



Instruction

Development Controller User Guide

Document No.:	INS10236
Version:	5
Description:	-
Written By:	JFR;PSH
Date:	2010-04-19
Reviewed By:	JKA;CHL
Restrictions:	Public

Approved by:

Date	CET	Initials	Name	Justification
2010-04-19	10:35:31	NTJ	Niels Thybo Johansen	

This document is the property of Zensys A/S. 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. Rev	Date	By	Pages affected	Brief description of changes
1	20050224	HEH	Section 3.1.2	Added jumper settings for ZM21xx modules
1	20050318	PSH	Section 4.4	Added description of group on/off abort
1	20050404	PSH	Section 4.2.2	Added information about group replication
2	20050801	JFR	Section 3.1.2	Jumper settings when loading the external EEPROM with a new home ID on a ZW0201 based module
3	20060105	MVO	All	New 1 st page/header/footer contents. New Doc No
4	20070228	JFR	Section 3 & 4.2.1	ZW0301 based modules added
4	20090119	PSH	Section 4.2	Added description of network wide inclusion mode at power up.
5	20100223	JFR	Section 4	Press the pushbutton 3 times within 1,5 sec. on the Z-Wave Node to transmit node information frame.

Table of Contents

1	ABBREVIATIONS.....	1
2	INTRODUCTION.....	1
2.1	Purpose	1
2.2	Audience and prerequisites.....	1
3	GETTING STARTED	2
3.1	Jumper settings	4
3.1.1	Setup when using ZM12xx Modules.....	4
3.1.2	Setup when using ZM21xx/ZM31xx Modules.....	5
3.2	LED0 indication	6
3.3	LED1 indication	6
4	SAMPLE APPLICATION DESCRIPTION	7
4.1	Features of the Development Controller sample application	7
4.2	Include a Node to the Z-Wave Network	8
4.2.1	Slave Node	8
4.2.2	Controller Node	9
4.2.3	Network wide inclusion	9
4.3	Associate a Slave Node to a Group.....	10
4.4	Switching a Group of Nodes On and Off.....	10
4.4.1	Aborting a Group On or Off command	10
4.5	Dimming a Group of Nodes.....	10
4.6	Assigning a Route to a Routing Slave.....	11
4.7	Excluding a Node from the Z-Wave Network (reset Node).....	12
4.8	Resetting the ZW0102 Development Controller.....	12
4.9	Using Automatic Network Update Feature of Controllers	13
4.10	Using Inclusion Controllers.....	14
5	FUNCTION OVERVIEW	15
6	REFERENCES	16

List of Figures

Figure 1.	ZW0x0x Development Controller Unit.....	2
Figure 2.	User Interface for ZW0102 Development Controller	3
Figure 3.	Jumper Settings when using ZM12xx Modules.....	4
Figure 4.	Jumper Settings when using ZM21xx/ZM31xx Modules.....	5
Figure 5.	Position of LED0.....	6
Figure 6.	Position of LED1.....	6
Figure 7.	ZW0102 Controller/Slave Unit.....	8

1 ABBREVIATIONS

Abbreviation	Explanation
SIS	SUC ID Server
SUC	Static Update Controller
NWI	Network Wide Inclusion

2 INTRODUCTION

2.1 Purpose

The purpose of this document is to describe the functionalities and features of the ZW0x0x Development Controller sample application.

2.2 Audience and prerequisites

The audience of this document is Z-Wave partners and Zensys. It is assumed that the Z-Wave partner already is familiar with the Z-Wave Developer's Kit.

3 GETTING STARTED

The ZW0x0x Development Controller sample application is designed for the ZW0x0x Development Controller Unit, which is an assembly of the ZW010x Development Module [2] and a ZW0102/ZW0201/ZW0301 based Z-Wave Module.

