bit):

0 – Watchdog has not been started on the Z-Wave SerialAPI module.

1 – Watchdog has been started and is serviced on the Z-Wave SerialAPI module.

deviceOptionMask (8 bit):

Current device option setting as set by the HOST.

nodeType_generic (8 bit):

Current Generic Device Type defined for the Z-Wave SerialAPI controller static node as set by the HOST.

nodeType_specific (8 bit):

Current Generic Device Type defined for the Z-Wave SerialAPI controller static node as set by the HOST.

cmdClassLength (8 bit):

Parameter indicating the number of Command Classes descriptors following this parameter.

cmdClass[cmdClassLength] (8 bit * cmdClassLength):

The Command Classes currently supported by the HOST as set by the HOST.

SerialAPI ZW_SendData() transmit time measurements

For DevKit 6.51+ controller nodes the ZW_SendData() implementation in the serialAPI has been enhanced with a transmit time measurement. The serialAPI returns the time the transmission took. The time is measured from the ZW_SendData() call is made to the protocol and until the protocol returns the callback. The returned time is a 16 bit value in steps of 10ms where MSB is send first.

HOST->ZW: REQ | 0x13 | nodeID | dataLength | pData[] | txOptions | funcID

ZW->HOST: RES | 0x13 | RetVal

ZW->HOST: REQ | 0x13 | funcID | txStatus | txTime

For DevKit 6.6x+ controller nodes the callback parameters has been extended and a description of the ZW_SendData SerialAPI protocol flow can be found in [3].

SerialAPI lock on ZW_AddNodeToNetwork() and ZW_RemoveNodeFromNetwork()

The SerialAPI implementation of ZW_AddNodeToNetwork() and ZW_RemoveNodeFromNetwork() will now ignore any calls made to the two functions if the protocol is in a state where add or remove is already in progress and it is not allowed to call the two functions. The host implementation of ZW_AddNodeToNetwork() and ZW_RemoveNodeToNetwork() should follow the flow chart in the application programmers guide to ensure that error and exception handling is done correctly.

For debug purpose only a command that removes the lock on ZW_AddNodeToNetwork() and ZW_RemoveNodeFromNetwork() has been made. This command should NOT be used in production code and is only implemented for test purposes. The command has the function id FUNC_ID_PROPRIETARY_1.

APPENDIX C TEST TIME

The network health and the rediscovery process sends a lot of frames between nodes so running the different tests and recovery steps will take time. The time a test takes depends on the number of devices in the network, the size of the network, the number of FLiRS nodes and also the result of the test.

This section describes some formulas that can be used for calculating how long it takes to run the different tests and also gives some examples of network topologies and test times.

It is not possible to calculate the exact time a test will take because of the random back off timing in the protocol and the delay introduced by the application running the test but the calculation below will give a good approximation of the time.

Appendix C.1 Network Health test time

For network health the test time depends on what the outcome of the test is. In some situations the network health algorithm will do power level testing and in some it won't. So for timing calculation we calculate how long a test will take if it gives a green NHS and if it gives a red NHS.

For network health there is a number of factors that has an impact on how long the test takes. The factors are the following:

No_{Hops} Number of hops in LWR

The formula for calculating the test time is then:

Green result ($NHV > 7$)

$$T_{TestFLiRS} = 6 * ((2*60ms) + (1170ms*10) + (12*60ms)*(No_{Hops}))$$

$$T_{TestRep} = 6 * (12*60ms)*(No_{Hops} + 1)$$

Red result ($NHV = 0$, last link in LWR fails)

$$T_{TestFLiRS} = 6 * ((2*60ms) + (1200ms*10*3) + (12*60ms)*(No_{Hops}))$$

$$T_{TestRep} = 6 * ((2*60ms) + (10*630ms) + (12*60ms)*(No_{Hops}))$$

Appendix C.2 Rediscovery time

For rediscovery there are four factors that have an impact on how long the rediscovery takes. The factors are:

