System Design Specification

Z-Wave Protocol Overview

Document Part Number:	**903100105**
Document Revision:	**01.05**
Description:	**This document is a high level description of the Z-Wave Protocol.**
Written By:	**Peter Shorty**
Date:	**February 17, 2005**
Reviewed By:	**NTJ and JFR**
Audience:	**Z-Wave Partners and Zensys**

Approved by: Authority	Date
Niels Thybo Johansen, R&D	February 17, 2005

REVISION RECORD

Doc. Rev	Date	By	Pages affected	Brief description of changes
01.00	20010607	PSH	All	Initial version
01.01	20010713	PSH	All	Added description of collision avoidance
01.02	20020411	PSH	All	Corrected various spelling issues. Added description of routing slaves and static controllers to 3.2 Fixed section 3.2 Controller and Slave nodes so it matches description in Command class specification.
01.03	20030502	PSH	All	New document template used Clarified collision avoidance in static controllers Added installer and SUC to node types Added controller shift to replication
01.04	20041228	JFR	Paragraph 3.3.2 Paragraph 3.3.4	Added SUC ID Server (SIS) and inclusion controller Added Bridge controller
01.05	20050128	JFR		Removed paragraph about assignment of home and node ID Removed paragraph about controller replication

Table of Contents

1. INTRODUCTION

1.1 Purpose

The purpose of this document is to describe the Z-Wave^{TM} Radio Frequency Protocol that is used to communicate between the control node(s) and slave nodes in a Z-Wave system.

1.2 Scope

The scope of this document is to give an overview of:

- The MAC layer
- The Transfer Layer
- The Routing Layer
- The Frame Layer

1.3 Audience and Prerequisites

The audience of this document is Z-Wave partners and Zensys A/S.

2. NOTATION, TERMS AND SYMBOLS

EOF	End Of Frame
MAC	Media Access Control
PIR	Passive Infra Red movement sensor
RF	Radio Frequency
SIS	SUC ID Server
SOF	Start Of Frame
SUC	Static Update Controller

Table 1: Terms and abbreviations

3. Z-WAVE PROTOCOL

3.1 Overview

The Z-Wave protocol is a low bandwidth half duplex protocol designed for reliable wireless communication in a low cost control network. The protocols main purpose is to communicate short control messages in a reliable manner from a control unit to one or more nodes in the network.

The protocol is not designed to transfer large amounts of data or to transfer any kind of streaming or timing critical data.

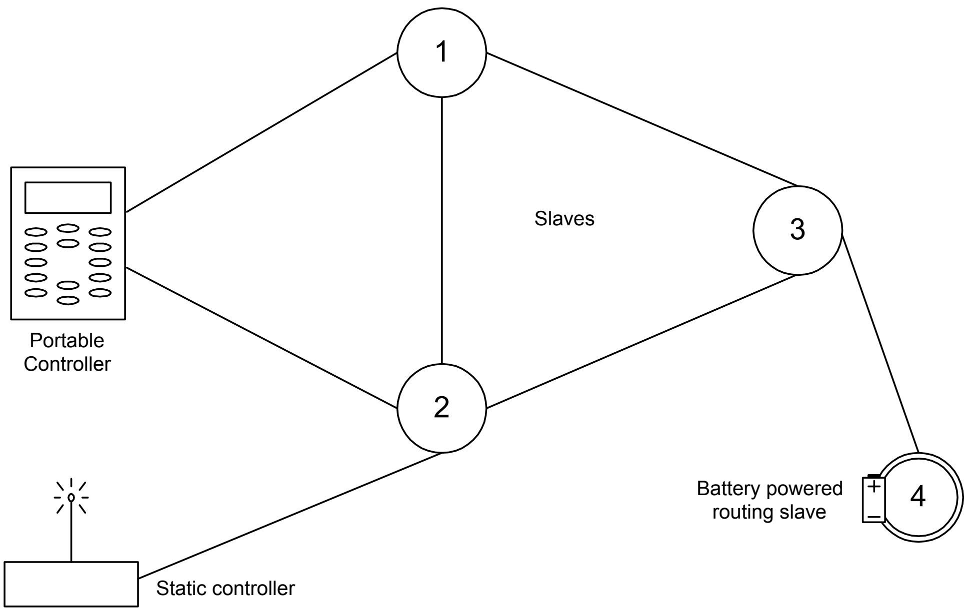
The protocol consist of 4 layers, the MAC layer that controls the RF media, the Transfer Layer that controls the transmitting and receiving of frames, the Routing Layer that controls the routing of frames in the network, and finally the application layer controls the payload in the transmitted and received frames.