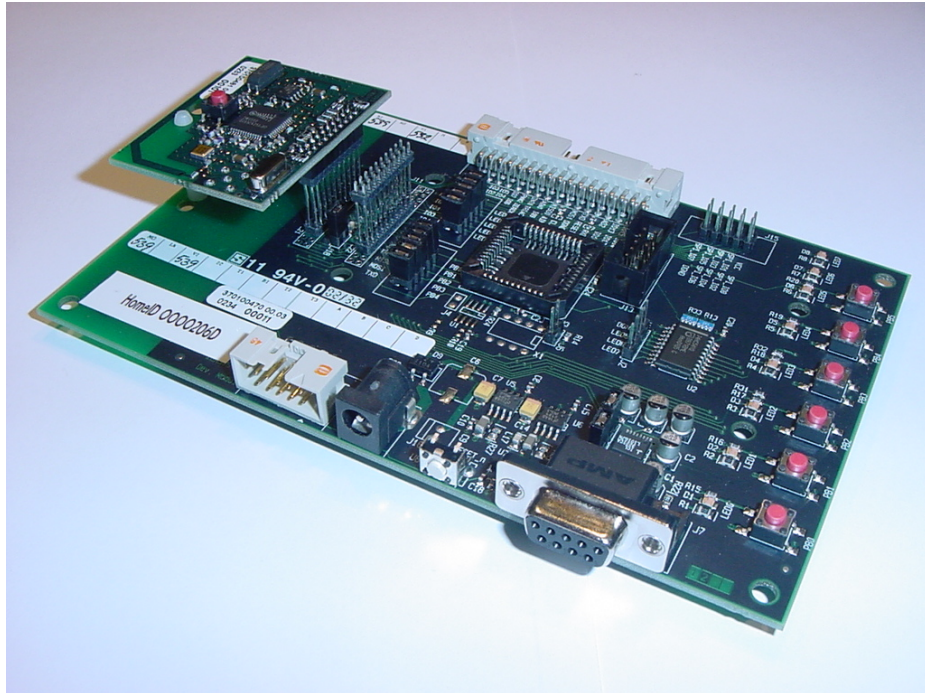


Figure 1. ZW0x0x Development Controller Unit

The application user interface is based on 5 pushbuttons (named PB0, PB1, PB2, PB3 and PB4) and 2 LED's (named LED0 and LED1) on the ZW010x Development Module.

Note: The ZW010x Development Module is equipped with 6 pushbuttons and 8 LED's, but only 5 pushbuttons and 2 LED's are used in this version of the ZW0102 Development Controller sample application – as depicted in Figure 2 below.

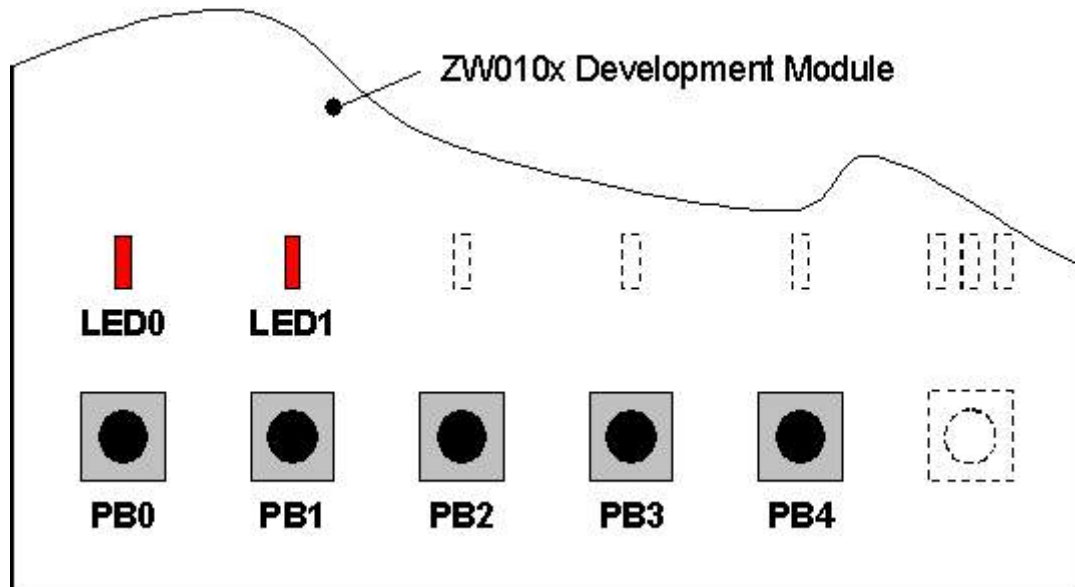


Figure 2. User Interface for ZW0102 Development Controller

The two LED's are the users only indication of the status of the Development Controller and the various states of those two LED's will be described in the following.

3.1 Jumper settings

In order to run the Development Controller sample application on the ZW010x Development Module some jumpers have to be shorted to connect the Z-Wave Module with Push Button PB[4:0] and LED[1:0] on the Development Module.

3.1.1 Setup when using ZM12xx Modules.

Seven jumpers have to be implemented on J1 and J5 as shown in the figure below.