No_{FLiRS}	Number of FLiRS nodes in the network
No_{Reps}	Number of repeaters in the network
Nb_{FLiRS}	Number of FLiRS nodes in range of the tested node
Nb_{Reps}	Number of repeaters in range of the tested node

The formula for calculating how long it takes to do a rediscovery of a single node in the network is then:

$$T_{RediscoveryFLiRS} = (1100ms * 4) + (60ms * 4) + ((Nb_{Reps} + 1) * 60ms) + (Nb_{FLiRS} * 1100ms) + ((No_{Reps} - Nb_{Reps}) * 250ms) + ((No_{FLiRS} - Nb_{FLiRS} - 1) * 1200ms)$$

$$T_{RediscoveryRep} = (60ms * 4) + (60ms * 4) + ((Nb_{Reps} + 1) * 60ms) + (Nb_{FLiRS} * 1100ms) + ((No_{Reps} - Nb_{Reps} - 1) * 250ms) + ((No_{FLiRS} - Nb_{FLiRS}) * 1200ms)$$

And then that will give a formula for a full rediscovery that looks like this:

$$T_{NetRediscovery} = (No_{FLiRS} * T_{RediscoveryFLiRS}) + (No_{Reps} * T_{RediscoveryRep})$$

Appendix C.3 Timing examples

Timing example A

Network with 6 devices. One static controller, 2 FLiRS devices and 3 repeater devices. All nodes in the network are in direct range of the controller. All nodes can see each other in direct range.

NodeID 1 – Static controller

NodeID 2 – FLiRS device

NodeID 3 - FLiRS device

NodeID 4 – Repeater device

NodeID 5 – Repeater device

NodeID 6 – Repeater device

Network health time (all Green)

Node 2, FLiRS, $No_{Hops} = 0$

Calculated = 70920

Node 3, FLiRS, $No_{Hops} = 0$

Calculated = 70920

Node 4, Repeater, $No_{Hops} = 0$

Calculated = 4320

Node 5, Repeater, $No_{Hops} = 0$

Calculated = 4320

Node 6, Repeater, $No_{Hops} = 0$

Calculated = 4320

Total network health test time Green: Calculated = 154800ms

Network health time (one Red)

Node 2, FLiRS, $No_{Hops} = 0$, Red

Calculated = 216720ms

Total network health test time Red: Calculated = 300600ms

Rediscovery time

Node 2 – Calculated = 5980ms

Node 3 – Calculated = 5980ms

Node 4 – Calculated = 2860ms

Node 5 – Calculated = 2860ms

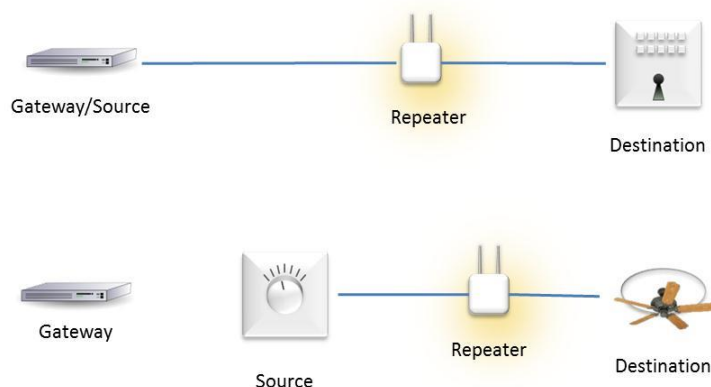
Node 6 – Calculated = 2860ms

Total rediscovery time: Calculated = 20540ms

APPENDIX D ADVANCED REPEATER INSTALLATION

The Powerlevel Command Class [2] enables placement of repeaters in the optimal position obtaining similar RF communication links between source/repeater and repeater/destination device.

