ZigBee PRO Democase User Guide

1vv0300900 Rev.5 – 2013-09-24

APPLICABILITY TABLE

SPECIFICATIONS SUBJECT TO CHANGE WITHOUT NOTICE

Notice

While reasonable efforts have been made to assure the accuracy of this document, Telit assumes no liability resulting from any inaccuracies or omissions in this document, or from use of the information obtained herein. The information in this document has been carefully checked and is believed to be entirely reliable. However, no responsibility is assumed for inaccuracies or omissions. Telit reserves the right to make changes to any products described herein and reserves the right to revise this document and to make changes from time to time in content hereof with no obligation to notify any person of revisions or changes. Telit does not assume any liability arising out of the application or use of any product, software, or circuit described herein; neither does it convey license under its patent rights or the rights of others.

It is possible that this publication may contain references to, or information about Telit products (machines and programs), programming, or services that are not announced in your country. Such references or information must not be construed to mean that Telit intends to announce such Telit products, programming, or services in your country.

Copyrights

This instruction manual and the Telit products described in this instruction manual may be, include or describe copyrighted Telit material, such as computer programs stored in semiconductor memories or other media. Laws in the Italy and other countries preserve for Telit and its licensors certain exclusive rights for copyrighted material, including the exclusive right to copy, reproduce in any form, distribute and make derivative works of the copyrighted material. Accordingly, any copyrighted material of Telit and its licensors contained herein or in the Telit products described in this instruction manual may not be copied, reproduced, distributed, merged or modified in any manner without the express written permission of Telit. Furthermore, the purchase of Telit products shall not be deemed to grant either directly or by implication, estoppel, or otherwise, any license under the copyrights, patents or patent applications of Telit, as arises by operation of law in the sale of a product.

Computer Software Copyrights

The Telit and 3rd Party supplied Software (SW) products described in this instruction manual may include copyrighted Telit and other 3rd Party supplied computer programs stored in semiconductor memories or other media. Laws in the Italy and other countries preserve for Telit and other 3rd Party supplied SW certain exclusive rights for copyrighted computer programs, including the exclusive right to copy or reproduce in any form the copyrighted computer program. Accordingly, any copyrighted Telit or other 3rd Party supplied SW computer programs contained in the Telit products described in this instruction manual may not be copied (reverse engineered) or reproduced in any manner without the express written permission of Telit or the 3rd Party SW supplier. Furthermore, the purchase of Telit products shall not be deemed to grant either directly or by implication, estoppel, or otherwise, any license under the copyrights, patents or patent applications of Telit or other 3rd Party supplied SW, except for the normal non-exclusive, royalty free license to use that arises by operation of law in the sale of a product.

Usage and Disclosure Restrictions

License Agreements

The software described in this document is the property of Telit and its licensors. It is furnished by express license agreement only and may be used only in accordance with the terms of such an agreement.

Copyrighted Materials

Software and documentation are copyrighted materials. Making unauthorized copies is prohibited by law. No part of the software or documentation may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language or computer language, in any form or by any means, without prior written permission of Telit

High Risk Materials

Components, units, or third-party products used in the product described herein are NOT fault-tolerant and are NOT designed, manufactured, or intended for use as on-line control equipment in the following hazardous environments requiring fail-safe controls: the operation of Nuclear Facilities, Aircraft Navigation or Aircraft Communication Systems, Air Traffic Control, Life Support, or Weapons Systems (High Risk Activities"). Telit and its supplier(s) specifically disclaim any expressed or implied warranty of fitness for such High Risk Activities.

Trademarks

TELIT and the Stylized T Logo are registered in Trademark Office. All other product or service names are the property of their respective owners.

CONTENTS

1. Introduction

1.1. Aim of the Document

The goal of the ZigBee® DemoCase proposed by Telit is:

- to show to customers the possibilities offered by the ZigBee® technology
- to show to customers the know-how of Telit regarding this emerging technology
- to allow customers to start the development of their own application

1.2. Contact Information, Support

For general contact, technical support, to report documentation errors and to order manuals, contact Telit Technical Support Center (TTSC) at:

TS-SRD@telit.com
TS-NORTHAMERICA@telit.com
TS-LATINAMERICA@telit.com
TS-APAC@telit.com

Alternatively, use:

http://www.telit.com/en/products/technical-support-center/contact.php

For detailed information about where you can buy the Telit modules or for recommendations on accessories and components visit:

http://www.telit.com

To register for product news and announcements or for product questions contact Telit Technical Support Center (TTSC).

Our aim is to make this guide as helpful as possible. Keep us informed of your comments and suggestions for improvements.

Telit appreciates feedback from the users of our information.

1.3. Text Conventions

Danger – This information MUST be followed or catastrophic equipment failure or bodily injury may occur.

Caution or Warning – Alerts the user to important points about integrating the module, if these points are not followed, the module and end user equipment may fail or malfunction.

Tip or Information – Provides advice and suggestions that may be useful when integrating the module.

All dates are in ISO 8601 format, i.e. YYYY-MM-DD.

1.4. Reference documents

- [1] ZE51/61-2.4 RF module User Guide 1vv0300868
- [2] SR Manager Tool User Guide 1vv0300899
- [3] ZigBee PRO Democase Getting Started 1vv0300901
- [4] Z-ONE PRO Protocol Stack User Guide 1vv0300902

2. Description

2.1. General Description

2.1.1. The DemoCase philosophy

The goal of the ZigBee® DemoCase proposed by Telit is:

- to show to customers the possibilities offered by the ZigBee® technology,
- to show to customers the know-how of Telit regarding this emerging technology,
- to allow customers to start the development of their own application.

This DemoCase represents a complete ZigBee® solution in terms of hardware, integration, firmware and managing software.
It allows customers to create a complete ZigBee® network and to start developing their own application thanks to the numerous interfaces available on the DemoCase devices.
DemoCase can be ordered in two different versions:

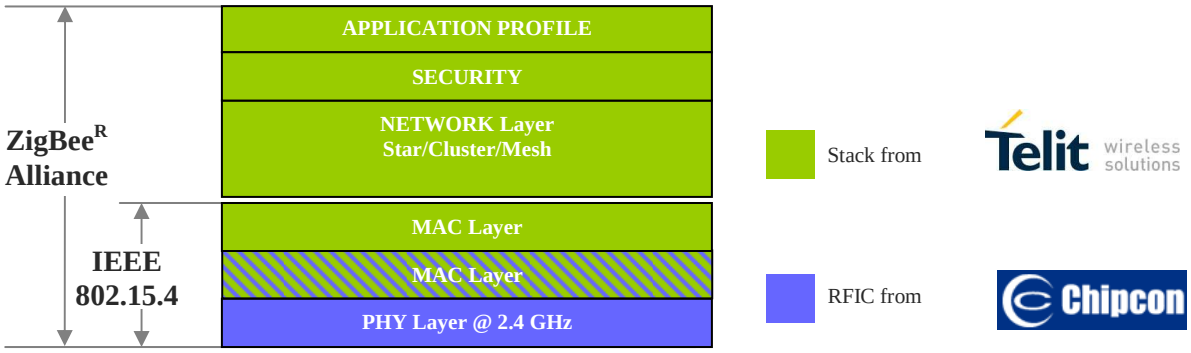
1. based on the ZE51-2.4 modules
2. based on the ZE61-2.4 modules

Both versions embed the Telit ZigBee® PRO stack (Z-One PRO).
This technical manual gives a detailed description of the DemoCase regarding the hardware and the software. It also gives a functional description of how creating and managing a ZigBee® network with the different elements of the DemoCase.

2.1.2. The ZE51/ZE61-2.4 module

Telit has developed two complete ZigBee®-ready solution integrated in a miniaturized SMD module, referenced ZE51-2.4 and ZE61-2.4. This module contains the radio part and the microcontroller with the ZigBee® PRO stack in only 26x15x3 mm^3.

It includes the complete ZigBee® PRO stack described below:



Find below its main characteristics:

Radio Chip :	Chipcon CC2530	**Microcontroller :**	SoC with 8051
Frequency Band :	2.4 GHz, worldwide	**Serial Interface :**	RS232 TTL
Channels :	16	**Serial Data Rate :**	115200 bauds
Radio Data Rate :	250 kbps		
		Power Supply :	2.4 to 3.6V
Output Power :	2.5 mW (ZE51)	**Consumption :**	35mA @ 3.6V (ZE51)
	100 mW (ZE61)		150mA @ 3.6V (ZE61)
		Stand-by Mode :	2µA
Sensitivity :	-97 dBm (ZE51) -100 dBm (ZE61)		
Typical Range :	up to 200m (ZE51) Up to 1500m(ZE61)	**Temperature :**	-40 to +85°C

2.1.3. The DemoCase devices

The DemoCase includes three ZE61s (one with Integrated antenna, two without) and three ZE51s (one with Integrated antenna, two without). One ZE61 without integrated antenna is flashed with Coordinator firmware. The other modules are flashed with End Device firmware.
Telit PC software SRManagerTool can be used to reflash the modules with different firmwares based on customer needs, router, end point or coordinator firmware (For more information see SRManagerTool User Guide [2]).
Telit PC software SRManagerTool allows setting of the parameters, sniffing RF packets and flashing of firmwares. It is available on Telit official web site www.telit.com under download zone together with dedicated User Guide and latest firmware for re-flash/upgrade.
Finally, all accessories necessary to the network implementation (serial cables, power supplies …) are provided in the DemoCase.

NOTE:

For OTA functionality contact Telit support.

2.2. List of equipments

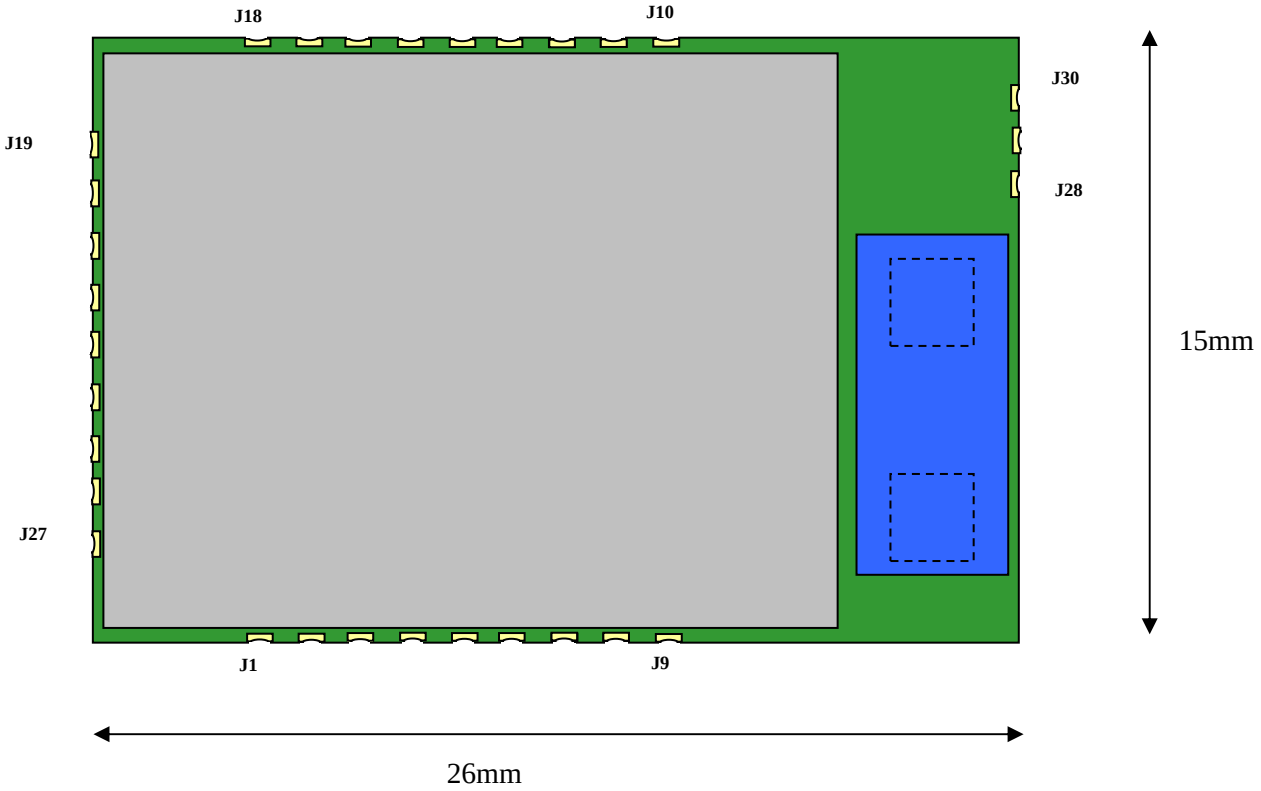
REF	ITEM	QTY
DemoCase		
COORD	Coordinator DIP Module(1)(2)	1
RFD	End Device DIP Modules(1)(3)	5
MOTHERBOARD	Board to plug DIP modules and I/O Demonstration Boards	4
I/O REPORT	I/O Demonstration Board	4
EA01	2.4GHz antenna, (SMA connector)	4
CB01	Demoboard USB Cable	4
PS01	AC/DC Block for Demoboard (+12V)	2
IT01	9V Cells	2

(1) Boards can be re-flashed to change their type
(2) ZE61 without integrated antenna.
(3) Two ZE61s (one with Integrated antenna, one without), three ZE51s (one with Integrated antenna, two without).