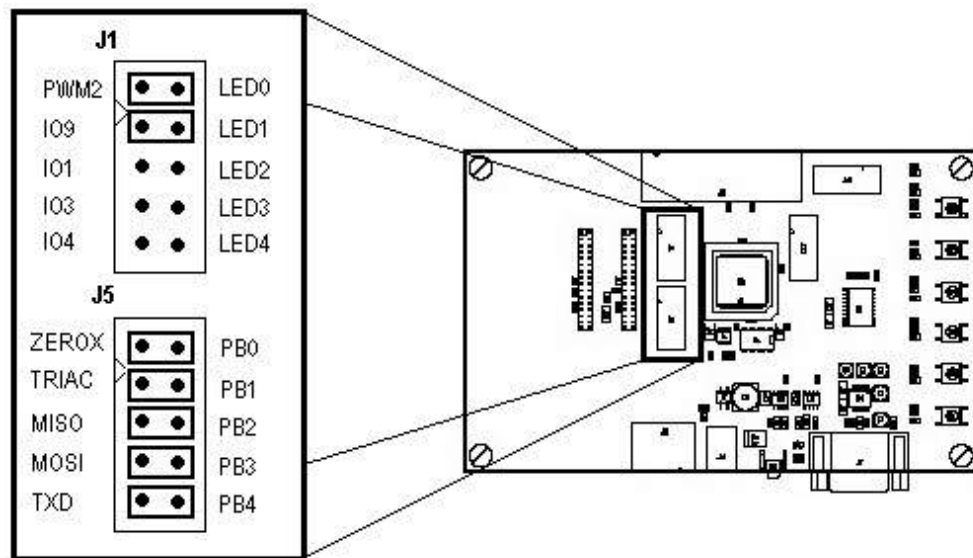


Figure 3. Jumper Settings when using ZM12xx Modules

One example of the ZM12xx family of modules could be the ZM1220 Z-Wave Module [3].

Note: Make sure that no other external circuitry connected to the Development Module is driving the signals used

3.1.2 Setup when using ZM21xx/ZM31xx Modules.

LED1 is controlled by RXD when using ZM21xx modules. Because of this RXD (J9) to LED1 (J1) must be connected as indicated by the grey line.

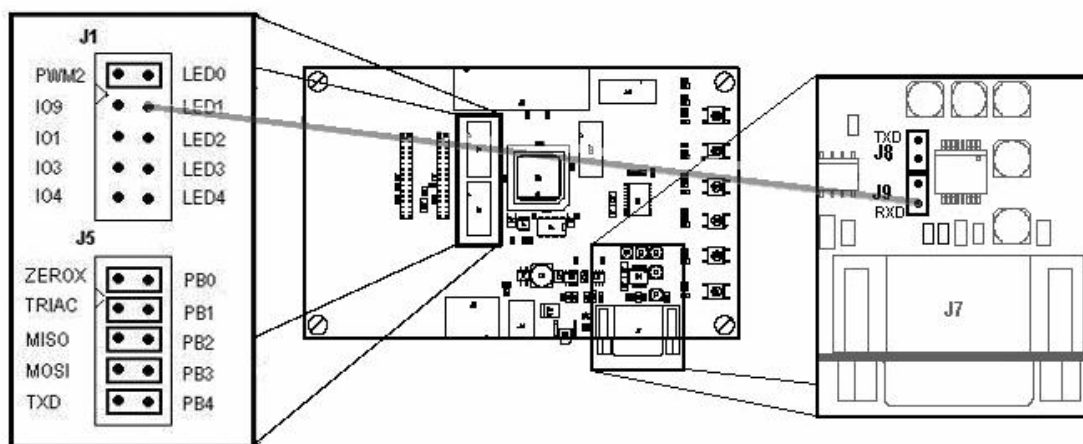


Figure 4. Jumper Settings when using ZM21xx/ZM31xx Modules

Remember to remove the cable shown on the figure above and reinstall jumper J9 in case a new home ID is to be loaded into the external EEPROM.

One example of the ZM21xx family of modules could be the ZM2120 Z-Wave Module [6]. With respect to the ZM31xx family of modules could it be the ZM3120 Z-Wave Module [7].

Note: Make sure that no other external circuitry connected to the Development Module is driving the signals used.