Application Layer
Routing Layer
Transfer Layer
MAC Layer
RF Media

The 4 layers are described in the chapters 4 to 7.

3.2 Controller and Slave nodes

The Z-Wave protocol has 2 basic kinds of devices; controlling devices and slave nodes. Controlling devices are the nodes in a network that initiate control commands and sends out the commands to other nodes, and slave nodes are the nodes that reply on and execute the commands. Slave nodes can also forward commands to other nodes, which make it possible for the controller to communicate with nodes out of the direct radio wave reach.



3.3 Controllers

A controller is a Z-Wave device that has a full routing table and is therefore able to communicate with all nodes in the Z-Wave network. The functionality available in a controller depends on when it entered the Z-Wave network. In case the controller is used to create a new Z-Wave network it automatically become the primary controller. The primary controller is the “master” controller in the Z-Wave network and there can only be one in each network. Only primary controllers have the capability to include/exclude nodes in the network and therefore always have the latest network topology.
Controllers added to the network using the primary controller are called secondary controllers and don’t have the capability to include/exclude nodes in the network.

3.3.1 Portable Controller

A portable controller is a controller, which is designed to change position in the Z-Wave network. The portable controller uses a number of mechanisms to estimate the current location and hereby calculating the fastest route through the network.

An example of a portable controller could be a remote control.

3.3.2 Static Controller

A static controller is a fixed controller that mustn’t change position in the network and has to be powered up all the time. This controller has the advantage that Routing slaves can report unsolicited status messages to it, and it also has the advantage of always knowing where it is located in the network. A static controller will typically be a secondary controller in a Z-Wave network.

An example of a static controller could be an Internet gateway that monitors a Z-Wave system.

3.3.2.1 Static Update Controller

A Z-Wave network can optionally have a static controller with enabled Static Update Controller (SUC) functionality to distribute network topology updates. A SUC is a static controller that will receive notifications from the primary controller regarding all changes made to the network topology. In addition the SUC is capable of sending network topology updates to other controllers and routing slaves upon request. It is the application in a primary controller that requests a static controller to become a SUC. There can only be one SUC in a Z-Wave network.

3.3.2.2 SUC ID Server

A Z-Wave network can optionally have a SUC with enabled node ID server functionality (SIS). The SIS enables other controllers to include/exclude nodes in the network on its behalf. The SIS is 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. When including additional controllers to the network they become inclusion controllers because they have the capability to include/exclude nodes in the network on behalf of the SIS. The inclusion controllers network topology is dated from last time a node was included or it requested a network update from the SIS and therefore it can’t be classified as a primary controller.

3.3.3 Installer Controller

An Installer controller is a portable controller that has additional functionality, which enables it to do more sophisticated network management and network quality testing than other controllers.

An example of an installer controller could be an installation tool used by an installer to install a Z-Wave network at a customer site.

3.3.4 Bridge Controller

A Z-Wave network can optionally have a bridge controller. A bridge controller is an extended static controller, which incorporates extra functionality that can be used to implement controllers, targeted for bridging between the Z-Wave network and other networks. The bridge controller device stores the information concerning the nodes in the Z-Wave network and in addition it can control up to 128 virtual slave nodes. A virtual slave node is a slave node that corresponds to a node, which resides on a different network type.

An example of a bridge controller could be a bridge between an UPnP network and a Z-Wave network to link broadband and narrowband devices together in a home entertainment application.

3.4 Slaves

3.4.1 Slave

Slave nodes are nodes in a Z-Wave network that receives commands and performs an action based on the command. Slave nodes are unable to send information directly to other slaves or controllers unless they are requested to do so in a command.

An example of a slave node could be a light dimmer.

3.4.2 Routing Slave

Routing slaves has the same overall functionality as a slave. The major difference is that a routing slave can send unsolicited messages to other nodes in the network. They store a number of static routes for use when sending unsolicited messages to a limited number of nodes.

An example of a routing slave node could be a thermostat or a Passive Infra Red (PIR) movement sensor.

3.4.3 Enhanced Slave

Enhanced slaves have the same functionality as routing slaves and they are handled in the same way in the network. The difference between routing slaves and enhanced slaves is that enhanced slaves have a real time clock and an EEPROM for storing application data.