3. Hardware Description

3.1. The ZE51/ZE61-2.4 module

3.1.1. External description



3.1.2. Generic Pin out of the module

This pinout reflects the capability of the module. Some functions are not available with standard software, but can be included with a custom development.
All Analog Inputs can be used as Digital I/O

Pin	Pin name	DIR	Signal level	Function
J30	GND	-		Ground connection for External antenna
J29	Ext_Antenna	-		External antenna connection
J28	GND	-		Ground connection for External antenna
J27	GND	-		Ground
J26	GND	-		Ground
J25	VDD	-		Digital and Radio part supply pin
J24	CTS	I		Clear To Send
J23	RESET	I		µC reset
J22	RTS	O		Request To Send
J21	RXD	I		RxD UART – Serial Data Reception
J20	GND	-		Ground
J19	TXD	O		TxD UART – Serial Data Transmission
J18	STAND_BY	I		Standby
J17	GND	-		Ground
J16	PROG			
J15	GND	-		Ground
J14	DEBUG_D	I/O		
J13	GND	-		Ground
J12	GND	-		Ground
J11	GND	-		Ground
J10	DEBUG_C	I/O		
J9	RESERVED	-	-	-
J8	RESERVED	-	-	-
J7	IO7_A	I/O		Analog Input N°7 (Digital I/O capability)
J6	IO6_A	I/O		Analog Input N°6 (Digital I/O capability)
J5	IO5_A	I/O		Analog Input N°5 (Digital I/O capability)
J4	IO4_A	I/O		Analog Input N°4 (Digital I/O capability)
J3	IO3_A	I/O		Analog Input N°3 (Digital I/O capability)

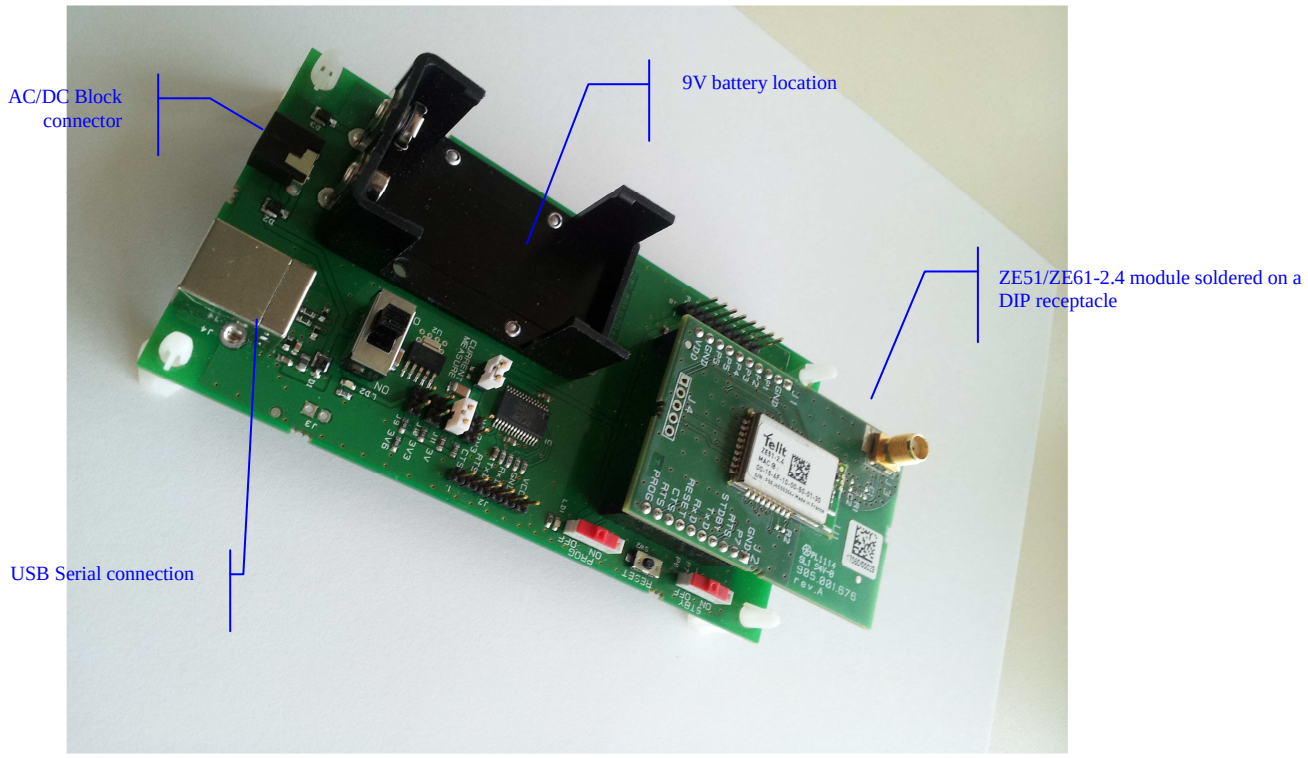
J2	IO2_P	I/O		Digital I/O N°2 with 20mA drive capability
J1	IO1_P	I/O		Digital I/O N°1 with 20mA drive capability

WARNING:

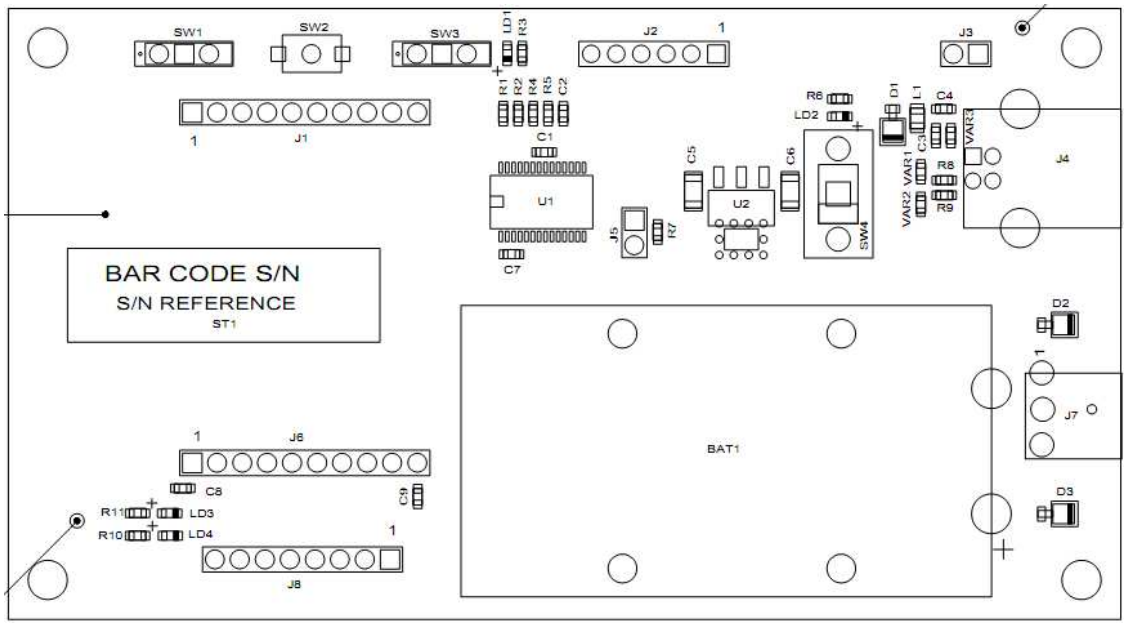
Reserved pins must not be connected

3.2. The Demoboard

3.2.1. External description



The demo mother board is the platform for the ZE module in DIP Version and can be connected to the PC via standard USB port.



Switches, connectors and LEDs placement on the Demo mother board

Designation	Feature
SW1	Stand-by switch
SW3	Programming switch
SW2	Reset push button
SW4	ON/OFF switch
LD1	PROG Yellow LED
LD2	ON/OFF Yellow LED
LD3	Red LED
LD4	Green LED

3.2.2. The Serial connection and power supply

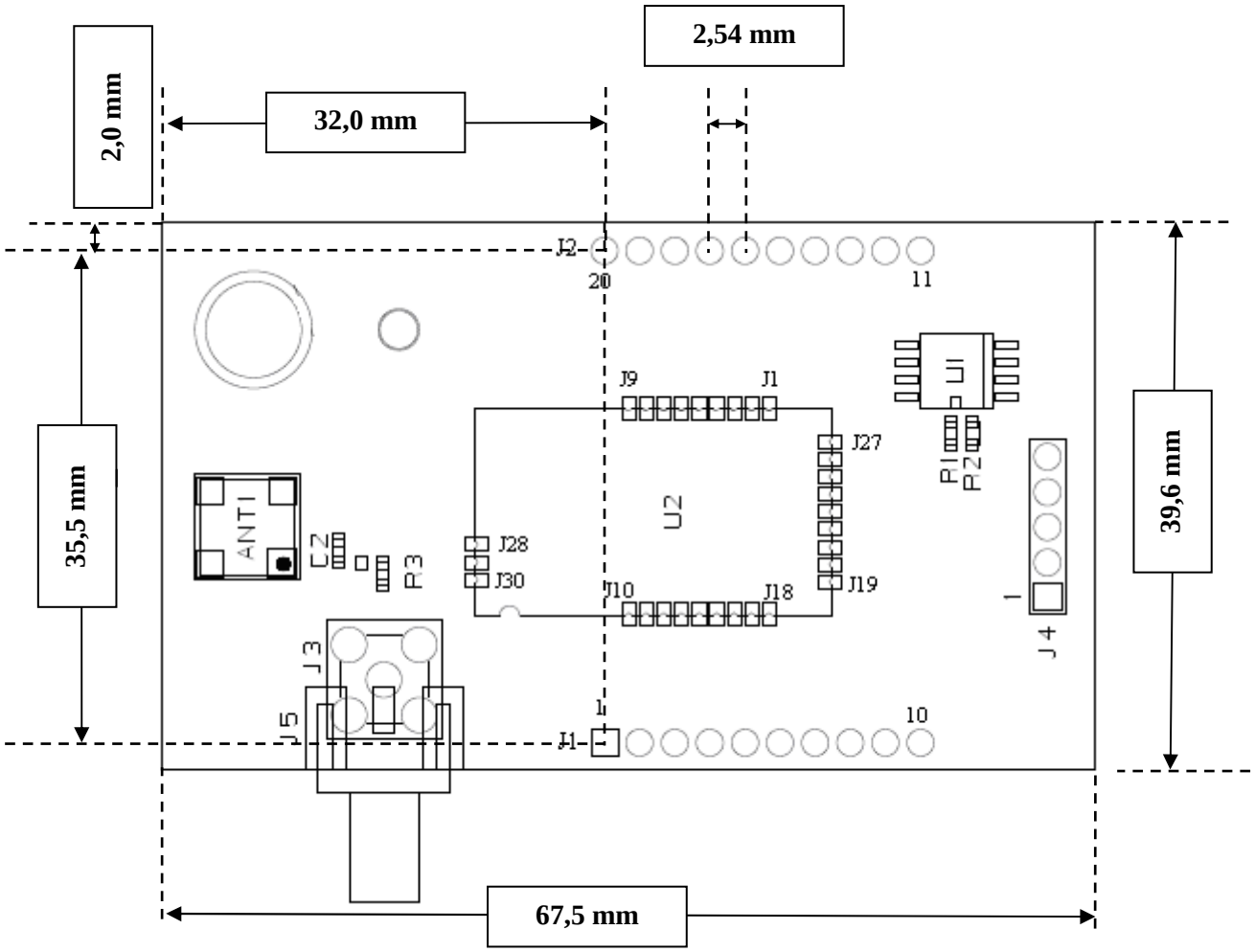
The serial connection is provided through an USB cable.
The power supply is brought :

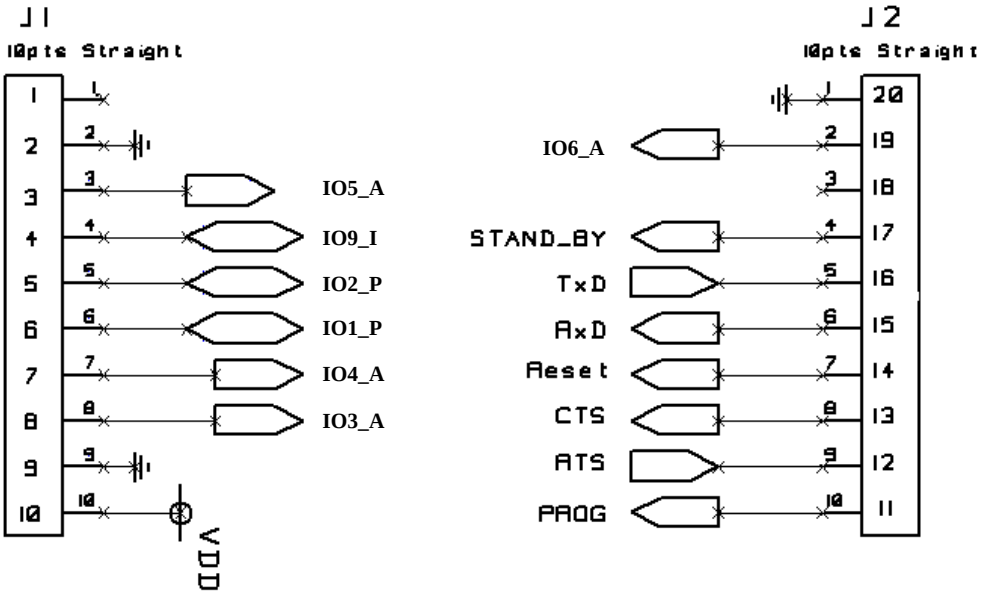
- Through the 12V AC/DC converter block for Coord, Router and End device,
- Through +9V battery for End device.

- Through the USB connection

External 12V power supply has priority against USB and Battery power supply.
USB power supply has priority against battery power supply.

3.2.3. DIP Support Pinout




J1
10pts Straight
1
2
3
4
5
6
7
8
9
10
IO5_A
IO9_I
IO2_P
IO1_P
IO4_A
IO3_A
VDD
J2
10pts Straight
20
19
18
17
16
15
14
13
12
11
IO6_A
STAND_BY
TxD
RxD
Reset
CTS
RTS
PROG

3.2.4. The I/O connection

All the I/Os of the ZE51/ZE61-2.4 module are available on the DIP receptacle and on the digital interface. Refer to the module pin-out (§3.1.2).