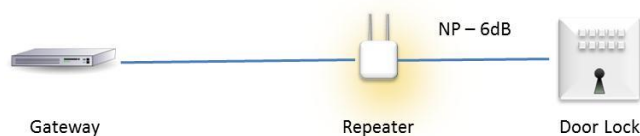
The Powerlevel Command Class is used for testing how much the transmit power can be lowered without breaking the communication between 2 devices. The command class must be supported by all the devices (repeater and destination) and device (gateway or source) initiating the test must be able to control it.



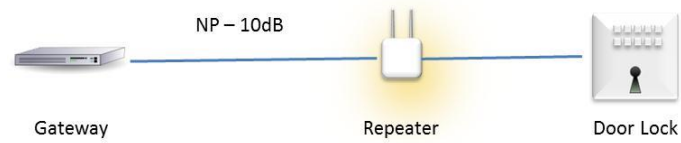
The following steps are done when finding the optimal position for a repeater.

NOTE: The protocol already uses -6dB compared to normal power (NP) to reduce range with 1/3 when finding neighbors so the protocol already ensures that there is a good margin in the routing table.

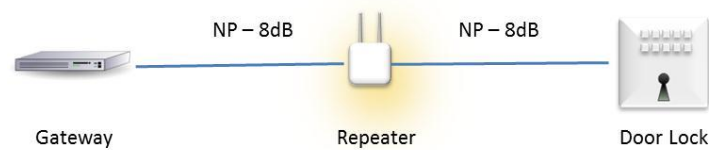
- 1 Place the repeater in the preferred position and include it to the network.
- 2 Test that the gateway (or source) can communicate with the destination device after the repeater has been included.
- 3 Gateway (or source) instruct destination device using Powerlevel Test Node Set/Get command to test link between destination and repeater. Start with the lowest possible transmit power and keep decreasing the transmit power reduction until communication works without failures. Transmit a NOP frame 10 times for a given transmit power.



- 4 Gateway (or source) instruct repeater device using Powerlevel Test Node Set/Get command to test link between repeater and gateway (or source). Starts with the highest reduction of transmit power and keep lowering the reduction until communication works without failures. Transmit a NOP frame 10 times for a given transmit power.



- 5 If the repeater is not in the wanted optimal position move the repeater and repeat the power level test until the repeater is in the optimal position



In mesh networks containing many repeaters and multiple routes to destinations it is not possible for the application to determine what route the protocol is going to use when communicating with a specific device. So using the Powerlevel Command Class in such a network becomes very time consuming and the result is very hard to use for diagnostics of the network.

If an application wants links with a better margin than -6dB in the mesh then the API call `ZW_RFPowerlevelRediscoverySet() [1]` can be called in the devices to reduce the output power further when finding neighbors.

APPENDIX E NETWORK HEALTH TOOL

The Network Health algorithm including source code and windows executable is contained in "IMAtoolbox.exe" file.

The sample application is implemented as a PC console application with a command line interface. The application is written in C language making it easier to adapt it to a new environment.

The Network Health functionality is primary contained in the "networkManagement" module. Network Health is determined by calling NetworkManagement_NetworkHealthStart, which together with dumping the result to screen and file also delivers the result in the sNetworkManagement structure returned when calling the start function.

Some data structures are needed to be defined in the HOST application and delivered when calling the NetworkManagement_NetworkHealthStart function; a BYTE array containing the nodes under test and the number of existing repeater in network.

The PowerLevel functionality is primary contained in the "PowerLevelTest" and "PowerLevelMan" modules. The test is initialized by calling PowerLevelTest_CPowerLevelTest() and PowerLevelTest_Init(). To start the actual test call PowerLevelTest_Set() that returns the result in the given callback function. An additional functionality that verifies a repeater placement is initialized and started by calling PowerLevelTestMan_CPowerLevelTestMan() and the result is returned in the given callback function.

The tests are started in main() in "IMAtoolbox.cpp" and from there the appropriate functions are called in the test specific modules.