An example of an enhanced slave node could be a weather station.

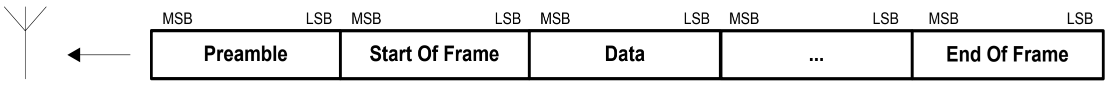
3.5 Home ID and Node ID

The Z-Wave protocol uses a unique identifier called the Home ID to separate networks from each other. The Home ID is a 32 bit unique identifier that is pre programmed in all controller devices. All slave nodes in the network will initially have a home ID that is zero, and they will therefore need to have a home ID assigned to them by a controller in order to communicate with the network. Controllers in a network can exchange home ID's so more than one controller can control slave nodes in a network.

Node ID's are used to address individual nodes in a network, they are only unique within a network defined by a unique home ID. A node ID is an 8 bit value and like home ID's they are assigned to slave nodes by a controller.

4. MAC LAYER

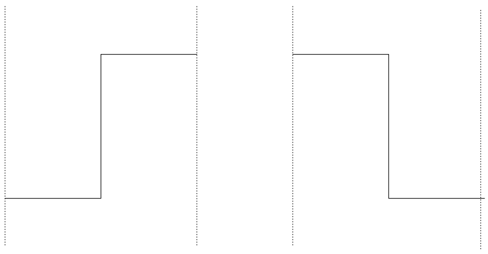
The Z-Wave MAC layer controls the radio frequency medium. The data stream is Manchester coded and consists of a preamble, start of frame (SOF), frame data and an end of frame (EOF) symbol. The frame data is the part of the frame that is passed on to the transport layer.



All data is sent in little endian format.

The MAC layer is independent of the RF media, frequency and modulation method but the MAC layer requires either access to the frame data when received or to the whole signal in binary form either as an decoded bit stream or to the Manchester coded bit stream.

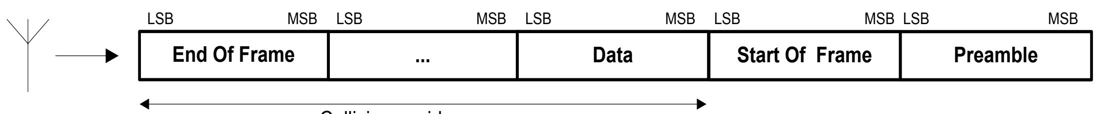
Data are transmitted in blocks of 8bit, most significant bit first and the data is Manchester coded in order to have a DC free signal.



Manchester 0 Manchester 1

4.1 Collision avoidance

The MAC layer has a collision avoidance mechanism that prevents nodes from starting to transmit while other nodes are transmitting. The collision avoidance is achieved by letting nodes be in receive mode when they are not transmitting, and then delay a transmit if the MAC layer is currently in the data phase in the receiver. The collision avoidance is active on all types of nodes when they have the radio activated.



The transmission of the frame is delayed a random number of milliseconds.

5. TRANSFER LAYER

The Z-Wave transfer layer controls the transfer of data between two nodes including retransmission, checksum check and acknowledgements.

5.1 Frame Layout

The Z-Wave transfer layer contains 4 basic frame formats used for transferring commands in the network. All 4 frames uses the following frame layout:

Bit

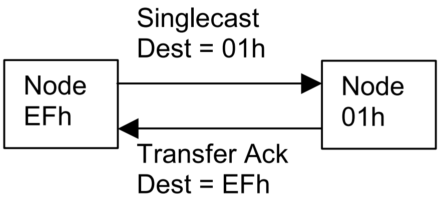
<table>
<tr><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="8">Home ID
..</td></tr>
<tr><td colspan="8">Source Node ID</td></tr>
<tr><td colspan="8">Frame header
..</td></tr>
<tr><td colspan="8">Length</td></tr>
<tr><td colspan="8">Destination address
..</td></tr>
<tr><td colspan="8">Data byte 0-x</td></tr>
<tr><td colspan="8">..</td></tr>
<tr><td colspan="8">..</td></tr>
<tr><td colspan="8">Checksum</td></tr>
</table>

Z-Wave basic frame format

5.1.1 Singlecast Frame Type

Singlecast frames are always transmitted to one specific node, and the frame is acknowledged so the transmitter knows that the frame has been received.