Name on Module	Description DemoBoard/IO Board	End Point
IO6		6: Switching Load
RTS		5: Switching Load
IO3		7: Analogical Sensor
IO4	Send on each rising edge and off on each falling edge. Have to be wake up.	2: Switch Remote Control
IO1	Red LD3/Yellow LD2	-
IO2	Green LD4/Red LD1	-
IO5	Interruptible Pin. Send toggle on each rising edge.	4: Switch Remote Control

3.2.5. The antenna

The antenna used for DemoBoard is SMA rubber antenna referenced Titanis from GigaAnt. Find below its main characteristics :

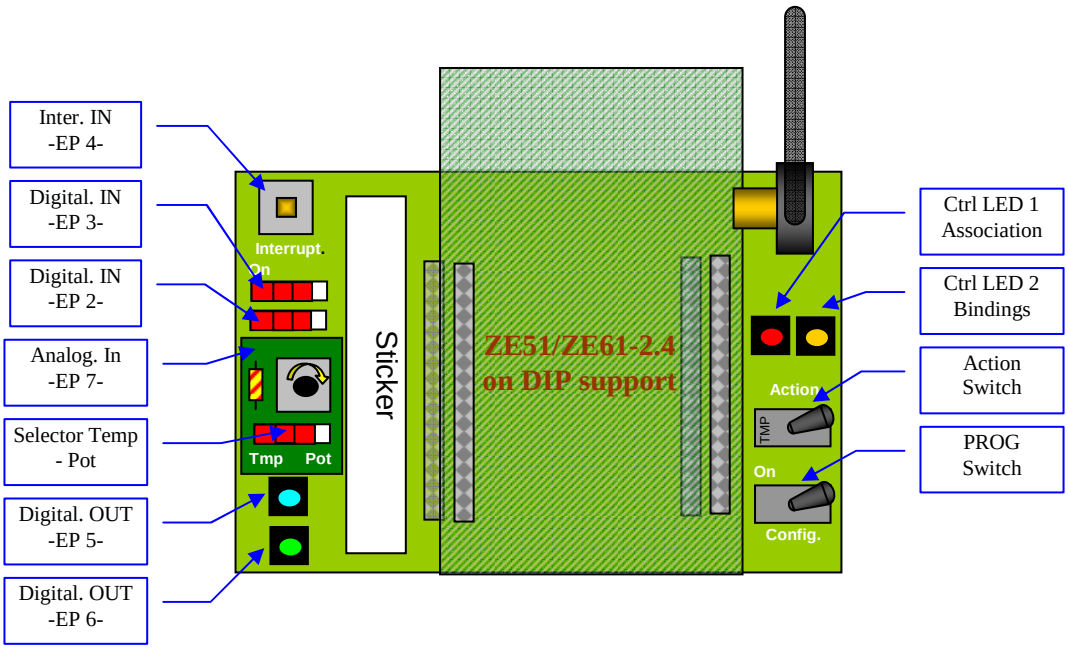
Bandwidth : 2.30 – 2.50 GHz
VSWR : < 1:1.5
Gain : 4 dBi

3.2.6. I/O Report

This board is used for an easy access to the control switches as well as to the Inputs/Outputs used for demonstration.

It is very useful for a quick start-up of a network, allowing to associate, bind and test the board features in a very short time.

It is mounted between the Demoboard itself and the DIP support



4. FUNCTIONAL DESCRIPTION

4.1. Network construction

4.1.1. DEMOCASE setup

Before all operations it is necessary to choose the extended PAN-ID, the Channel Mask, using security, the Nwk Key and the mode Preconfigured, by SRManagerTool in the Configuration Wizard (see the SRManagerTool user guide [2] for more information).
Once this configured, the module is ready to enter or to begin a network. When the device are connected at the network all the information about the network is stored, so the device keeps the network information and takes the same place in the network after a Switching on/off.
To reset the device it is necessary to put the STBY pin at high and after put the PROG pin at high. During the blinking LD4 release all pin.
To reset the device with the default value it is necessary to put the STBY pin at high during 20 seconds. During the blinking LD4 release all pin.
For more information on network creation and device joining see section 4.3.

4.1.1.1. Coordinator

To start a network, send the Start command on the serial link, or switch on/off the STBY Pin; from then the coordinator scans the selected channels and selects a non-noisy channel. The LD4 led blinks and stays red when the network is done.
By default the coordinator accepts association, so all devices can make an association on it.

4.1.1.2. Router

To associate a DemoCase in FFD mode, send the Start command on the serial link, or switch on/off the STBY Pin. The LD4 led blinks and stays green when the association is done.
By default the FFD accepts association, so all devices can make an association on its.

4.1.1.3. End Device

To associate a DemoCase in RFD mode, send the Start command on the serial link, or switch on/off the STBY Pin. The LD4 led blinks and stays green when the association is done.

If the End Device is sleeping and the CTS pin of the host is high the led LD4 will stay on for some milliseconds (depending on what the device has to do when is awake) every sleeping time period.

If the CTS pin is set LOW by the host the module stays awake and the LD4 stays on.

NOTE:

during association or creation of the network, depending on the status of IO board switches or if the demoboard is used without IO Board, from the serial link can be received AF Indirect Confirm with status set to 0xA8 (APS_NO_BOUND_DEVICE).

Binding:
Binding with End Device Binding command: once devices are associated to the network, keeps the STBY pin at high until the choose End Point on the first device. The First blinking of the LD3 corresponds to the First End Point, the second blinking to the second End point… Makes the same operation on the other device before 20 seconds. The LD3 stays ON while the binding is not made and it switches off at the end of the process. If at the end the LD4 blinks then the process didn't have worked.

Unbinding:
Unbinding with End Device Binding: makes the same operation that binding.

End Point:
There are 8 End Points by default on each device. The 1st End Point is the serial Link. The 2nd and 3rd are a switch remote control, when the Pin is high the ON command is sent and when the Pin is Low the OFF command is sent. The 4th is a switch remote control too, but it sends a TOGGLE command on rising edge. The 5th and 6th are a switching load End Points. The 7th is an analog End Point. The 8th (EP10) is a switch remote control end point and a analog read request end point (Note this end point cannot be bind using STBY pin).

End Point	Functionality	Comments
1	Serial link	Payload has to be < 80 bytes
2	Switch remote	Send ON on High Level and Send OFF on Low level.
3	Switch remote	Send ON on High Level and Send OFF on Low level.
4	Switch remote	Interruptible, send TOGGLE on rising edge.
5	Switch Load	-
6	Switch Load	-
7	Analogical sensor	Value on 2 bytes
10	Switch remote and Analogical read request	Cannot bind using STBY switch

4.1.2. DEMOCASE Profile

This profile is designed to demonstrate how to use the Zigbee PRO Stack with some different applications:
On/Off
Analogical Measure
Data exchange

The Profile ID is **0xC07C**

4.1.2.1. Device Descriptions

Device descriptions specified in this profile are summarized in the next table along with their respective Device IDs.

Device	DeviceID
On/Off Switch	0x 0000
On/Off Output	0x 0002
Analogical Measure	0x 0020
Data Exchange	0x 0010
Tool	0x 0030

4.1.2.2. Device Specifications

On/Off Switch
The On/Off Switch device is capable of sending command to control a remote Output.
Supported Clusters:

Server side	Client side
None	On/Off

On/Off Output
The On/Off Output device is capable to switch our own Output regarding a received command.
Supported Clusters:

Server side	Client side
On/Off	None

Analogical Measure
The Analogical Measure device is capable to read an analogical input.
Supported Clusters:

Server side	Client side
Analogical	None

Data Exchange
The Data exchange device is capable to send and receive data from/to RS232.
Supported Clusters:

Server side	Client side
Serial Data	Serial Data

Tool
The Tool device is capable to control and manage Analogical Measure or On/Off Output devices.
Supported Clusters:

Server side	Client side
None	On/Off
	Analogical

4.1.2.3. Cluster specifications

The different clusters are developed on Zigbee Cluster Library ("075123r02ZB_AFG-ZigBee_Cluster_Library_Specification.pdf") except the Serial Data Cluster. For other it is necessary to use the format of frame described in ZCL (Chapter 2.3). In this ZCL it is described how to read an attribute or how to send a command to a cluster.

On/Off
Attributes and commands for switching devices between 'On' and 'Off' states.
ClusterID: **0x0004**
Attributes:

Identifier	Name	Type	Range	Access
0x0000	OnOff	Boolean	0x00-0x01	Read Only

Command:

Off	**0x00**
On	**0x01**
Toggle	**0x02**

Read ZCL specification for the using of the Cluster.

Analogical
Attributes for reading the analogical input of remote devices.
ClusterID: **0x0020**
Attributes:

Identifier	Name	Type	Range	Access
0x0000	Analogical Value	UINT-16	0x0000-0x03FF	Read Only

Command:

None

Effect on Receipt:
At the reception of the reading attribute, the application read the analogical input on 10 bits and sends the answer.

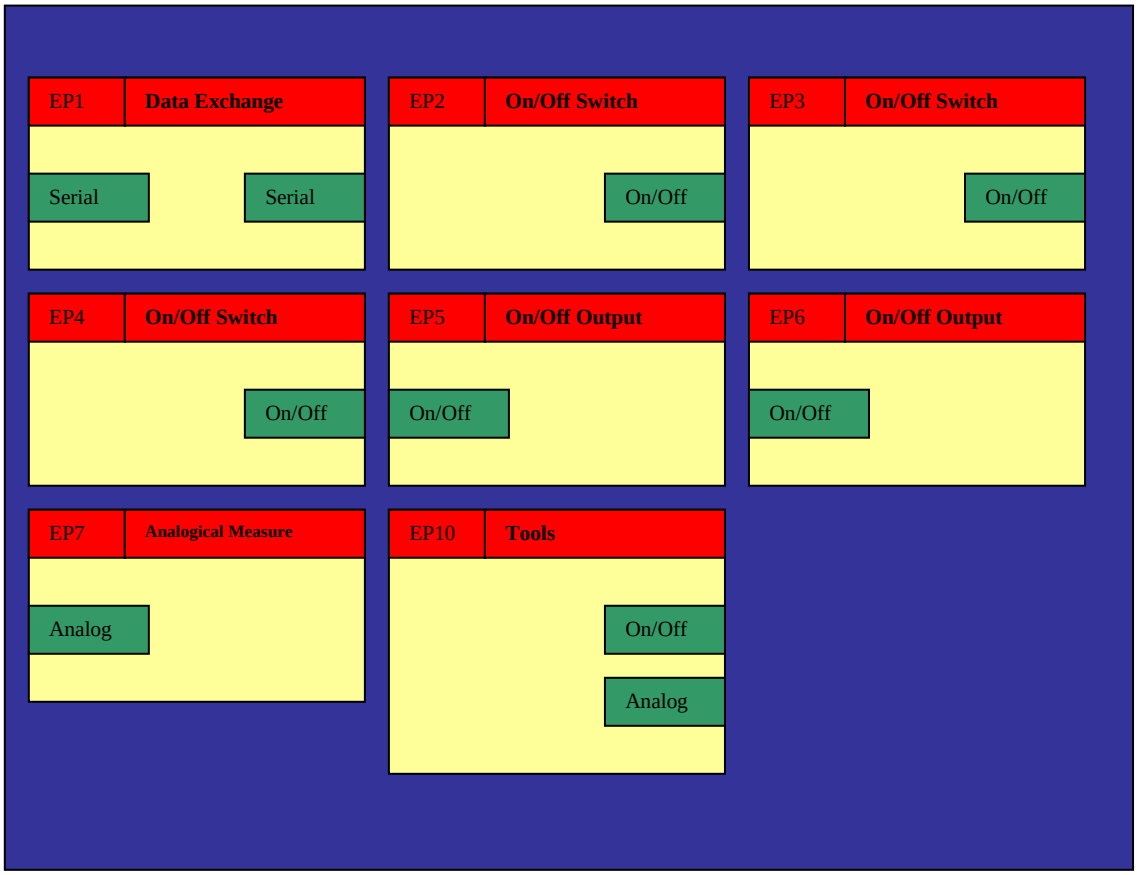
Serial Data
ClusterID: **0x0060**
The format of the *afdu* frame is specific and very simple as following:

Afdu
Serial Data (MSB First)

Frame Generated:
The client generates the frame at reception of data on serial link.
Effect on Receipt:
At the reception of this frame, application sends *afdu* on the serial link.

4.1.2.4. Applications Framework of Device

All devices of the demo case have the following applications:



4.2. Device configuration

To communicate with a device which has a serial interface (PAN-C, FFD, Demoboard), a configuration interface has been done. The **serial configuration** has to be **115200,8,1,N**.
To go to the serial configuration mode there is 2 possibilities:
send '+++' / the device responds [0x0D]
Put the Pin 'PROG' at 1.
To exit this mode:
send [0x01][0x00] / the device responds [0x02][0x00][0x00]
Put the Pin 'PROG' at 0.

With this configuration interface, it is possible to manage the internal register of the device and with the coordinator to manage the network.

4.2.1. General format

The general format of the serial frame is:

Frame Length (1 Byte)	Command (1 Byte)	Payload (Variable)

With Frame Length = 1 + Length of the Payload.

4.2.2. Internal Registers

4.2.2.1. Primitives

Set Request

Offset	Name	Value	Description
0	Frame Length	-	
1	Command	0x12	
2	Attribute	0x00-0xFF	Defined by users
3	Attribute Length	0x01-0xFF	The number of byte of the attribute
4 and more	Attribute Value	-	Set of octets

Set Confirm

Offset	Name	Value	Description
0	Frame Length	3	
1	Command	0x13	
2	Status	0x00-0xFF	0x00: Success
3	Attribute	0x00-0xFF	Attribute defined by users

Get Request

Offset	Name	Value	Description
0	Frame Length	2	
1	Command	0x14	
2	Attribute	0x00-0xFF	Defined by users
3	Attribute Value	Variable	It is an optional field, generally it is not present and its presence is indicated in the specific attribute.