3.2 LED0 indication

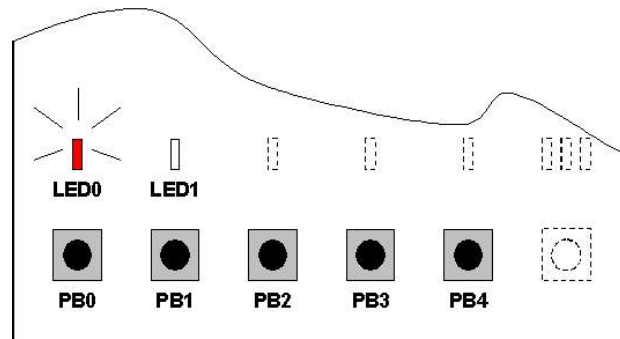


Figure 5. Position of LED0

LED0 indicate the following states:

ON	Waiting for Z-Wave frame from another Node
Flashing	Controller is processing
OFF	Controller is idle

3.3 LED1 indication

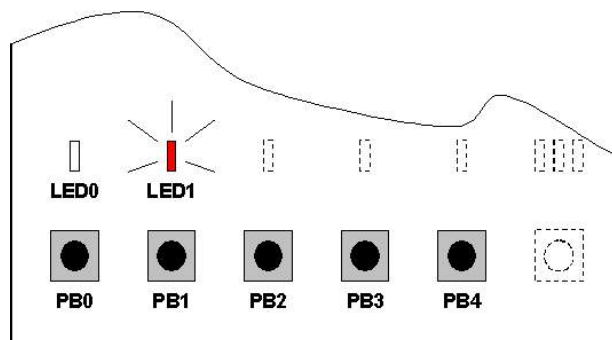


Figure 6. Position of LED1

LED1 indicate the following states:

ON	<ul style="list-style-type: none"> System error – errors can be reset by pushing any of the push buttons Assigning routes to a binary sensor: LED1 will go ON when awaiting signal (Z-Wave frame) from the Binary sensor When resetting the Controller: Reset in progress <p>Please refer to the descriptions of the various system functionalities in the following chapter.</p>
Flashing	Not used
OFF	System is ready

4 SAMPLE APPLICATION DESCRIPTION

4.1 Features of the Development Controller sample application

The ZW0102 Development Controller sample application shows the following features of the Z-Wave protocol:

1. Include Slave and Controller Nodes to the Z-Wave Network
2. Associate a Node to a Group
3. Switching a Group of Nodes on or off (toggle function)
4. Dimming a Group of multilevel switches (e.g. LED Dimmers)
5. Assigning a Route to a routing slave (e.g. a Binary Sensor)
6. Excluding a Node from the Z-Wave Network
7. Resetting the ZW0102 Development Controller
8. Requesting network updates to a Secondary Controller from a Static Update Controller (SUC) in case it is present in the network
9. Work as Inclusion Controller when a SUC ID Server (SIS) is present in the network
10. Request network wide inclusion in a Z-Wave network

In the following sections is each feature described in details.

4.2 Include a Node to the Z-Wave Network

Adding a slave node and a controller node is done in a generic way on Z-Wave protocol level, but the user interface can differ depending on the hardware and software implementation.

When building a new Z-Wave Network using the Development Controller sample application, please make sure that all units are reset in order to avoid problems with duplicate Node ID's when including Nodes to the network. Please refer to section 4.7 for information on how to reset nodes or section 4.8 for information on how to reset the Development Controller.

4.2.1 Slave Node

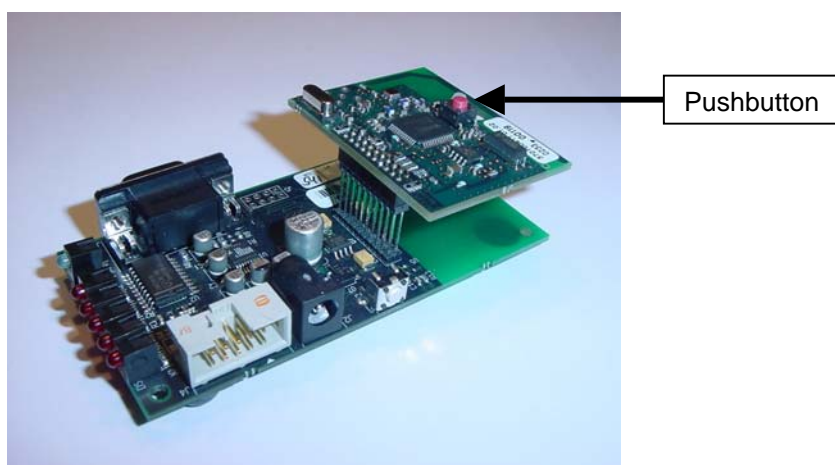
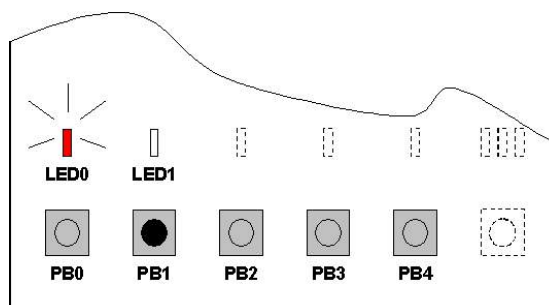