A singlecast transmission has the following frame flow.



If the singlecast frame or the transfer acknowledge frame is lost or corrupted, the singlecast frame is retransmitted. In order to avoid potential collisions with parallel systems the retransmissions are delayed with a random delay. The random delay must be in steps of the time it takes to send a frame of the maximum frame size and receive the Transfer Ack.

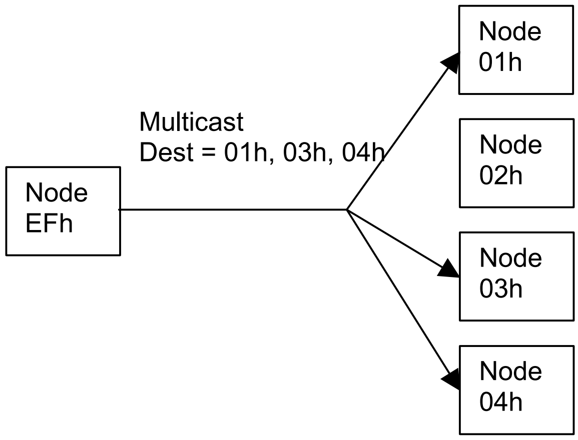
The singlecast frame can optionally be used without acknowledgement in a system where reliable communication isn't required.

5.1.2 Transfer Acknowledge Frame Type

The transfer acknowledge is a Z-Wave singlecast frame where the size of the data section is zero. For further description of the singlecast frame see section 5.1.1

5.1.3 Multicast Frame Type

Multicast frames are transmitted to a number of nodes ranging from 1 to 232 nodes. This frame type doesn't support acknowledge.

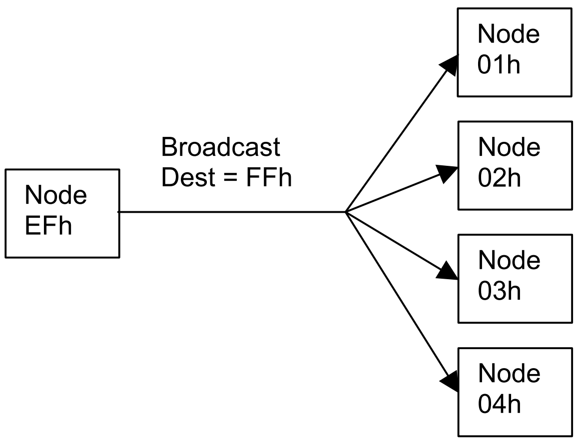


The multicast destination address is used to address selected nodes without having to send a separate frame to each node.

Note that a multicast frame doesn't get acknowledged so this type of frame can't be used for reliable communication. If reliable communication is needed a multicast must be followed by a singlecast frame to each destination node.

5.1.4 Broadcast Frame Type

Broadcast frames are received by all nodes in a network, and the frame is not acknowledged by any nodes.



Note that a broadcast frame doesn't get acknowledged so this type of frame can't be used for reliable transfer. If reliable communication is needed a broadcast must be followed by a singlecast frame to each destination node.

6. ROUTING LAYER

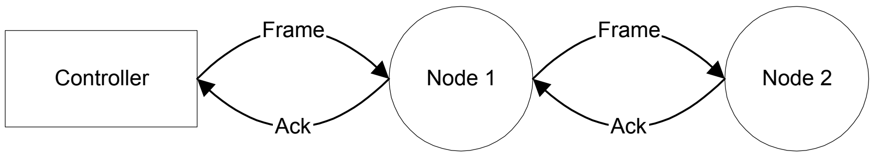
The Z-Wave routing layer controls the routing of frames from one node to another. Both controllers and slaves can participate in routing of frames in case they are always listening and have a static position. The layer is responsible for both sending a frame with a correct repeater list, and also to ensure that the frame is repeated from node to node. The routing layer is also responsible for scanning the network topology and maintaining a routing table in the controller.

6.1 Frame Layout

The Z-Wave routing layer has 2 kinds of frames that are used when repeating of frames is necessary.