Get Confirm

Offset	Name	Value	Description
0	Frame Length	AttributeLength +4	
1	Command	0x15	
2	Status	0x00-0xFF	0x00: Success
3	Attribute	0x00-0xFF	Defined by users
4	Attribute Length	0x01-0xFF	The number of byte of the attribute
5 and more	Attribute Value	-	The Set of octets

4.2.2.2. Attribute

All attributes are listed below.

Access	Attribute	Length	Name	Description
R	0x6F	8	IEEE Address	The Extended Address of the device
R/W	0x10	1	Join Type	0x00: One shot join 0x01: Periodic Join. If the join fail the device will try to join the network every sysRetJoinPeriodP1 seconds for sysRetJoinRetriesP1 Attempts (Phase 1) after that it will try to join every sysRetJoinPeriodP2 seconds (Phase 2). Default: 0x00 (One shot)
R/W	0x11	4	Join Period PHASE 1	Interval (seconds) between two join attempt in Phase 1 Range: 0x1E – 0xFFFFFFFF Default: 0x3C
R/W	0x12	4	Join Period PHASE 2	Interval (seconds) between two join attempt in Phase 2 Range: 0x78 – 0xFFFFFFFF Default: 0xE10
R/W	0x13	1	Join Retries PHASE 1	Number of join attempts in Phase 1. Range: 1-255 Default: 15
R/W	0x14	1	Jitter Phase 1	Jitter (seconds) between two attempts of join in PHASE 1. Range: 0-255 Default: 15
R/W	0x15	1	Jitter Phase 2	Jitter (minutes) between two attempts of join in PHASE 2. Range: 0-255 Default: 30
R/W	0x1A	1	Disable Compiled Simple Descriptors	This parameter allows to disable/enable the simple descriptor defined by the compiled Profile: 1=Disable, 0= Enable. Range: 0-1 Default: 0

<table>
<tr>
<td>R/W</td>
<td>0x1B</td>
<td>Variable</td>
<td>Read/Write Simple Descriptor</td>
<td>This function allows to add a new simple descriptor. The simple descriptor will be removed only after a software HARD reset.
The first byte in the “Attribute Value” is the end point, the other bytes are the simple descriptor. The simple descriptor format is:

<table>
<tr><th>Field</th><th>Length (Byte)</th></tr>
<tr><td>Profile ID</td><td>2</td></tr>
<tr><td>Device ID</td><td>2</td></tr>
<tr><td>Device Ver.</td><td>1</td></tr>
<tr><td>In Cluster Count</td><td>1</td></tr>
<tr><td>Out Cluster Count</td><td>1</td></tr>
<tr><td>In Cluster List</td><td>2 * In Cluster Count</td></tr>
<tr><td>Out Cluster List</td><td>2 * Out Cluster Count</td></tr>
</table>

In the get the optional field “Attribute Value” shall be present and it has to hold the end point. End points shall be in the range 8-9</td>
</tr>
<tr>
<td>R/W</td>
<td>0X52</td>
<td>1</td>
<td>RxOnWhenIdle</td>
<td>Only on the RFD. This parameter indicates whether the device can be expected to receive packets over the air during idle portions. It can be set only before the joining is started.
If it is 0x00 the device will be sleeping, otherwise the device will stay awake in IDLE state.
Range: 0x00-0x01
Default: 0x01</td>
</tr>
<tr>
<td>R/W</td>
<td>0X56</td>
<td>1 or 4</td>
<td>Sleeping Time</td>
<td>Only on the RFD. This parameter indicates how much time in second the RFD makes its synchronization with its parent.
In the set Attribute request it could be 1 or 4 bytes long.
In the get Attribute request it is 4 bytes long.
If the attribute is one byte long its value has to be between 1 and 60 seconds.
If the attribute is four bytes long its value has to be between 1 and 0xFFFFFFFF.
In cyclic wakeup mode, the CTS pin can be used to wake up the device if needed (configuration, emergency frame …)
Range:0x00000001-0xFFFFFFFF
Default: 0x00000003</td>
</tr>
</table>

<table>
<tr><td>R/W</td><td>0x57</td><td>1</td><td>Rejoin Type</td><td>It is a bit mask and indicate how the rejoin is configured.
<table>
<tr><td>bit</td><td>7-3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>Read/write</td><td>Reserved</td><td>Secure rejoin enabled</td><td>Unsecure Rejoin enabled</td><td>Periodic Rejoin enabled</td></tr>
<tr><td>Reset</td><td>0</td><td>1</td><td>1</td><td>1</td></tr>
</table>
Bit 1 and 2 cannot be set to 0 at the same time.
Range: 0x02-0x07
Default: 0x07</td></tr>
<tr><td>R/W</td><td>0x58</td><td>4</td><td>Rejoin Interval</td><td>Interval (seconds) between rejoins of the same attempt in the same channel and between different attempts in the first phase.
Range: 0x01-0xFFFFFFFF
Default: 60</td></tr>
<tr><td>R/W</td><td>0x59</td><td>4</td><td>Max Rejoin Interval</td><td>Interval (seconds) between rejoin attempts after the first Phase.
Range: Range: 0x01-0xFFFFFFFF
Default: 900</td></tr>
<tr><td>R/W</td><td>0x5A</td><td>1</td><td>Max Rejoin Retries first Phase</td><td>Number of attempts for which Rejoin interval is used.
If it is 0xFF the Rejoin Interval is used as interval between every Attempt.
Range: 0x01-0xFF
Default: 0xFF</td></tr>
<tr><td>R/W</td><td>0x5B</td><td>1</td><td>Secure Rejoin Retries</td><td>Number of secure Rejoin retries per Attempt.
Range: 0x01-0xFF
Default: 1</td></tr>
<tr><td>R/W</td><td>0x5C</td><td>1</td><td>Rejoin Retries</td><td>Number of Rejoin retries per Attempt. Pay attention that if the secure rejoin is enable the module will try secure rejoin before the unsecure one.
Range: 0x01-0xFF
Default: 1</td></tr>
<tr><td>R/W</td><td>0x01</td><td>2</td><td>Radio Channel</td><td>Manages the channels mask in which the device will try to associate.
It is on 2 bytes representing the channel 11 to 26.
It is writable only before association.
<table>
<tr><td>bit</td><td>15</td><td>14</td><td>…</td><td>1</td><td>0</td></tr>
<tr><td>Read/write</td><td>Channel 26</td><td>Channel 25</td><td>Channel …</td><td>Channel 12</td><td>Channel 11</td></tr>
<tr><td>Reset</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr>
</table></td></tr>
<tr><td>R</td><td>0x00</td><td>1</td><td>Current Channel</td><td>Return the current channel of the device
Range:0x0B-0x1A
Default:0x0B</td></tr>
<tr><td>R</td><td>0x04</td><td>String</td><td>Version Stack</td><td></td></tr>
<tr><td>R</td><td>0x05</td><td>String</td><td>Version Bootloader</td><td></td></tr>
<tr><td>R</td><td>0x0A</td><td>String</td><td>Version Application</td><td></td></tr>
<tr><td>R/W</td><td>0xC4</td><td>8</td><td>ExtendedPanID</td><td>ExtendedPanID used to start or to associate to a network.
Default: 0x0000000000000000</td></tr>
<tr><td>W</td><td>0xCA</td><td>8</td><td>TrustCenter</td><td>Only when device is associated</td></tr>
</table>

<table>
<tr><td>R/W</td><td>0x0C</td><td>1</td><td>Serial speed</td><td>Set/Get serial speed Id:
<table><tr><td>Speed Id</td><td>Serial Speed (Baud/Sec)</td></tr><tr><td>3</td><td>9600</td></tr><tr><td>4</td><td>19200</td></tr><tr><td>5</td><td>38400</td></tr><tr><td>6</td><td>57600</td></tr><tr><td>7</td><td>115200</td></tr></table>Default: 7 (115200 Baud/Sec)</td></tr>
<tr><td>R/W</td><td>0x99</td><td>1</td><td>USB device</td><td>Define if the device is a USB dongle. (0: normal – 1: USB)
Range: 0x00-0x01
Default: 0x00</td></tr>
<tr><td>R</td><td>0x06</td><td>1</td><td>Type of device</td><td>0x10= Pan-C, 0x11= FFD, 0x12=RFD
0x90= Pan-C on USB, 0x91= FFD on USB, 0x92= RFD on USB</td></tr>
<tr><td>R</td><td>0x07</td><td>1</td><td>Is associated</td><td>1=associated, 0=not associated
Range: 0x00-0x01
Default: 0x00</td></tr>
<tr><td>R</td><td>0x96</td><td>2</td><td>Nwk address</td><td>The 16-bit address of the module
Range: 0x0000-0xFFFF
Default: 0xFFFF</td></tr>
<tr><td>R/W</td><td>0xC9</td><td>1</td><td>Fragmentation Inter Frame Delay</td><td>Time before retransmission of fragment
Range: 0-255
Default : 100</td></tr>
<tr><td>R/W</td><td>0xCD</td><td>2</td><td>Fragmentation Window Size</td><td>The first byte is the end point and the second one is the window size.
The End Point value has to be between 1 and 240.
Window size value has to be between 1 to 3 (number of blocks used for fragmentation transmission).
If windows size is set to 0xFF it is reset to default for the specific End Point.
In the get the optional field “Attribute Value” shall be present and it has to hold the end point.
Range : 0x01-0x03,0xFF
Default : 0x03</td></tr>
<tr><td>R/W</td><td>0xD0</td><td>1</td><td>End Device Binding Timeout</td><td>Only on the Coord. It is the timeout in seconds for End Device Binding management.
Range: 0x01-0xF0
Default: 0x14</td></tr>
<tr><td colspan="5">Security</td></tr>
<tr><td>R/W</td><td>0xA3</td><td>1</td><td>Use Security</td><td>Identify if the device uses security: 0=Disabled, 1= Enabled.
Range: 0-1
Default: 0</td></tr>
</table>

R/W	0xE4	1	HasPreconfiguredNwkKey	Identify if the device has a Nwk Preconfigured Key: 0=Disabled, 1= Enabled. Range: 0-1 Default: 0
W	0xA4	16	Nwk Key	Configure the Key for all devices if are in Preconfigured Key else just the Coordinator must be configured.
R/W	0xE5	1	HasPreconfiguredTCLinkKey	On the Coordinator identifies if the Trust Center Link Key mechanism is enabled (1). On the other devices identifies if they have a Trust Center Link Preconfigured key: 0=Disabled, 1= Enabled. Range: 0-1 Default: 0
W	0xA5	24	Link Key(16) + IEEE address(8)	Configure the Trust Center link Key shared with the Trust Center. IEEE address is the Trust Center address. On the joining device: if the IEEE address is set to 0x0000000000000000 the device will try to join all the coordinators scanned until will not find the one with the correct link key. On the coordinator: if the IEEE address is set to 0x0000000000000000 it will manage all the devices with the same trust center link key.
R/W	0x02	1	Output power Attenuation	Sets an attenuation on output power with 1dB step. Range: 0-25 Default: 1.
R	0x50	2	PAN Id	Returns the current PAN Id of the network the device is associated with.

4.2.3. Network Management

To configure and manage the network it is possible to use primitives listed below. All primitives are available on each device (Coordinator, FFD, RFD).

4.2.3.1. Start

This primitive is used to start the network or to associate the device to the network.

Start request

Offset	Name	Value	Description
0	FrameLength	1	
1	Command	0x16	

Start confirm

Offset	Name	Value	Description
0	FrameLength	2	
1	Command	0x17	
2	Status	SUCCESS,ERROR_ACQ_COORD, ERROR_SCANNING,ERROR_JOIN_ROUTER, ERROR_ACQ_DEVICE	Confirm Status Table

4.2.3.2. EndDevice Annonce

This primitive is receipt on all devices when a new device has joined the network.

EndDevice Annonce indication

Offset	Name	Value	Description
0	FrameLength	12	
1	Command	0xD5	
2-3	Short Address(Little Endian)	-	Nwk Address of the new device
4-11	IEEEAddress (Little Endian)	-	IEEE Address of the new device
12	Capabilty	-	Capability of the new device

4.2.3.3. Set Installation Code

This command is used to set an Installation code used to create a Trust Center Link Key.
The request format is different depending on the fact it has been issued to the trust center or to another device.

Set Installation Code request (Trust Center)

Offset	Name	Value	Description
0	FrameLength	11 + Installation Code Lenght	
1	Command	0x46	
2	Installation Code Size ID	0x00-0x03	Installation code size ID: **Size Id** / Installation Code Size without CRC (Bytes) / Installation Code Size with CRC (Bytes) **0** / 6 / 8 **1** / 8 / 10 **2** / 12 / 14 **3** / 16 / 18
3	CRC Settings	0x00-0x02	0x00: The installation code has the CRC but it shall not be verified 0x01: The installation code has the CRC and it shall be verified 0x02: The installation code does not have the CRC so shall be calculated
4 – (3 + Installation Code Lenght)	Installation Code	-	Defined by users
(4 + Installation Code Lenght) – (11 + Installation Code Lenght)	IEEE Address of Joining Device	-	Defined by users

NOTE:
The Trust Center is able to manage up to 5 Installation Codes.

Set Installation Code request (Other devices)

Offset	Name	Value	Description
0	FrameLength	3 + Installation Code Lenght	
1	Command	0x46	
2	Installation Code Size ID	0x00-0x03	Installation code size ID: **Speed Id** / Installation Code Size without CRC (Bytes) / Installation Code Size with CRC (Bytes) **0** 6 8 **1** 8 10 **2** 12 14 **3** 16 18
3	CRC Settings	0x00-0x02	0x00: The installation code has the CRC but it shall not be verified 0x01: The installation code has the CRC and it shall be verified 0x02: The installation code does not have the CRC so shall be calculated
4 – (3 + Installation Code Lenght)	Installation Code	-	Defined by users

Set Installation Code confirm

Offset	Name	Value	Description
0	FrameLength	2	
1	Command	0x47	
2	Status	SUCCESS, ERROR	Confirm status table

4.2.3.4. Network Address

This primitive is used to inquire as to the 16-bit address of the Remote Device based on its known IEEE address, and know which devices are associated at the Remote Device if the Request Type is set to Extended.