Figure 7. ZW0102 Controller/Slave Unit

The slave applications Led Dimmer, Binary Sensor or Binary Sensor Battery is designed for the Controller/Slave Unit, which is an assembly of the of the ZW010x Interface Module [4] and a ZW0102/ZW0201/ZW0301 based Z-Wave Module. To include this type of module to the Z-Wave Network when using a Development Controller as primary requires the following steps:

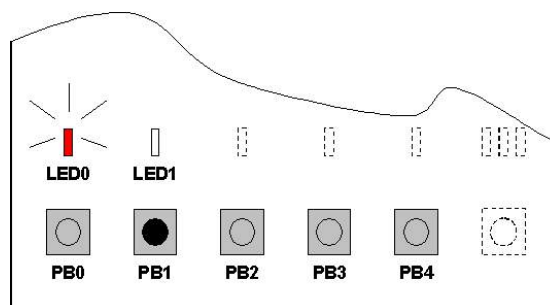
1. On the primary Development Controller, press and hold PB1 – this will turn LED0 ON, indicating that the system is awaiting node information from the Node which is being included to the Network
2. Press the pushbutton 3 times within 1,5 sec. on the Z-Wave slave Node being included in the network. The pushbutton (red) is located on the ZM1220 Z-Wave Module.
3. When LED0 starts to flash on the primary Development Controller, PB1 can be released. When the node information has been received and processed by the Development Controller, LED0 will turn OFF, which means that the Node has now been included in the network.
4. Repeat step 1 and 3 to include more Nodes in the Network.



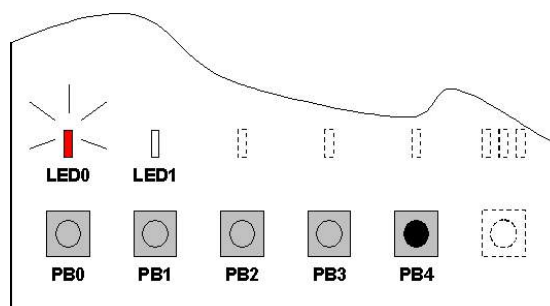
4.2.2 Controller Node

The controller application Development Controller is provided on the ZM1220 Z-Wave Module connected to the ZW010x Development Module. To include this type of module to the Z-Wave Network when using a Development Controller as primary requires the following steps:

1. On the primary Development Controller, press and hold PB1 – this will turn LED0 ON, indicating that the system is awaiting node information from the Node which is being included to the Network



2. Press and hold PB4 on the Development Controller that you want to include into the network as a secondary controller. Remember to keep the PB4 pressed until LED0 lights up. If LED1 lights up the replication failed and should be repeated!



3. When the node information has been received and processed by the Development Controller, LED0 will turn OFF, which means that the Node has now been included in the network and PB1 can be released.
4. Repeat step 1 and 3 to include more Nodes in the Network.

Please note that the process of replication does include transfer of information about which nodes are included in the Group – So the new controller will be able to control the same group as the controller used to include it.

When additional nodes are included to the network then the secondary Development Controller's network topology is incomplete in case a static update controller (SUC) is not present. By repeating the inclusion process on the secondary Development Controller it will receive the updated network topology. Regarding automatic network updates refer to section 4.9.

4.2.3 Network wide inclusion

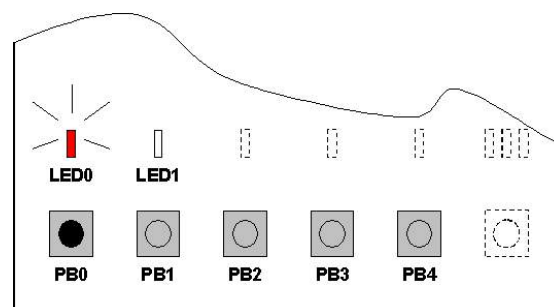
By default, the controller will enter network wide inclusion (NWI) when it is powered up and have not already been included or have included other nodes itself. The controller will stay in NWI mode for 4 minutes or until it has been included into the network. Any key press will terminate the NWI mode and the only way to make the controller enter NWI mode again is by doing hardware reset by either remove and reapply the power or press the reset button on the side of the board.

4.3 Associate a Slave Node to a Group.

The Development Controller has only one Group that Nodes can be associated to – this Group is controlled via pushbutton PB0.