A list of nodes that should be tested is created in ReloadNodeList in "IMAtoolbox.cpp" and in that function we also determine if a node is a "Listening" node or a repeater node.

A timing module "timing" using windows timing functionality is used for time sampling. Minimum resolution should be milliseconds.

Start the program from a command prompt by writing IMAtoolbox [comport] where com port is the number of the com port where the static controller is connected.

The user interface is character based and to get a list of supported commands press <Enter>. Below is a short list of the most important commands:

- A** : Add Node (only supported by Controllers)
- R** : Remove Node (only supported by Controllers)
- F** : remove Failed current nodeID 002
- U** : ZW_RequestNodeNeighborUpdate
- N** : Network Health Test Menu
- #** : Set current nodeID
- .** : Call ZW_GetNodeProtocolInfo for current nodeID
- "** : Call ZW_RequestNodeInfo for current nodeID 002
- *** : ZW_SendData with explorer to current nodeID (002)

%	: Dump NodeInfo for current nodeID 002
\	: Broadcast Node Information Frame
P	: Get serialapi capabilities
V	: Get version
I	: Initialize nodelist and dump nodeinfo (only supported by Controllers)
L	: ZW_SetLearnMode
D	: ZW_SetDefault
W	: ZW_SoftReset
!	: Print incoming frames - TRUE
J	: Toggle serial logging to screen – 0
Q	: Quit

The 'Network Health Test Menu' menu contains a number of tests used in the IMA software. The following commands are present in the 'Network Health Test Menu' menu:

1 - Z-Wave Network Health – Full network

Execute Network Health on all eligible nodes in network, logs result in mainlog and a timestamped NetworkHealth logfile

2 - Z-Wave Network Health – current NodeID 002

3 - Z-Wave Network Health – Maintenance Mode

4 – Ping all nodes

5 - Transmit frame to current NodeID 002 using Maintenance ZW_SendData

Transmit a BASIC SET frame to 'current NodeID' using Network Health provided ZW_SendData wrapper function, which uses the transmit to update the nodes logged transmit metrics for use in the Network Health Number functionality.

6 - Rediscovery of known Listening nodes

Does a full Rediscovery of known nodes.

7 - Check if Repeater is a valid repeater to Node

Test that a repeater node is a valid repeater for a destination node.

8 - Dump node neighbors

Dumps the nodes neighbors.

9 – Get Version from all nodes

Uses Version Command Class to query nodes under test for Z-Wave and Application version.

P - Power level test

Power test menu. The 'Power level test' menu supports the Powerlevel Command Class [2].

S - Set Network Health Maintenance Sample Period (Default if = 0) – 0

Set the Maintenance sample period – how many seconds between every Maintenance sample transmit. If 0 (default) is selected then the sample period will be set to 60minutes/NumberOfNodeUnderTest (in seconds).

R - Set Maintenance rounds before Rediscovery list is executed – 5

Set the number of full Maintenance rounds before the accumulated Rediscovery list is executed. A Maintenance round is when all nodes have been ‘sampled’ and have their NH updated. When the Rediscovery list is executed all nodes which are flagged as needing a rediscovery is rediscovered.

- Enter new current NodeID 002**I – Inject LWR for current NodeID 002**

manually set the LWR for current NodeID using ZW_SetLastWorkingRoute (only Devkit 6.51+). For Devkit 6.6x+ ZW_SetLastWorkingRoute has been obsoleted by ZW_SetPriorityRoute and when used sets an Application Priority Route (APR) that following always will be tried prior to any possible existing LWR and will only be removed if the Application calls ZW_SetPriorityRoute again.

T – get TX Timers

Get current number of milliseconds the Z-Wave RF has been in TX (“RF in TX” Timers) on the 3 channels possible. Which channels are valid are region specific (only Devkit 6.51+).

C – Clear TX Timers

Clear (Reset to ZERO) the “RF in TX” Timers (only Devkit 6.51+).