Nwk Addr request

Offset	Name	Value	Description
0	FrameLength	11	
1	Command	0xC0	
2-9	IEEEAddress (Little Endian)	-	Defined by users
10	RequestType	0x00-0x01	0x00: Single 0x01: Extended
11	Start index	0x00-0xFF	Defined by users

Nwk Addr confirm

Offset	Name	Value	Description
0	FrameLength	14+2* NumAssocDev	
1	Command	0xC1	
2	Status	SUCCESS, ZDP Enumeration	Confirm status table
3-10	IEEEAddrRemoteDev	-	IEEE address of the remote device
11-12	NWKAddrRemoteDev	-	NWK address of the remote device
13	NumAssocDev	- Present only for Extended request type	Number of Devices associated to the remote device
14	StartIndex	- Present only if NumAssocDev > 0	Index of the first associated device
15 – (14 + 2* NumAssocDev)	NWKAddrAssocDevList	-	List of the associated devices

4.2.3.5. IEEE Address

This primitive is used to inquire as to the IEEE address of the Remote Device based on its known NWK address, and know which devices are associated at the Remote Device if the Request Type is set to Extended.

IEEE Addr request

Offset	Name	Value	Description
0	FrameLength	5	
1	Command	0xC2	
2-3	NWKAddrOfInterest (Little Endian)	-	Defined by users
4	RequestType	0x00-0xFF	0x00: Single 0x01: Extended
5	Start index	0x00-0xFF	Defined by users

IEEE Addr confirm

Offset	Name	Value	Description
0	FrameLength	14+2* NumAssocDev	
1	Command	0xC3	
2	Status	SUCCESS, ZDP Enumeration	Confirm Status Table
3-10	IEEEAddrRemoteDev	-	IEEE address of the remote device
11-12	NWKAddrRemoteDev	-	NWK address of the remote device
13	NumAssocDev	- Present only for Extended request type	Number of Devices associated to the remote device
14	StartIndex	- Present only if NumAssocDev > 0	Index of the first associated device
15 – (14 + 2* NumAssocDev)	NWKAddrAssocDevList	-	List of the associated devices

4.2.3.6. Node Descriptor

This primitive is used to inquire as the node descriptor of the remote device.
The node descriptor contains information about the capabilities of the ZigBee®Node.

Node desc request

Offset	Name	Value	Description
0	FrameLength	3	
1	Command	0xC4	
2-3	NWKAddrOfInterest (Little Endian)	-	Defined by users

Node desc Confirm

Offset	Name	Value	Description
0	FrameLength	17	
1	Command	0xC5	
2	Status	SUCCESS, ZDP Enumeration	Confirm Status Table
3-4	DeviceAddress (Little Endian)	-	NWK address of the remote device
5-17	NodeDescriptor	-	(See the chapter 2.3.2.3 in the ZigBee®specification 053474r17)

4.2.3.7. Power Descriptor

This primitive is used to inquire as the node descriptor of the remote device.
The node power descriptor gives a dynamic indication of the power status of the Node.

Power desc request

Offset	Name	Value	Description
0	FrameLength	3	
1	Command	0xC6	
2-3	NWKAddrOfInterest (Little Endian)	-	Defined by users

Power desc confirm

Offset	Name	Value	Description
0	FrameLength	6	
1	Command	0xC7	
2	Status	SUCCESS, ZDP Enumeration	Confirm Status Table
3-4	DeviceAddress (Little Endian)	-	NWK address of the remote device
5-6	PowerDescriptor	-	(See the chapter 2.3.2.4 in the ZigBee® specification 053474r17)

4.2.3.8. Simple Descriptor

This primitive is used to inquire as the Simple descriptor on a specified Endpoint of the remote device.
The simple descriptor contains information specific to each endpoint contained in this remote node.

Simple desc request

Offset	Name	Value	Description
0	FrameLength	4	
1	Command	0xC8	
2-3	NWKAddrOfInterest (Little Endian)	-	Defined by users
4	Endpoint	0x00-0xFF	Defined by users

Simple desc confirm

Offset	Name	Value	Description
0	FrameLength	5+ Lenght	
1	Command	0xC9	
2	Status	SUCCESS, ZDP Enumeration	Confirm Status Table
3-4	DeviceAddress (Little Endian)	-	NWK address of the remote device
5	Length		Length of the Simple Descriptor
6-(5+Length)	SimpleDescriptor	-	(See the chapter 2.3.2.5 in the ZigBee® specification 053474r17)

4.2.3.9. Active Endpoint

This primitive is used to inquire for the list of Endpoints on the Remote device.
Active EP request

Offset	Name	Value	Description
0	FrameLength	3	
1	Command	0xCA	
2-3	NWKAddrOfInterest (Little Endian)	-	Defined by users

Active EP confirm

Offset	Name	Value	Description
0	FrameLength	5+ ActiveEPCount	
1	Command	0xCB	
2	Status	SUCCESS, ZDP Enumeration	Confirm Status Table
3-4	DeviceAddress (Little Endian)	-	NWK address of the remote device
5	ActiveEPCount		Number of active Endpoint
6-(5+ ActiveEPCount)	ActiveEPList	-	List of Endpoints

4.2.3.10. User Descriptor

These primitives are used to inquire or set the user descriptor of the remote. The user descriptor contains information that allows the user to identify the device using a user-friendly character string, such as “Bedroom TV” or “Stairs light”.

User desc request

Offset	Name	Value	Description
0	FrameLength	3	
1	Command	0xD0	
2-3	NWKAddrOfInterest (Little Endian)	-	Defined by users

User desc confirm

Offset	Name	Value	Description
0	FrameLength	5+ Lenght	
1	Command	0xD1	
2	Status	SUCCESS, ZDP Enumeration	Confirm Status Table
3-4	DeviceAddress (Little Endian)	-	NWK address of the remote device
5	Length		Length of the User Descriptor
6-(5+Length)	UserDescriptor	-	(See the chapter 2.3.2.7 in the ZigBee® specification 053474r17)

User desc set request

Offset	Name	Value	Description
0	FrameLength	4+ Lenght	
1	Command	0xE8	
2-3	DeviceAddress (Little Endian)	-	NWK address of the remote device
4	Length		Length of the User Descriptor
5-(4+Length)	UserDescriptor	-	16 Bytes Max (See the chapter 2.3.2.7 in the ZigBee® specification 053474r17)

User desc set confirm

Offset	Name	Value	Description
0	FrameLength	4	
1	Command	0xE9	
2	Status	SUCCESS, ZDP Enumeration	Confirm Status Table
3-4	DeviceAddress (Little Endian)	-	NWK address of the remote device

4.2.3.11. Match Descriptor

To find remote devices supporting a specific simple descriptor match criterion.
Match Desc request

Offset	Name	Value	Description
0	FrameLength	7+2*NumInClusters+2* NumOutClusters	
1	Command	0xCC	
2-3	DeviceAddress (Little Endian)	-	NWK address of the remote device
4-5	ProfileID(Little Endian)	-	Defined by users
6	NumInClusters	-	Defined by users
7-(6+2*NumInClusters)	InClusterList	-	Defined by users
7+2*NumInClusters	NumOutClusters	-	Defined by users
(8+2*NumInClusters)-(7+2*NumOutClusters+2*NumOutClusters)	OutClusterList	-	Defined by users

Match Desc confirm

Offset	Name	Value	Description
0	FrameLength	5+MatchLength	
1	Command	0xCD	
2	Status	SUCCESS, DEVICE_NOT_FOUND, INV_REQUESTTYPE, NO_DESCRIPTOR	
3-4	DeviceAddress (Little Endian)	-	
5	MatchLength	-	
6-(5+MatchLength)	MatchList	-	

4.2.3.12. System Server Discovery

Discover the location of a particular system server as defined.
System Server Discovery request

Offset	Name	Value	Description
0	FrameLength	3	
1	Command	0x20	
2-3	ServerMask (Little Endian)	-	Defined by users

System Server Discovery confirm

Offset	Name	Value	Description
0	FrameLength	7/13	
1	Command	0x21	
2	Status	SUCCESS	
3	SrcAddress Mode	-	Source Address of the Remote server
4-5 for Nwk address 4-11 for IEEE address	SrcAddress (Little Endian)	-	Address of the Remote server
6-7 12-13	ServerMask (Little Endian)	-	Sent by remote server

4.2.3.13. Bind/Unbind

These primitives are used to bind and unbind a remote device with another remote device.
Bind request

Offset	Name	Value	Description
0	FrameLength	16 or 22	
1	Command	0xD8	
2-9	SrcAddress (Little Endian)	-	Defined by users
10	SrcEndpoint	0x00-0xFF	Defined by users
11-12	ClusterID(Little Endian)	-	Defined by users
13	DstAddrMode	0x01 :Group Address 0x03: IEEE Address	Defined by users
14-15 (for group address) 14-21 (for 64-bit address)	DstAddress(Little Endian)	-	Defined by users
22 (for 64-bit address) 16 (for group address)	DstEndpoint	0x00-0xFF	Defined by users

Bind confirm

Offset	Name	Value	Description
0	FrameLength	2	
1	Command	0xD9	
2	Status	SUCCESS, ZDP Enumeration	Confirm Status Table

Unbind request

Offset	Name	Value	Description
0	FrameLength	16 or 22	
1	Command	0xDA	
2-9	SrcAddress (Little Endian)	-	Defined by users
10	SrcEndpoint	0x00-0xFF	Defined by users
11-12	ClusterID(Little Endian)	-	Defined by users
13	DstAddrMode	0x01 :Group Address 0x03: IEEE Address	Defined by users
14-15 (for group address) 14-21 (for 64-bit address)	DstAddress(Little Endian)	-	Defined by users
22 (for 64-bit address) 16 (for group address)	DstEndpoint	0x00-0xFF	Defined by users

Unbind confirm

Offset	Name	Value	Description
0	FrameLength	2	
1	Command	0xDB	
2	Status	SUCCESS, ZDP Enumeration	Confirm Status Table

4.2.3.14. End Device Bind request

These primitives are used to do an End device bind.
End Device Bind request

Offset	Name	Value	Description
0	FrameLength	6 +2*NumInClusters+2*NumOutClusters	
1	Command	0xD6	
2	SrcEndpoint	0x00-0xFF	Defined by users
3-4	ProfileID(Little Endian)	-	Defined by users
5	NumInClusters	-	Defined by users
6-(5+2*NumInClusters)	InClusterList(Little Endian)	-	Defined by users
6+2*NumInClusters	NumOutClusters	0x00-0xFF	Defined by users
(7+2*NumInClusters)-(6+2*NumInClusters +2*NumOutClusters)	OutClusterList(Little Endian)	-	Defined by users

End Device Bind confirm

Offset	Name	Value	Description
0	FrameLength	2	
1	Command	0xD7	
2	Status	SUCCESS, NOT_SUPPORTED, INVALID_EP, TIMEOUT, NO_MATCH	Confirm Status Table

4.2.3.15. Group

These primitives are used to add or remove group to an endpoint.

Add Group Request

Offset	Name	Value	Description
0	FrameLength	4	
1	Command	0xAB	
2-3	GroupAddress (Little Endian)	-	Defined by users
4	EndPoint	-	Defined by users

Add Group Confirm

Offset	Name	Value	Description
0	FrameLength	5	
1	Command	0xAC	
2	Status	SUCCESS, INVALID_PARAMETER, TABLE_FULL	Confirm Status Table
3-4	GroupAddress (Little Endian)	-	
5	EndPoint	-	

Remove Group Request

Offset	Name	Value	Description
0	FrameLength	4	
1	Command	0xAD	
2-3	GroupAddress (Little Endian)	-	Defined by users
4	EndPoint	-	Defined by users

Remove Group Confirm

Offset	Name	Value	Description
0	FrameLength	5	
1	Command	0xAE	
2	Status	SUCCESS, INVALID_PARAMETER	Confirm Status Table
3-4	GroupAddress (Little Endian)	-	
5	EndPoint	-	

Remove All Groups Request

Offset	Name	Value	Description
0	FrameLength	2	
1	Command	0xAF	
2	EndPoint	-	Defined by users

Remove All Groups Confirm

Offset	Name	Value	Description
0	FrameLength	3	
1	Command	0xBE	
2	Status	SUCCESS, INVALID_PARAMETER	Confirm Status Table
3	EndPoint	-	

4.2.3.16. Request Key

These primitives are used to request a Key.

Request Key request

Offset	Name	Value	Description
0	FrameLength	10 or 18	
1	Command	0xBA	
2-9	DstAddress(Little Endian)	-	Trust Center (Coordinator) IEEE address
10	KeyType	-	Only link key (0x02) is supported
11-18	PartnerAddress	-	If Key Type = 0x02

NOTE:

if security is not enabled the Request Key request is not sent.
if Key Type is link key (0x02) could happened the Trust Center does not have address information about the modules participating in key exchange then any Transport Key will not be received. In this case to receive an end to end link key a new Request Key request has to be sent.

4.2.3.17. Management Permit Joining

This primitive is used to request that a remote device allow or disallow association.

Mgmt permit joining request

Offset	Name	Value	Description
0	FrameLength	5	
1	Command	0xEA	
2-3	DstAddress (Little Endian)	-	Defined by users
4	PermitDuration	0x00-0xFF	Defined by users 0x00: remote device disallow association 0xFF: remote device allow association 0x01-0xFE: time during which remote device allow association
5	TC_Significance	0x00-0x01	Defined by users 0x00: No Trust Centre relevance 0x01: Affects Trust Centre authentication. If Duration is 0xFF, it allows authentication. If Duration is 0x00, then no device will be authenticated.