When associating Nodes to the Group, please follow the description below:

1. Make sure that the Node which you want to associate to the Group is already included in the Z-Wave network
2. Press and hold PB0 on the Development Controller – this will turn LED0 ON, indicating that the system is awaiting node information from the Node you want to associate.



3. Press on the pushbutton 3 times within 1,5 sec. on the Z-Wave module mounted on the Node being associated. This will prompt the Z-Wave Module to send out a node information frame.
4. When the node information has been received and processed by the Development Controller, LED0 is flashing. The Node has now been associated to the Group. LED0 will turn OFF when PB0 is released.
5. Repeat step 1 to 4 if more Nodes should be associated to the Group.

To remove an association from the group hold down PB0 and press shortly on the pushbutton located at the Z-Wave module to be removed. The node is still a part of the Z-Wave network.

4.4 Switching a Group of Nodes On and Off.

The switching On and Off of the Development Controller Group is done by pressing PB0 shortly. When PB0 is pressed, LED0 turns briefly ON and turns OFF again, indicating that the processor is processing the signal.

4.4.1 Aborting a Group On or Off command

When a group On or Off command is issued the development controller will start to send commands to each node in the group. If that process wants to be aborted then the PB0 button can be pressed again to abort the commands to the remaining nodes.

4.5 Dimming a Group of Nodes

The dimming of a Group of Nodes is done by pressing, and holding PB0. While dimming is ongoing, LED0 will be ON indicating that the processor is busy.

Dimming will continue for as long as PB0 is kept pressed, but the Nodes will only continue to either Full ON or Full OFF. Releasing PB0 and then keep pressing again will toggle the direction of the dimming.

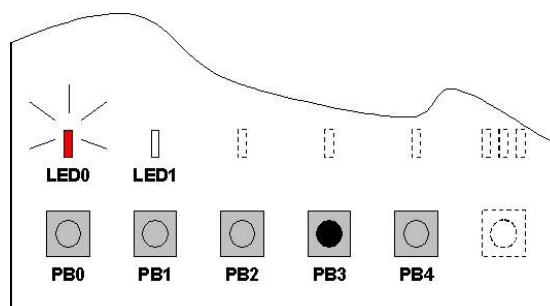
4.6 Assigning a Route to a Routing Slave

The Development Controller can also be used for assigning a Route to a Routing Slave e.g. a Binary Sensor. Please refer to [1] for detailed information on how the assigning of routes is actually handled by the protocol.

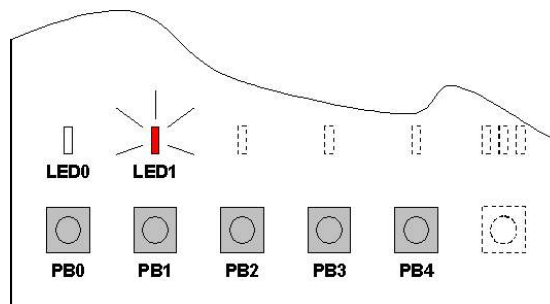
Assigning a Route to a Routing Slave is done to make the Routing Slave capable of reaching another Node even via routing when certain conditions are met. An example could be a setup consisting of a Binary Sensor and a LED Dimmer (or any other kind of Node). The user would like the LED Dimmer turned on, when the Binary Sensor is activated – this would require the assigning of a route to the Binary Sensor in order to make this Node know, which Node to control.

Assigning a Route to the Binary Sensor is done as described below:

1. Press shortly on PB3 – this will turn LED0 ON, indicating that the system is awaiting node information from the LED Dimmer that the Binary Sensor should get a route to.



2. Press on the pushbutton 3 times within 1,5 sec. on the Z-Wave module, which is a part of the LED Dimmer. This will prompt the Z-Wave Module to send out a node information frame.
3. When the node information has been received and processed by the Development Controller, LED0 will turn OFF and LED1 will turn ON, indicating that the system is awaiting node information from the Binary Sensor to which you want to assign a route.

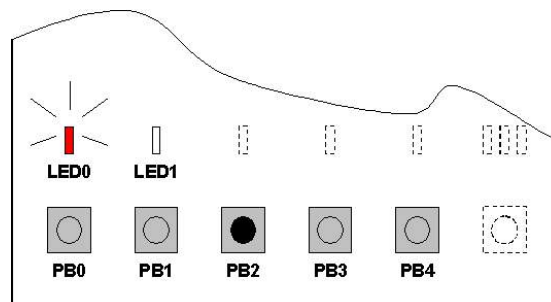


4. Now press the pushbutton 3 times within 1,5 sec. on the Z-Wave module, which is a part of the Binary Sensor.
5. When the node information has been received and processed by the Development Controller, LED1 will turn OFF, and a route has been assigned from the Binary Sensor to the LED Dimmer.
6. When the Binary Sensor is activated (in this case by pressing the pushbutton on the Binary Sensor) the LED Dimmer will turn on for a few seconds and then it turns off again.