L – Toggle LWR Lock OFF

Toggle the LWR Lock ON/OFF using the ZW_LockRoute, if ON then existing LWR are not removed (DevKit 6.6x+ or exiled) if failed and are only changed if a successful Explore frame communication occurs (if Explore TX option is selected) or a Inject using ZW_SetLastWorkingRoute are done.

M – Measure and show RSSI Map

This will measure and show a controller-centric map of the LWRs from controller to each node in the network. The RSSI level of each hop of each route is printed. The background RSSI level is also sampled, averaged and displayed.

B – Show background RSSI

Print a single sample of the background RSSI for all channels.

0 - Exit Network Health Test Menu

REFERENCES

- [1] SD, INS12308, Instruction, Z-Wave 500 Series Appl. Prg. Guide v6.50.01 BETA.
- [2] SD, SDS12652, Software Design Specification, Z-Wave Command Class Specification N-Z.
- [3] SD, INS13044, Instruction, Z-Wave 500 Series Appl. Programmers Guide v6.60.00

INDEX

L

LNK.....	18
----------	----

N

NB.....	17
Network health.....	17
Network Health Symbol.....	21
Network health value.....	21
NHS.....	21
NHV.....	21
Number of failed transmissions.....	17
Number of repeater neighbors.....	17
Number of times the protocol needed additional routes.....	17

P

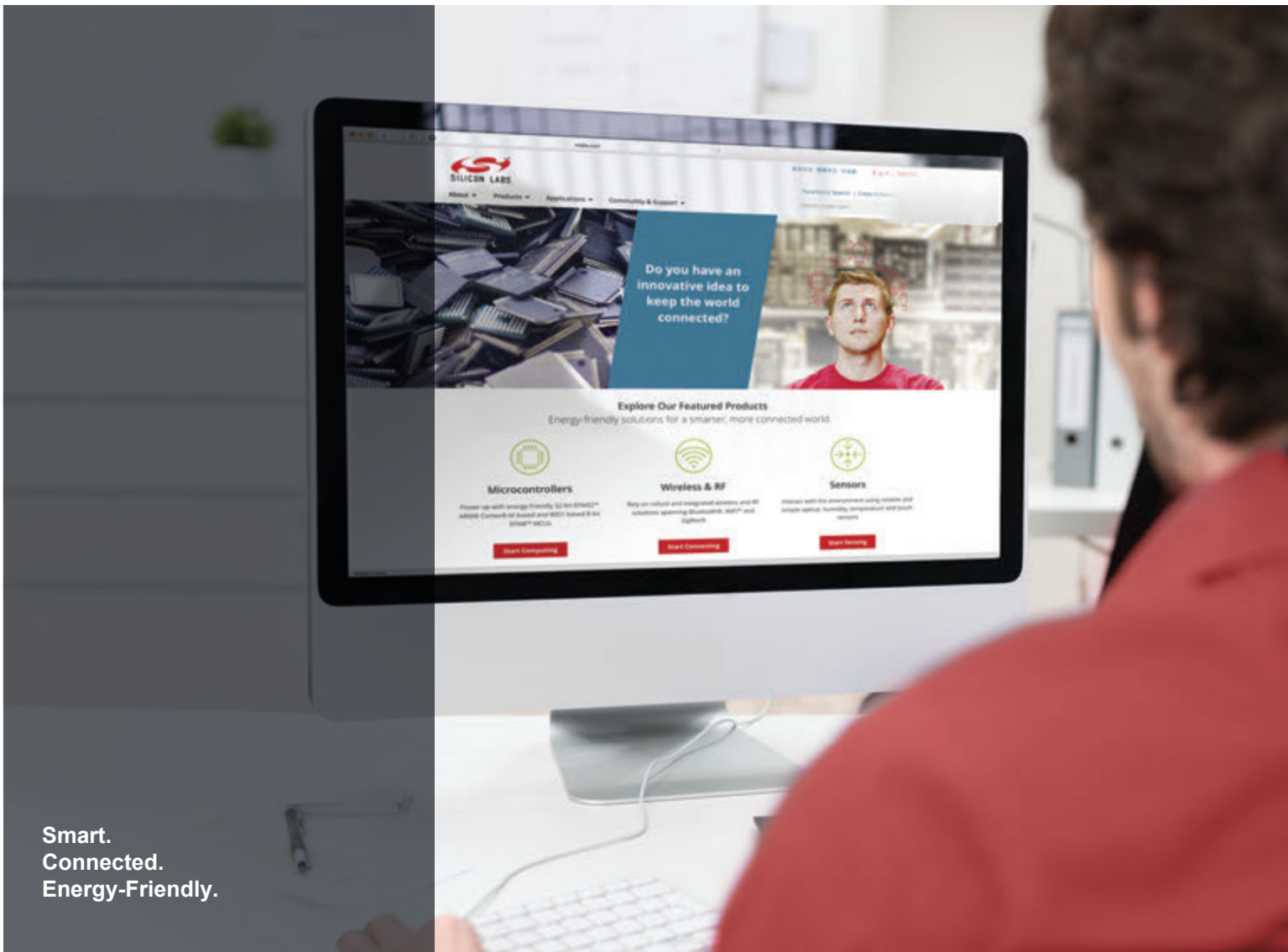
PER.....	17
Powerlevel Command Class.....	31