Mgmt permit joining confirm

Offset	Name	Value	Description
0	FrameLength	2	
1	Command	0xEB	
2	Status	SUCCESS, ZDP Enumeration	Confirm Status Table

4.2.3.18. Management Leave

This primitive is used to ask at a remote device to leave the network.

Mgmt leave request

Offset	Name	Value	Description
0	FrameLength	12	
1	Command	0xE4	
2-3	DestNwkAddress (Little Endian)	-	Defined by users
2-9	DeviceAddress (Little Endian)	-	Defined by users. If set to 0x0000000000000000 the device with DestNwkAddress will leave the network
10	Bit0: Rejoin Bit1: RemoveChildren Bit2- Bit7: Reserved	-	Defined by users

Mgmt leave response

Offset	Name	Value	Description
0	FrameLength	2	
1	Command	0xE5	
2	Status	SUCCESS, ZDP Enumeration	Confirm Status Table

Mgmt leave indication

Offset	Name	Value	Description
0	FrameLength	10	
1	Command	0x7B	
2-9	DeviceAddress (Little Endian)		
10	Rejoin	-	

4.2.3.19. Management Nwk Update

This primitive is used to update or request information from device on network conditions in the local operating environment.

Mgmt Nwk Update request

Offset	Name	Value	Description
0	FrameLength	11	
1	Command	0x18	
2-5	Scan Channels(Little Endian)	-	Defined by users
6	Scan Duration	-	in seconds. Defined by users
7	ScanCount	-	Defined by users
8-9	Nwk Manager Address(Little Endian)	-	Defined by users
10-11	DstAddress (Little Endian)	-	Defined by users

Mgmt Nwk Update notify

Offset	Name	Value	Description
0	FrameLength		
1	Command	0x19	
2	Status	SUCCESS, NOT_SUPPORTED	
3-6	Scanned Channels	-	
7-8	Total Tx	-	
9-10	Tx Failures	-	
11	ScanListCount	-	
12+ ScanListCount	ScanList	-	

4.2.3.20. List of Binding

This primitive is used to ask at a remote device to retrieve the contents of the Binding table.

Mgmt bind request

Offset	Name	Value	Description
0	FrameLength	4	
1	Command	0xE2	
2-3	DestNwkAddress (Little Endian)	-	Defined by users
4	StartIndex	0x00-0xFF	Defined by users . Starting Index for the requested elements of the Binding Table

Mgmt bind confirm

Offset	Name	Value	Description
0	FrameLength	-	
1	Command	0xE3	
2	Status	SUCCESS, ZDP Enumeration	Confirm Status Table
3	BindingTableEntries	-	Total Number of binding in the Remote device
4	StartIndex	-	Starting index within the Binding Table to begin reporting for the BindingTableList.
5	BindingTableListCount	-	Number of Binding included within BindingTableList
6-	BindingTableList	-	Present only if BindingTableListCount >0. List Of Binding Descriptor (See the Table 2.129 and 2.130 chapter 2.4.4.3.4 in the ZigBee® specification 053474r17)

4.2.3.21. Application Frame Direct

This primitive is used to communicate with an application of a remote device.
AF Direct request

Offset	Name	Value	Description
0	FrameLength	10+afdulenght	
1	Command	0xF3	
2-3	DstAddress (Little Endian)	-	Defined by users
4	DstEndPoint	0x00-0xFF	Defined by users
5	SrcEndPoint	0x00-0xFF	Defined by users
6-7	ClusterID(Little Endian)	-	Defined by users
8	afduLenght	-	Defined by users
9-(8+afduLenght)	Afdu	-	
9+afduLenght	Txoption	Bit 0: encrypt request Bit 2: APS Ack Bit 3: Frag	Defined by users
10+ afduLenght	BroadcastRadius	-	Defined by users

NOTE:

Maximum afdu Lenght:
With fragmentation 128 bytes
Without fragmentation and no security 99 bytes
Without fragmentation and with network security 81 bytes.

NOTE:

if fragmentation is enabled and the receiver or the sender are sleeping end device the transmission could does not succeeded for timeout of fragmentation algorithm.

AF Direct indication

Offset	Name	Value	Description
0	FrameLength	14+afdulenght	
1	Command	0xF5	
2	DstEndPoint	0x00-0xFF	
3-4	SrcAddress (Little Endian)	-	
5	SrcEndPoint	-	
6-7	ClusterID(Little Endian)	-	
8	afduLenght	-	
9-(8+afduLenght)	Afdu	-	
9 +afduLenght	WasBroadcast	-	
10 + afduLenght	SecurityStatus	0xAE,0xAC,0xAB	0xAE : unsecured packet 0xAC : secured with network key 0xAB : secured with link key
(11-12)+ afduLenght	Last hop source	0x0000-0xFFF0	Source address of last hop
13 + afduLenght	RSSI	0x00-0xFF	RSSI of the frame. Signed char
14 + afduLenght	Correlation	50-110	Correlation indication of frame. Can be seen as a Chip error rate. 50 : bad reception 110 : excellent reception

AF Direct Confirm

Offset	Name	Value	Description
0	FrameLength	6	
1	Command	0xF4	
2-3	DstAddress	-	
4	DstEndPoint	-	
5	SrcEndPoint	-	
6	Status	SUCCESS or others	

4.2.3.22. Application Frame Indirect

This primitive is used to communicate with a bind application of a remote device.
AF Indirect request

Offset	Name	Value	Description
0	FrameLength	7+afdulenght	
1	Command	0xF0	
2	SrcEndPoint	0x00-0xFF	Defined by users
3-4	ClusterID(Little Endian)	-	Defined by users
5	afduLenght	-	Defined by users
6-(5+afduLenght)	Afdu	-	
6+afduLenght	Txoption	Bit 0: encrypt request Bit 2: APS Ack Bit 3: Frag	Defined by users
7+ afduLenght	BroadcastRadius	-	Defined by users

NOTE:

if fragmentation is enabled and the receiver or the sender are sleeping end device the transmission could does not succeeded for timeout of fragmentation algorithm.

NOTE:

if the binding table has entries with Group address an AF Group Confirm will be received instead an AF Indirect Confirm
AF Indirect Confirm

Offset	Name	Value	Description
0	FrameLength	4	
1	Command	0xF1	
2	DstEndPoint	-	
3	SrcEndPoint	-	
4	Status	SUCCESS or others	

4.2.3.23. Poll for Indirect reception

This primitive is used by sleeping end devices to retrieve waiting frames in device's parent. RFD only.
Poll request

Offset	Name	Value	Description
0	FrameLength	1	
1	Command	0x7C	

Poll Confirm

Offset	Name	Value	Description
0	FrameLength	2	
1	Command	0x7D	
2	Status	-	0x00 for success, or error code

4.2.3.24. Application Frame Group

This primitive is used to communicate with a group of remote application
AF Group request

Offset	Name	Value	Description
0	FrameLength	9+afdulenght	
1	Command	0xF9	
2-3	GroupAddress (Little Endian)	-	Defined by users
4	SrcEndPoint	0x00-0xFF	Defined by users
5-6	ClusterID(Little Endian)	-	Defined by users
7	afduLenght	-	Defined by users
8-(7+afduLenght)	Afdu	-	
8+afduLenght	Txoption	Bit 0: encrypt request Bit 2: APS Ack	Defined by users
9+ afduLenght	BroadcastRadius	-	Defined by users

AF Group indication

Offset	Name	Value	Description
0	FrameLength	11+afdulenght	
1	Command	0xFB	
2-3	GroupAddress (Little Endian)	-	
4	DstEndPoint	0x00-0xFF	
5-6	SrcAddress(Little Endian)	-	
7	SrcEndPoint	0x00-0xFF	
8-9	ClusterID(Little Endian)	-	
10	afduLenght	-	
11->(10+afduLenght)	Afdu	-	
11+afduLenght	SecurityStatus	-	

AF Group Confirm

Offset	Name	Value	Description
0	FrameLength	5	
1	Command	0xFA	
2-3	GroupAddress	-	
4	SrcEndPoint	-	
5	Status	SUCCESS or others	

4.2.4. Others

4.2.4.1. Boot loader

This primitive is used to jump in the bootloader to programming a new firmware.
Bootloader request

Offset	Name	Value	Description
0	FrameLength	1	
1	Command	0xFF	

Bootloader Confirm

Offset	Name	Value	Description
0	FrameLength	2	
1	Command	0xFF	
2	Status	0x00	Success

4.2.4.2. Reset

This primitive is used to reset device
Reset request

Offset	Name	Value	Description
0	FrameLength	2	
1	Command	0x10	
2	Attributes	0x00-0x01	0x00: Hard reset (default value) 0x01: Soft reset (disconnect of the network)

Reset confirm

Offset	Name	Value	Description
0	FrameLength	2	
1	Command	0x11	
2	Status	SUCCESS	Confirm Status Table

The table below explains what are the attributes reset (✔) during a specific reset and what is the default value to which they are set.

Attribute	Name	Soft reset	Hard reset
0x6F	IEEE Address	✘	✘
0x11	Join Period PHASE 1	✘	✔ (0x3C)
0x12	Join Period PHASE 2	✘	✔ (0xE10)
0x13	Join Retries PHASE 1	✘	✔ (15)
0x14	Jitter Phase 1	✘	✔ (15)
0x15	Jitter Phase 2	✘	✔ (30)
0x1A	Disable Compiled Simple Descriptors	✘	✔ (0)
0x1B	Read/Write Simple Descriptor	✘	✔
0x52	RxOnWhenIdle	✘	✔ (0x01)
0x56	Sleeping Time	✘	✔ (0x03)
0x57	Rejoin Type	✘	✔ (0x07)
0x58	Rejoin Interval	✘	✔ (60)
0x59	Max Rejoin Interval	✘	✔ (900)
0x5A	Max Rejoin Retries first Phase	✘	✔ (0xFF)
0x5B	Secure Rejoin Retries	✘	✔ (1)
0x5C	Rejoin Retries	✘	✔ (1)
0x01	Radio Channel	✘	✔ (0xFFFF)
0x00	Current Channel	✔ (0x0B)	✔ (0x0B)
0x04	Version Stack	✘	✘
0x05	Version Bootloader	✘	✘
0x0A	Version Application	✘	✘
0x0C	Serial speed	✘	✔ (0x07)
0xC4	ExtendedPanID	✘	✔ (0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00)
0xCA	TrustCenter	✘	✔ (0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00)
0x99	USB device	✘	✘
0x06	Type of device	✘	✘
0x07	Is associated	✔ (0x00)	✔ (0x00)
0x96	Nwk address	✔ (0xFF,0xFF)	✔ (0xFF,0xFF)
0xC9	Fragmentation Inter Frame Delay	✘	✔ (0x64)
0xCD	Fragmentation Window Size	✘	✔ (0x03)

0xD0	End Device Binding Timeout	✗	✓ (0x14)
0xA3	Use Security	✗	✓ (0x00)
0xE4	HasPreconfiguredNwkKey	✗	✓ (0x00)
0xA4	Nwk Key	✗	✓ (0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF, 0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF)
0xE5	HasPreconfiguredLinkKey	✗	✓ (0x00)
0xA5	Link Key(16) + IEEE address(8)	✗	✓
0x02	Output power Attenuation	✗	✓ (0x01)
0x50	PAN Id	✓ (0xFF,0xFF)	✓ (0xFF,0xFF)

Hard reset takes about 500ms and soft reset about 150ms.

4.3. How to create a network

There are two phases in network creation:

- A coordinator forms a network
- Devices (Routers, End Devices or Sleeping End Devices) join the network formed by the coordinator.

The network can be formed using or not security.

The coordinator decides which level of security shall be used on the network.

Three levels of security can be chosen

1. No security
2. Network security
3. Network security with network key exchanged through Trust Center Link Key

4.3.1. Form a network without security

The steps to form a network are listed and described below. All the commands shall be sent to the coordinator.

1. Enter in command mode:

	Command
Hex	2B 2B 2B
ASCII	"+++"
Direction	Host -> ZEx1 module

Expected packet sent by the ZEx1 module:

	Command
Hex	0D
Direction	ZEx1 module -> Host

2. Reset the module:

	Command
Hex	02 10 00
Description	10 : Reset Request
	00 : Hard reset
Direction	Host -> ZEx1 module

Expected packet sent by the ZEx1 module:

	Command
Hex	02 11 00
Description	11 : Reset Confirm
	00 : Success status
Direction	ZEx1 module -> Host

3. Enter in command mode:

	Command
Hex	2B 2B 2B
ASCII	"+++"
Direction	Host -> ZEx1 module

Expected packet sent by the ZEx1 module:

	Command
Hex	0D
Direction	ZEx1 module -> Host

NOTE:
After a power on, an hardware reset or a software reset (Hard or Soft) the module is set automatically in data mode so "+++" shall be sent to the module to switch to command mode

4. Set channel mask (Optional):

	Command
Hex	05 12 01 02 00 10
Description	12 : Set Request
	01 : Channel mask attribute ID
	02 : channel mask length
	00 10 : use only channel 15
Direction	Host -> ZEx1 module

NOTE:
If channel mask is not set the default value, FF FF, is used and the coordinator will take in account all the 802.15.4 channels on 2.4GHz band during the formation process

Expected packet sent by the ZEx1 module:

	Command
Hex	03 13 00 01
Description	13 : Set Confirm
	00 : Success status
	01 : Channel Mask attribute ID
Direction	ZEx1 module -> Host

5. Set Extended PAN ID (Optional):

	Command
Hex	0B 12 C4 08 00 00 00 00 11 22 33 44
Description	12 : Set Request
	C4 : Extended PAN ID Attribute ID
	08 : Extended PAN ID Length
	00 00 00 00 11 22 33 44: extended Pan ID
Direction	Host -> ZEx1 module

NOTE:
If extended PAN ID is not set the default value is used 00 00 00 00 00 00 00 00. In this case the coordinator will use its IEEE address as Extended Pan ID.