4.7 Excluding a Node from the Z-Wave Network (reset Node)

The Development Controller application is capable of excluding Nodes and deleting them from the Z-Wave Network. The procedure is as follows:

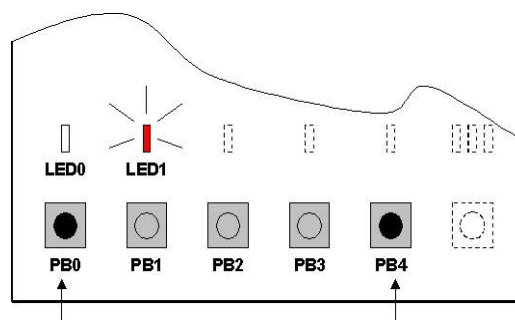
1. Press and hold PB2 on the Development Controller – this will turn LED0 ON, indicating that the system is awaiting node information from the Node to be excluded from the Network.
2. Press the pushbutton 3 times within 1,5 sec. on the Z-Wave slave Node to be excluded from the network.
Or press and hold the PB4 button on the secondary Development Controller to be excluded from the network.
3. When LED0 turns OFF on the Development Controller then the Node has been excluded and reset. PB2 can now be released.



4.8 Resetting the ZW0102 Development Controller

When resetting a Development Controller all information about the network topology is removed. Remember that resetting the primary Development Controller does not reset the Nodes that are included in the network that the Development Controller hosts. Each Node must be reset individually and this can be done either before or after having reset the primary Development Controller.

1. Press and hold both PB0 and PB4 on the Development Controller – after 2 seconds, LED1 will turn ON signaling that the reset is in progress, and the pushbuttons can be released.
2. After a few seconds LED1 will turn OFF and LED0 will flash shortly, meaning that the Development Controller has been reset.



4.9 Using Automatic Network Update Feature of Controllers

The secondary controllers will have an outdated network topology when the primary Development Controller includes/excludes new nodes. Having a static update controller (SUC) present in the network allows the secondary controllers to obtain the latest network topology by requesting network updates from the SUC. The PC based Controller application [5] can be used as SUC.

All changes in the network are transferred from the primary Development Controller to the SUC. The SUC can hold up to 64 changes of the network. A network change can be a node inclusion or exclusion to/from the Z-Wave network.

To use this functionality the following should be done:

1. Use the primary Development Controller to include a static controller to the network. The primary Development Controller will automatically try to allocate the first static controller included in the network to be a SUC. Refer to section 4.2 regarding how to include a node.
2. Add a new Development Controller to the network. During the inclusion process the new secondary Development Controller is informed about the presence of a SUC.
3. Now the secondary Development Controller will start requesting network updates from the SUC. A random request interval is used between 1 and 4 minutes.

When a new node is included to the network then the secondary Development Controller will receive this information at the next request to the SUC and will then be able to control it. Similar will the secondary Development Controller be updated when a node is removed.

4.10 Using Inclusion Controllers

Only the primary Development Controller has the capability to include/exclude nodes to/from the Z-Wave network. Having a SUC ID Server (SIS) present in the network allows all controllers to include/exclude nodes on its behalf. A SIS is a SUC with enabled node ID server functionality. The PC based Controller application [5] can be used as SIS.