R

RC.....	17
RF communication link margin.....	11

T

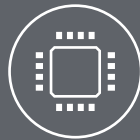
T.....	17
T _{app}	17



Smart.
Connected.
Energy-Friendly.



Products
www.silabs.com/products



Quality
www.silabs.com/quality



Support and Community
community.silabs.com

Disclaimer

Silicon Labs intends to provide customers with the latest, accurate, and in-depth documentation of all peripherals and modules available for system and software implementers using or intending to use the Silicon Labs products. Characterization data, available modules and peripherals, memory sizes and memory addresses refer to each specific device, and "Typical" parameters provided can and do vary in different applications. Application examples described herein are for illustrative purposes only. Silicon Labs reserves the right to make changes without further notice and limitation to product information, specifications, and descriptions herein, and does not give warranties as to the accuracy or completeness of the included information. Silicon Labs shall have no liability for the consequences of use of the information supplied herein. This document does not imply or express copyright licenses granted hereunder to design or fabricate any integrated circuits. The products are not designed or authorized to be used within any Life Support System without the specific written consent of Silicon Labs. A "Life Support System" is any product or system intended to support or sustain life and/or health, which, if it fails, can be reasonably expected to result in significant personal injury or death. Silicon Labs products are not designed or authorized for military applications. Silicon Labs products shall under no circumstances be used in weapons of mass destruction including (but not limited to) nuclear, biological or chemical weapons, or missiles capable of delivering such weapons.

Trademark Information

Silicon Laboratories Inc.®, Silicon Laboratories®, Silicon Labs®, SiLabs® and the Silicon Labs logo®, Bluegiga®, Bluegiga Logo®, Clockbuilder®, CMEMS®, DSPLL®, EFM®, EFM32®, EFR®, Ember®, Energy Micro, Energy Micro logo and combinations thereof, "the world's most energy friendly microcontrollers", Ember®, EZLink®, EZRadio®, EZRadioPRO®, Gecko®, ISOmodem®, Micrium, Precision32®, ProSLIC®, Simplicity Studio®, SiPHY®, Telegesis, the Telegesis Logo®, USBXpress®, Zentri, Z-Wave and others are trademarks or registered trademarks of Silicon Labs. ARM, CORTEX, Cortex-M3 and THUMB are trademarks or registered trademarks of ARM Holdings. Keil is a registered trademark of ARM Limited. All other products or brand names mentioned herein are trademarks of their respective holders.



Silicon Laboratories Inc.
400 West Cesar Chavez
Austin, TX 78701
USA

<http://www.silabs.com>