Expected packet sent by the ZEx1 module:

	Command
Hex	03 13 00 C4
Description	13 : Set Confirm
	00 : Success status
	C4 : Extended PAN ID Attribute ID
Direction	ZEx1 module -> Host

6. Start the network:

	Command
Hex	01 16
Description	16 : Start Request
Direction	Host -> ZEx1 module

NOTE:
During network formation the coordinator verifies which is the best channel (among the ones enabled with the channel mask) to create the network.

Expected packet sent by the ZEx1 module:

	Command
Hex	02 17 00
Description	17 : Start Confirm
	00 : Success status
Direction	ZEx1 module -> Host

NOTE:
If a hard or a soft reset is sent to the coordinator the network shall be formed again.

4.3.2. Form a network with network security

The steps to form a network are listed and described below. All the commands shall be sent to the coordinator.

1. All the steps from 1 to 5 described in section 4.3.1.
2. Enable Security:

	Command
Hex	04 12 A3 01 01
Description	12 : Set Request
	A3 : Security Enable attribute ID
	01 : Security Enable length
	01 : use security
Direction	Host -> ZEx1 module

Expected packet sent by the ZEx1 module:

	Command
Hex	03 13 00 A3
Description	13 : Set Confirm
	00 : Success status
	A3 : Security Enable attribute ID
Direction	ZEx1 module -> Host

3. Set network key:

	Command
Hex	13 12 A4 10 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
Description	12 : Set Request
	A4 :Network key attribute ID
	10 : Network key length
	00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F: Network key
Direction	Host -> ZEx1 module

NOTE:
If a network key is not set the default one is used FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF.

Expected packet sent by the ZEx1 module:

	Command
Hex	03 13 00 A4
Description	13 : Set Confirm
	00 : Success status
	A4 :Network key attribute ID
Direction	ZEx1 module -> Host

NOTE:
The network key will be used to encrypt all the messages (Network layer payload) exchanged on the network.

4. Start the network:

	Command
Hex	01 16
Description	16 : Start Request
Direction	Host -> ZEx1 module

NOTE:
During network formation the coordinator verifies which is the best channel (among the ones enabled with the channel mask) to create the network.

Expected packet sent by the ZEx1 module:

	Command
Hex	02 17 00
Description	17 : Start Confirm
	00 : Success status
Direction	ZEx1 module -> Host

NOTE:
If a hard or a soft reset is sent to the coordinator the network shall be formed again.

4.3.3. Form a network with network security using Trust Center Link Key

The steps to form a network are listed and described below. All the commands shall be sent to the coordinator.

1. All the steps from 1 to 3 described in section 4.3.2.
2. Enable the Trust Center Link Key Mechanism:

	Command
Hex	04 12 E5 01 01
Description	12 : Set Request
	E5 : Has Preconfigured Trust Center Link Key attribute ID
	01 : Has Preconfigured Trust Center Link Key length
	01 : use Trust Center Link Key
Direction	Host -> ZEx1 module

Expected packet sent by the ZEx1 module:

	Command
Hex	03 13 00 E5
Description	13 : Set Confirm
	00 : Success status
	E5 : Has Preconfigured Trust Center Link Key attribute ID
Direction	ZEx1 module -> Host

NOTE:
The coordinator shall have a trust center link key for every device that joins the network. The coordinator is able to store up to 5 different trust center link keys. If more than 5 devices join the network and the security with trust center link key is enabled, the same Trust center link key shall be used for all the devices.

The trust center link key for a specific device (or the generic if only one is used) can be set at any time before joining the device.

How to set the trust center link key on both side (Trust center and joining device) is explained in joining process description.

5. Start the network:

	Command
Hex	01 16
Description	16 : Start Request
Direction	Host -> ZEx1 module

NOTE:
During network formation the coordinator verifies which is the best channel (among the ones enabled with the channel mask) to create the network.

Expected packet sent by the ZEx1 module:

	Command
Hex	02 17 00
Description	17 : Start Confirm
	00 : Success status
Direction	ZEx1 module -> Host

NOTE:
If a hard or a soft reset is sent to the coordinator the network shall be formed again.

4.3.4. Join a network without security

The steps to join a network are listed and described below. All the commands shall be sent to the joining device (Router, End Device or Sleeping End Device).

1. Enter in command mode:

	Command
Hex	2B 2B 2B
ASCII	"+++"
Direction	Host -> ZEx1 module

Expected packet sent by the ZEx1 module:

	Command
Hex	0D
Direction	ZEx1 module -> Host

2. Reset the module:

	Command
Hex	02 10 00
Description	10 : Reset Request
	00 : Hard reset
Direction	Host -> ZEx1 module

Expected packet sent by the ZEx1 module:

	Command
Hex	02 11 00
Description	11 : Reset Confirm
	00 : Success status
Direction	ZEx1 module -> Host

3. Enter in command mode:

	Command
Hex	2B 2B 2B
ASCII	"+++"
Direction	Host -> ZEx1 module

Expected packet sent by the ZEx1 module:

	Command
Hex	0D
Direction	ZEx1 module -> Host

NOTE:
After a power on, an hardware reset or a software reset (Hard or Soft) the module is set automatically in data mode so "+++" shall be sent to the module to switch to command mode

4. Set channel mask (Optional):

	Command
Hex	05 12 01 02 00 10
Description	12 : Set Request
	01 : Channel mask attribute ID
	02 : channel mask length
	00 10 : use only channel 15
Direction	Host -> ZEx1 module

NOTE:
If channel mask is not set the default value FF FF is used and the joining device will search an available network in all the 802.15.4 channels on 2.4GHz band.
Pay attention to enable at least the channels enabled on the coordinator.

Expected packet sent by the ZEx1 module:

	Command
Hex	03 13 00 01
Description	13 : Set Confirm
	00 : Success status
	01 : Channel Mask attribute ID
Direction	ZEx1 module -> Host

5. Set extended PAN ID (Optional):

	Command
Hex	0B 12 C4 08 00 00 00 00 11 22 33 44
Description	12 : Set Request
	C4 : Extended PAN ID Attribute ID
	08 : Extended PAN ID Length
	00 00 00 00 11 22 33 44: extended Pan ID
Direction	Host -> ZEx1 module

NOTE:
If extended PAN ID is not set the default value is used 00 00 00 00 00 00 00 00) and the device will join the first available network.
If an extended PAN ID is set the device will join only a network with the specified extended PAN ID.

Expected packet sent by the ZEx1 module:

	Command
Hex	03 13 00 C4
Description	13 : Set Confirm
	00 : Success status
	C4 : Extended PAN ID Attribute ID
Direction	ZEx1 module -> Host

6. Set sleeping feature (Optional and available only on End Device):

	Command
Hex	04 12 52 01 00
Description	12 : Set Request
	52 : Rx On When Idle Attribute ID
	01 : Rx On When Idle Length
	00 : The device is sleeping
Direction	Host -> ZEx1 module

NOTE:
If Rx On When Idle is not modified the default value 01 is used and the device will be an always awake device.
By definition Routers and Coordinator are always awake.

Expected packet sent by the ZEx1 module:

	Command
Hex	03 13 00 52
Description	13 : Set Confirm
	00 : Success status
	52 : Rx On When Idle Attribute ID
Direction	ZEx1 module -> Host

7. Set sleeping feature (Optional, available only on End Device and managed only for sleeping device):

	Command
Hex	07 12 56 04 00 00 00 0A
Description	12 : Set Request
	56 : Sleeping Time Attribute ID
	04 : Sleeping Time ID Length
	00 00 00 0A : The device will awake and poll the parent every 10 seconds
Direction	Host -> ZEx1 module

NOTE:
If Sleeping Time is not modified the default value 00 00 00 03 is used and the device will awake and poll the parent every 3 seconds.
This attribute has effect only on sleeping device.

Expected packet sent by the ZEx1 module:

	Command
Hex	03 13 00 56
Description	13 : Set Confirm
	00 : Success status
	56 : Sleeping Time Attribute ID
Direction	ZEx1 module -> Host

8. Start the network:

	Command
Hex	01 16
Description	16 : Start Request
Direction	Host -> ZEx1 module

Expected packet sent by the ZEx1 module:

	Command
Hex	02 17 00
Description	17 : Start Confirm
	00 : Success status
Direction	ZEx1 module -> Host

NOTE:
If a hard or a soft reset is sent to the Device, the join procedure shall be repeated.

NOTE:
If the joining device is a sleeping device and the joining process succeeded the device will awake and poll the parent every Sleeping Time. When the module sleeps the external host cannot communicate with it. Before communicating with the sleeping device the external host shall set low the CTS pin of the serial link to awake the device.

4.3.5. Join a network with network security

The steps to join a network are listed and described below. All the commands shall be sent to the joining device (Router, End Device or Sleeping End Device).

1. All the steps from 1 to 7 described in section 4.3.4.
2. Enable Security:

	Command
Hex	04 12 A3 01 01
Description	12 : Set Request
	A3 : Security Enable attribute ID
	01 : Security Enable length
	01 : use security
Direction	Host -> ZEx1 module

Expected packet sent by the ZEx1 module:

	Command
Hex	03 13 00 A3
Description	13 : Set Confirm
	00 : Success status
	A3 : Security Enable attribute ID
Direction	ZEx1 module -> Host

3. Set HasPreconfiguredNwkKey attribute (Optional):

	Command
Hex	04 12 E4 01 01
Description	12 : Set Request
	E4 : Has Preconfigured Nwk Key attribute ID
	01 : Has Preconfigured Nwk Key length
	01 : the device has a preconfigured network key
Direction	Host -> ZEx1 module

NOTE:
If the joining device has a preconfigured network key, the network key set on the coordinator shall be set on the joining device. This mechanism is more secure than the one without preconfigured network key because the coordinator will not send on clear the network key during association process.

Expected packet sent by the ZEx1 module:

	Command
Hex	03 13 00 E4
Description	13 : Set Confirm
	00 : Success status
	E4 : Has Preconfigured Nwk Key attribute ID
Direction	ZEx1 module -> Host

4. Set network key (Optional, needed only if Has Preconfigured Nwk Key is set to 1):

	Command
Hex	13 12 A4 10 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
Description	12 : Set Request
	A4 :Network key attribute ID
	10 : Network key length
	00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F: Network key
Direction	Host -> ZEx1 module

NOTE:
If a network key is not set the default one is used FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF.

The network key set on the coordinator shall be set also on the joining device

Expected packet sent by the ZEx1 module:

	Command
Hex	03 13 00 A4
Description	13 : Set Confirm
	00 : Success status
	A4 :Network key attribute ID
Direction	ZEx1 module -> Host

NOTE:
The network key will be used to encrypt all the messages (Network layer payload) exchanged on the network.

5. Start the network:

	Command
Hex	01 16
Description	16 : Start Request
Direction	Host -> ZEx1 module

Expected packet sent by the ZEx1 module:

	Command
Hex	02 17 00
Description	17 : Start Confirm
	00 : Success status
Direction	ZEx1 module -> Host

NOTE:
If a hard or a soft reset is sent to the Device, the join procedure shall be repeated.

NOTE:
If the joining device is a sleeping device and the joining process succeeded the device will awake and poll the parent every Sleep Time. When the module sleeps the external host cannot communicate with it. Before communicating with the sleeping device the external host shall set low the CTS pin of the serial link to awake the device.

4.3.6. Join a network with network security using Trust Center Link Key

The steps to join a network are listed and described below.

1. All the steps from 1 to 4 described in section 4.3.6.
2. Set on the coordinator a Trust Center link key related to the joining device. The Trust Center Link Key can be set using two mechanisms: setting directly the Trust Center link key or using the installation code.

a. Setting directly the Trust Center Link Key:

	Command
Hex	1B 12 A5 18 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 01 00 00 00 00 4F 15 00
Description	12 : Set Request
	A5 :Trust Center Link Key attribute ID
	18 : Trust Center Link Key length
	00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F: Trust Center link key key 01 00 00 00 00 4F 15 00 : IEEE address in Little Endian
Direction	Host -> ZEx1 module

NOTE:
If the IEEE address is set to 00 00 00 00 00 00 00 00 the coordinator will manage all the devices with the same Trust Center link key and it shall be set only once.
The coordinator can manage only 5 different Trust Center Link keys.

Expected packet sent by the ZEx1 module:

	Command
Hex	03 13 00 A5
Description	13 : Set Confirm
	00 : Success status
	A5 : Trust Center Link Key attribute ID
Direction	ZEx1 module -> Host

b. Using Installation code:

In this case a specific algorithm is used to calculate a trust center link key starting from a specific code (installation code). This mechanism has been defined for some application profiles (e.g. Smart Energy).

<table>
<tr><th></th><th>Command</th></tr>
<tr><td>Hex</td><td>13 46 00 01 83 FE D3 40 70 93 2B 70 01 00 00 00 00 4F 15 00</td></tr>
<tr><td rowspan="5">Description</td><td>46 : Set Installation Code</td></tr>
<tr><td>00 : Installation Code Size ID
<table>
<tr><th>Size Id</th><th>Installation Code Size without CRC (Bytes)</th><th>Installation Code Size with CRC (Bytes)</th></tr>
<tr><td>0</td><td>6</td><td>8</td></tr>
</table></td></tr>
<tr><td>01 : The installation code has the CRC but it shall be verified</td></tr>
<tr><td>83 FE D3 40 70 93: Installation code
2B 70: CRC</td></tr>
<tr><td>01 00 00 00 00 4F 15 00 : IEEE address in Little Endian</td></tr>
<tr><td>Direction</td><td>Host -> ZEx1 module</td></tr>
</table>

NOTE:
If the IEEE address is set to 00 00 00 00 00 00 00 00 the coordinator will manage all the devices with the same Trust Center link key generated starting from the Installation code and it shall be set only once.
The coordinator can manage only 5 different Installation code.