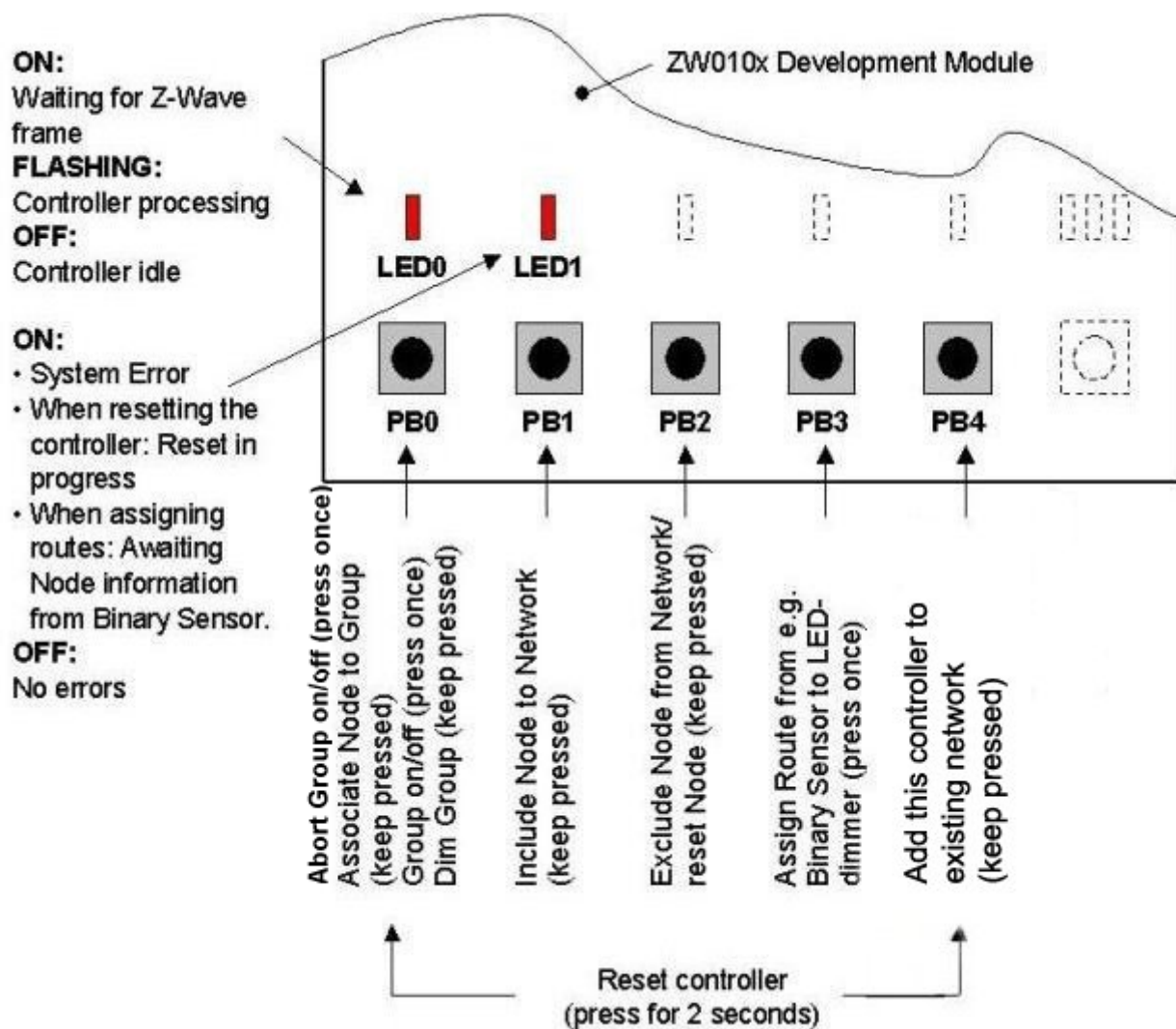
To use this functionality the following should be done:

1. Use the primary Development Controller to include a static controller to the network. The primary Development Controller will automatically try to allocate the first static controller included in the network to be a SIS. Refer to section 4.2 regarding how to include a node. In case a PC based Controller is used then remember to configure it to be a SIS when included to the network.
2. Add a new Development Controller to the network. During the inclusion process the new inclusion Development Controller is informed about the presence of a SIS.
3. Now the inclusion Development Controller will start requesting network updates from the SIS. A random request interval is used between 1 and 4 minutes. In addition it can include/exclude nodes in the network. The LED0 will flash on the inclusion Development Controller until a node ID is allocated from the SIS. When LED0 turns ON then the inclusion Development Controller is ready to include the node.

The SIS becomes the primary controller in the network because it has the latest update of the network topology and capability to include/exclude nodes in the network. The old primary Development Controller becomes an inclusion Development Controller. The SIS enables the inclusion Development Controller to include/exclude nodes in the network on its behalf.

5 FUNCTION OVERVIEW

The figure below summarizes the functions of the Development Controller and the descriptions that have been given in the previous sections.



6 REFERENCES

- [1] Zensys, INS10244, Instruction, Z-Wave Node Type Overview and Network Installation Guide
- [2] Zensys, DSH10087, Datasheet, ZW0102 Z-Wave Development Module
- [3] Zensys, DSH10033, Datasheet, ZM1220 Z-Wave Module
- [4] Zensys, DSH10086, Datasheet, ZW0102 Z-Wave Interface Module
- [5] Zensys, INS10240, Instruction, PC based Controller User Guide
- [6] Zensys, DSH10275, Datasheet, ZM2120C Z-Wave Module
- [7] Zensys, DSH10857, Datasheet, ZM3120C Z-Wave Module