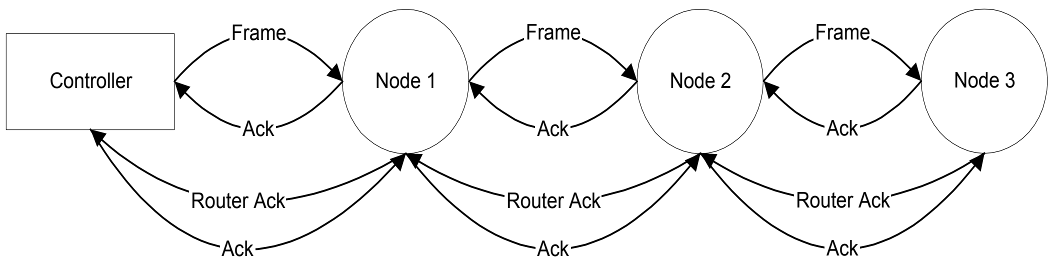
6.1.1 Routed Singlecast Frame Type

The Z-Wave routed singlecast is a one-node destination frame with acknowledge that contains repeater information. The frame is repeated from one repeater to another until it reaches its destination.



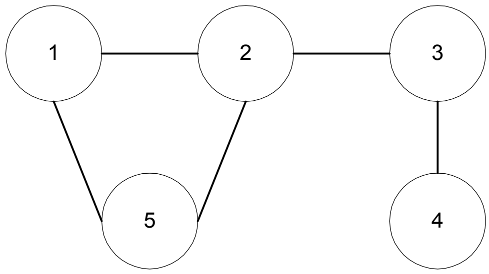
6.1.2 Routed Acknowledge Frame Type

The Z-Wave route acknowledge is a routed singlecast frame without payload that is used to tell the controller that the routed singlecast has reached its destination.



6.2 Routing Table

The routing table is where a controller keeps the information from the nodes about the network topology. The table is a bit field table where all information about what nodes that can see each other is kept. The figure below illustrates a network topology and the resulting routing table.



	1	2	3	4	5	6
1	0	1	0	0	1	0
2	1	0	1	0	1	0
3	0	1	0	1	0	0
4	0	0	1	0	0	0
5	1	1	0	0	0	0
6	0	0	0	0	0	0

Network topology and routing table.

The routing table is build by the primary controller based on information it receives from all the nodes in the network, at installation time, about each nodes range.

6.3 Route to Node

Finding the route to a node is a difficult task because a portable controller is defined as a device that will be moved around a lot (e.g. a remote control) Therefore a portable controller will always try to reach a node without routing and if that fails the portable controller will use several techniques to find the best route to the node.

7. APPLICATION LAYER

The Z-Wave application layer is responsible for decoding and executing commands in a Z-Wave network. The only part of the application layer that is described in this overview is the assignment of Home ID's and Node ID's and the replication of controllers. The rest of the application layer is implementation specific, and can be different from one implementation to another.

7.1 Frame Layout

The frame format used in the Z-Wave application layer is described in this section.

7.1.1 Application Layer Frame Format

Bit

7	6	5	4	3	2	1	0
Single/Multi/Broadcast frame header							
Application command class							
Application command							
Command parameter 1							
Command parameter 2							
...							
Command parameter x							

Z-Wave application frame format

Application Command class:
The application command class specifies which class of commands the command belongs to.

Currently defined command classes:

Command Class	Description
00h-1Fh	Reserved for the Z-Wave Protocol
20h-FFh	Reserved for the Z-Wave Application

Application command:
The application command specifies the specific command or action within the command class. The commands in the Application specific command classes are described in [1].

Command parameter 1-x:
The command parameters contain any parameters associated with the specified command. The number of parameters depends on the command.

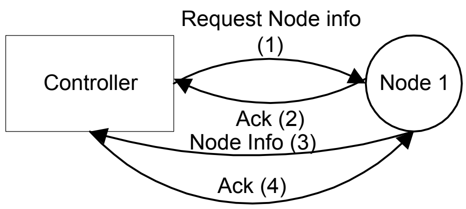
All frame types except acknowledge can contain an application command.

7.2 Node information

Because a controller in a Z-Wave network should be able to control many different kinds of nodes, it is necessary to have a frame that describes the capabilities of a node. Some of the capabilities will be protocol related and some will be application specific. All nodes will automatically send out their node information when the action button on the node is pressed. A controller can also get the node information from a node by requesting it with a “get node information” frame.

7.2.1 Node Information Frame Flow

The node information frame is send out by a node each time its action button is pressed. The frame is sent out as a broadcast to any controller/node that might be interested in the information. A controller can also request the node information from a node by sending a get node information frame to it.



Get node info frame flow

8. REFERENCES

[1] Zensys, 90310020x, Z-Wave Device Class Specification