Expected packet sent by the ZEx1 module:

<table>
<tr><th></th><th>Command</th></tr>
<tr><td>Hex</td><td>02 47 00</td></tr>
<tr><td rowspan="2">Description</td><td>47 : Set Installation Code Confirm</td></tr>
<tr><td>00 : Success status</td></tr>
<tr><td>Direction</td><td>ZEx1 module -> Host</td></tr>
</table>

3. Set on the joining device the Trust Center link key set to on the coordinator. The Trust Center Link Key can be set using two mechanisms: setting directly the Trust Center link key or using the installation code.

   a. Setting directly the Trust Center Link Key:

	Command
Hex	1B 12 A5 18 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 00 00 00 00 00 00 00 00
Description	12 : Set Request
	A5 :Trust Center Link Key attribute ID
	18 : Trust Center Link Key length
	00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F: Trust Center link key key 00 00 00 00 00 00 00 00 : IEEE address in Little Endian, automatically select the coordinator with the correct Trust Center link key
Direction	Host -> ZEx1 module

NOTE:
If the IEEE address is set to a value different to 00 00 00 00 00 00 00 00 the device will try to join only to the specified device.

Expected packet sent by the ZEx1 module:

	Command
Hex	03 13 00 A5
Description	13 : Set Confirm
	00 : Success status
	A5 : Trust Center Link Key attribute ID
Direction	ZEx1 module -> Host

b. Using Installation code:

In this case a specific algorithm is used to calculate a trust center link key starting from a specific code (installation code). This mechanism has been defined for some application profiles (e.g. Smart Energy).

<table>
<tr><th></th><th>Command</th></tr>
<tr><td>Hex</td><td>0B 46 00 01 83 FE D3 40 70 93 2B 70</td></tr>
<tr><td rowspan="4">Description</td><td>46 : Set Installation Code</td></tr>
<tr><td>00 : Installation Code Size ID
<table>
<tr><th>Size Id</th><th>Installation Code Size without CRC (Bytes)</th><th>Installation Code Size with CRC (Bytes)</th></tr>
<tr><td>0</td><td>6</td><td>8</td></tr>
</table></td></tr>
<tr><td>01 : The installation code has the CRC but it shall be verified</td></tr>
<tr><td>83 FE D3 40 70 93: Installation code
2B 70: CRC</td></tr>
<tr><td>Direction</td><td>Host -> ZEx1 module</td></tr>
</table>

NOTE:
The device will automatically select the coordinator with the correct installation code (so the correct Trust Center link key) .

Expected packet sent by the ZEx1 module:

<table>
<tr><th></th><th>Command</th></tr>
<tr><td>Hex</td><td>02 47 00</td></tr>
<tr><td rowspan="2">Description</td><td>47 : Set Installation Code Confirm</td></tr>
<tr><td>00 : Success status</td></tr>
<tr><td>Direction</td><td>ZEx1 module -> Host</td></tr>
</table>

4. Start the network:

	Command
Hex	01 16
Description	16 : Start Request
Direction	Host -> ZEx1 module

Expected packet sent by the ZEx1 module:

	Command
Hex	02 17 00
Description	17 : Start Confirm
	00 : Success status
Direction	ZEx1 module -> Host

NOTE:
If a hard or a soft reset is sent to the Device, the join procedure shall be repeated.

NOTE:
If the joining device is a sleeping device and the joining process succeeded the device will awake and poll the parent every Sleep Time. When the module sleeps the external host cannot communicate with it. Before communicating with the sleeping device the external host shall set low the CTS pin of the serial link to awake the device.

4.4. How to permit joining

By default all the devices that can be parent (Coordinator and routers) permit joining. To avoid a device joins a specific router or the coordinator the Management Permit Joining command can be used For example, in order to avoid association to the coordinator after the network creation, it is possible to sent to the serial link of the Coordinator the command below:

	Command
Hex	05 EA 00 00 00 00
Description	EA: Management Permit Joining
	00 00 : Destination Network Address
	00 : Not permit association 00 : It does not affect authentication
Direction	Host -> ZEx1 module

Expected packet sent by the ZEx1 module:

	Command
Hex	02 EB 00
Description	EB : Management Permit Joining Confirm
	00 : Success status
Direction	ZEx1 module -> Host

NOTE:
From the coordinator the Management permit join can be sent remotely. The correct network address shall be set and the indicated device shall be joined to the network

4.5. How to exchange data

Once a device is joined to a network it can communicate remotely with all the other devices participating to the same network. The simplest way to do it is using Application Frame Direct request.

To avoid interference with specific behaviour of Democase functionalities the serial end point (0x01) can be used with a cluster ID different from 0x0060 (Serial Data) or a new Simple descriptor can be registered and used.

An example of unicast Application Frame Direct request from coordinator to a joined device (for example with address 0x5566) using the serial end point (0x01) and a dummy cluster ID 0x0302 is provided below :

	Command
Hex	15 F3 66 55 01 01 02 03 0B 48 65 6C 6C 6F 20 57 6F 72 6C 64 00 00
Description	F3 : Application Frame Direct Request
	66 55 : Destination network Address (Note it is in little endian)
	01 : Destination End Point
	01 : Source End Point
	02 03 : Cluster ID (it is in little endian)
	0B : Payload length
	48 65 6C 6C 6F 20 57 6F 72 6C 64 : “Hello World”
	00 : no transmission option
	00 : Maximum number of hops
Direction	Host -> ZEx1 module

NOTE:
If the user prefers using application acknowledgement the transmission option shall be set to 0x04

The device with address 0x5566 will receive the message and will forward it through the serial link to the external host. The message that will be sent to the external host is an Application Frame Direct Indication:

	Command
Hex	19 F5 01 00 00 01 02 03 0B 48 65 6C 6C 6F 20 57 6F 72 6C 64 00 AF 00 00 C9 6C
Description	F5 : Application Frame Direct Indication
	01 : Destination End Point
	00 00 : source address (coordinator)
	01 : Source End Point
	02 03 : Cluster ID (it is in little endian)
	0B : Payload length
	48 65 6C 6C 6F 20 57 6F 72 6C 64 : “Hello World”
	00 : it was not a MAC broadcast
	AF : It was not encrypted (without security)
	00 00 : last hop source (the coordinator)
	C9 : RSSI
	6C : Correlation
Direction	ZEx1 module -> Host

NOTE:
RSSI and correlation provide information about link quality.

4.6. How to define a profile

The Democase profile is a simple profile to show the ZigBee functionalities provided by ZEx1 modules.

If the user has to develop its own profile he can disable the Democase profile, register the new profile info into the ZEx1 module and implement the new profile in an external host using the serial interface.

In the next example the Democase profile will be disabled and the info for a device of a new profile will be added.

The new device will have the features described below:

Field	Length (Byte)	Value
Profile ID	2	0xFC53
Device ID	2	0x0001
Device Ver.	1	0x01
In Cluster Count	1	0x02
Out Cluster Count	1	0x03
In Cluster List	4	0x0001 0x0002
Out Cluster List	6	0x0003 0x0004 0x0005

The steps to disable the democase profile and register the info related to the device with the new profile are described below.

1. Disable Democase profile:

	Command
Hex	04 12 1A 01 01
Description	12 : Set Request
	1A : Disable Compiled Simple Descriptors attribute ID
	01 : Disable Compiled Simple Descriptors length
	01 : the Democase profile is disabled
Direction	Host -> ZEx1 module

NOTE:
After disabling the Democase profile its end point cannot be used anymore.

Expected packet sent by the ZEx1 module:

	Command
Hex	03 13 00 1A
Description	13 : Set Confirm
	00 : Success status
	1A : Disable Compiled Simple Descriptors attribute ID
Direction	ZEx1 module -> Host

2. Register info related to the new device on End Point 8:

	Command
Hex	15 12 1B 12 08 FC 53 00 01 01 02 03 00 01 00 02 00 03 00 04, 00 05
Description	12 : Set Request
	1B : Simple Descriptors attribute ID
	12 : Simple Descriptors length
	08 : End Point FC 53 : Profile ID 00 01 : Device Type 01 : Device Version 02 : Number of Input Clusters 03 : Number of Output Clusters 01 00 02 00 : Input clusters list 03 00 04 00 05 00 : Output clusters list
Direction	Host -> ZEx1 module

Expected packet sent by the ZEx1 module:

	Command
Hex	03 13 00 1B
Description	13 : Set Confirm
	00 : Success status
	1B : Simple Descriptors attribute ID
Direction	ZEx1 module -> Host

NOTE:
After the new device info has been registered the device can start or join the network and use the new end point to exchange data through the Application Frame Direct request.

5. Glossary

ARIB	Association of Radio Industries and Businesses
BER	Bit Error Rate
Bits/s	Bits per second (1000 bits/s = 1Kbps = 1Kbaud)
CFR	Code of Federal Regulations
Chips	Chip or chip sequence refers to a spreading-code used to transform the original data to DSSS
dBm	Power level in decibel milliwatt (10 log (P/1mW))
EMC	Electro Magnetic Compatibility
DSSS	Direct Sequence Spread Spectrum
EPROM	Electrical Programmable Read Only Memory
ERC	European Radiocommunications Committee
ESR	Equivalent Series Resistance
ETR	ETSI Technical Report
ETSI	European Telecommunication Standard Institute
FCC	Federal Communications Commission
IEEE	Institute of Electrical and Electronics Engineers
ISM	Industrial, Scientific and Medical
KB	1024 bytes (1 byte = 8 bits)
kbps	kilobits/s
LBT	Listen Before Talk
LNA	Low Noise Amplifier
MAC	Medium Access Control
MHz	Mega Hertz (1 MHz = 1000 kHz)
Mchip/s	Mega chips per second (A measure of the speed with which chips are generated in DSSS)
PCB	Printed Circuit Board
PROM	Programmable Read Only Memory
PER	Packet Error Rate
PHY	Physical Layer
RF	Radio Frequency
RoHS	Restriction of Hazardous Substances
RSSI	Receive Strength Signal Indicator
Rx	Reception
SRAM	Static Random Access Memory
SRD	Short Range Device
SMD	Surface Mounted Device
Tx	Transmission
Via	Metal Hole on a printed circuit board
WPANs	Wireless Personal Area Networks

5.1. Document change log

Revision	Date	Changes
ISSUE # 0	2010-12-14	First Release
ISSUE # 1	2011-02-22	Modified par 2.2, par 4.1.1.3, par 4.2.2.2, par 4.2.3.18, par 4.2.4.2
ISSUE # 2	2011-06-08	Modified par 4.2.2.2, par 4.2.4.2
ISSUE # 3	2011-08-04	Modified par 4.1.1.3
ISSUE # 4	2013-05-02	Modified chapter 4
ISSUE # 5	2013-09-24	Updated with the new USB EVK

6. Annexes

Confirm Status Table and Enumerations:

Status	Value
SUCCESS	0x00
ERROR	0x01
ERROR_ACQ_COORD	0x0B
ERROR_SCANNING	0x0A
ERROR_JOIN_ROUTER	0x0F
ERROR_ALREADY_ASSOCIATED	0x10
ERROR_ACQ_DEVICE	0x0C

ZDP Enumeration	Value
ZDP_INV_REQUESTTYPE	0x80
ZDP_DEVICE_NOT_FOUND	0x81
ZDP_INVALID_EP	0x82
ZDP_NOT_ACTIVE	0x83
ZDP_NOT_SUPPORTED	0x84
ZDP_TIMEOUT	0x85
ZDP_NO_MATCH	0x86
ZDP_NO_ENTRY	0x88
ZDP_NO_DESCRIPTOR	0x89
ZDP_INSUFFICIENT_SPACE	0x8A
ZDP_NOT_PERMITTED	0x8B
ZDP_TABLE_FULL	0x8C
ZDP_NO_BOUND_DEVICE	0x8D
ZDP_NO_SHORT_ADDRESS	0x8E
ZDP_NO_ACK	0x8F

APS Enumeration	Value
APS_ASDU_TOO_LONG	0xA0
APS_DEFRAG_DEFERRED	0xA1
APS_DEFRAG_UNSUPPORTED	0xA2
APS_ILLEGAL_REQUEST	0xA3
APS_INV_BINDING	0xA4
APS_INV_GROUP	0xA5
APS_INVALID_PARAM	0xA6
APS_NO_ACK	0xA7
APS_NO_BOUND_DEVICE	0xA8
APS_NO_SHORT_ADDRESS	0xA9
APS_NOT_SUPPORTED	0xAA
APS_SECURED_LINK_KEY	0xAB
APS_SECURED_NWK_KEY	0xAC
APS_SECURITY_FAIL	0xAD
APS_TABLE_FULL	0xAE
APS_UNSECURED	0xAF
APS_UNSUPPORTED_ATTRIBUTE	0xB0

MAC Enumeration	
CHANNEL_ACCESS_FAILURE	0xE1
FRAME_TOO_LONG	0xE5
INVALID_PARAMETER	0xE8
NO_ACK	0xE9
	0xEB
TRANSACTION_EXPIRED	0xF0
UNSUPPORTED_ATTRIBUTE	0xF4

NWK Enumeration	Value
NWK_INVALID_PARAMETER	0xC1
NWK_INVALID_REQUEST	0xC2
NWK_NOT_PERMITTED	0xC3
NWK_STARTUP_FAILURE	0xC4
NWK_ALREADY_PRESENT	0xC5
NWK_SYNC_FAILURE	0xC6
NWK_TABLE_FULL	0xC7
NWK_UNKNOWN_DEVICE	0xC8
NWK_UNSUPPORTED_ATTRIBUTE	0xC9
NWK_NO_NETWORKS	0xCA
NWK_LEAVE_UNCONFIRMED	0xCB
NWK_NO_KEY	0xCD
NWK_NO_ROUTING_CAPACITY	0xCF
NWK_ROUTE_DISCOVERY_FAILED	0xD0
NWK_ROUTE_ERROR	0xD1