dresden elektronik

User Manual deRFdevelopmentKit ZigBee Light Link 2.4 GHz

Version V1.0

2012-09-18

Table of contents

Document history

Date	Version	Description
20112-09-18	1.0	Initial version

1. Overview

This document is intended for engineers and software developers evaluating and operating lighting applications designed in compliance to the ZigBee Light Link specification and the IEEE 802.15.4 standard.

The deRFdevelopmentKit ZigBee Light Link 2.4 GHz serves as a development environment for quick and easy start-up, development and evaluation of ZigBee Light Link (ZLL) applications.

The kit contains a complete development environment for wireless ZigBee Light Link networks based on the ZigBee/IEEE 802.15.4 standard. This out-of-the-box solution provides everything to operate. Connect the lights, control it by the key remote control and develop or evaluate network functionality.

The lights and the radio controller board for the key remote control are delivered with a preflashed ZLL residential lighting application based on Atmel's BitCloud ZigBee Light Link Software Development Kit (SDK), a software package for developing ZigBee Light Link profile applications.

The included CD-ROM contains a software package with example applications that covers typical IEEE 802.15.4 ZigBee Light Link features. Documentation, datasheets and manuals of the particular hard- and software components are also included.

The kit contains an USB radio stick and the software deCONZ, dresden elektronik's generic monitoring and control solution for ZigBee PRO networks. This software supports profiles defined by the ZigBee standard as well as additional manufacturer specific profiles.

deConz uses the USB radio stick to join the ZigBee PRO network and visualises its different nodes and their relationships. Clusters implemented in node's firmware can be read out and controlled using the graphical frontend of deCONZ.

In addition to the kit's key remote control, the deCONZ software gives a second way to control the network from a PC. Lights can be turned on/off, changed in intensity or color and much more.

2. Glossary

Term	Description
API	Application Programming Interface
BitCloud SDK	Atmel's Software Development Kit
BSP	Board Support Package, for BitCloud SDK
GUI	Graphical User Interface
IDE	Integrated Development Environment
IEEE 802.15.4	IEEE 802.15.4-standard, applicable to low-rate WPAN
ISM	Industrial, Scientific and Medical frequency band
JTAG	Joint Test Action Group, digital interface for debugging of embedded devices, also known as IEEE 1149.1 standard interface
MAC	Medium Access Control ... layer, address etc.
RCB	Radio Controller Board
RF4CE	Radio Frequency for Consumer Electronics Consortium
SDK	Software Development Kit
SoC	System on Chip
WPAN	Wireless Personal Area Network
WSN	Wireless Sensor Network
ZigBee	Low-cost, low-power wireless mesh network standard. The ZigBee Alliance is a group of companies that maintain and publish the ZigBee standard.
ZLL	ZigBee Light Link, a ZigBee profile for lighting applications
6LoWPAN	IPv6 over Low Power Wireless Personal Area Networks

3. Features

The following figure gives an overview of the parts contained in the deRFdevelopmentKit ZigBee Light Link 2.4 GHz.

Figure 1: Kit contents deRFdevelopmentKit ZigBee Light Link 2.4 GHz

Kit contents

- 3x deRFlight
- 3x AC Power Adapter with clip set for international power standards
- 1x Key remote control
- 1x Radio Controller Board deRCB128RFA1 V6.3.1 for key remote control
- 1x 2.4 GHz stub antenna for Radio Controller Board
- 2x AAA batteries
- 1x deRFdevelopmentKit ZigBee Light Link 2.4 GHz Quick Start Guide
- 1x deCONZ Quick Start Guide
- 1x USB radio stick deRFusb-23E06 JTAG
- 1x CD-ROM with
    - deCONZ software
    - USB driver
    - user manuals, datasheets, software examples

Features

- Development environment for ZigBee Light Link applications
- Pre-flashed ZigBee Light Link example for a residential lighting application
- Designed for 2.4 GHz frequency band
- CD-ROM with software examples, documentation and drivers
- deCONZ monitoring and control software tool for ZigBee PRO networks to allow a detailed look inside the network states

4. Hardware description

This section describes the hardware components supplied with the development kit.

4.1. deRFlight

The deRFlight as shown in Figure 2 consists of

- deRFnode-2TNP2 board
- pluggable radio module deRFmega128-22T00
- quad RGB power LED board
- diffusor plane, back plane

The diffusor and backplane give the deRFlight a more realistic lamp design. Components are plugged together and secured with 4 screws on both sides.

Figure 2: deRFlight

4.1.1. deRFnode-2TNP2 board

The deRFnode is the base board for the deRFlight. The main purpose is to hold the radio module and the RGB power LED board. Electrically it serves as interconnection between radio module and power LED PCB. Beyond that it distributes regulated voltage to the components. Figure 3 shows the deRFnode board. More details can be found in the deRFnode Datasheet [1] and the deRFnode User Manual [2].

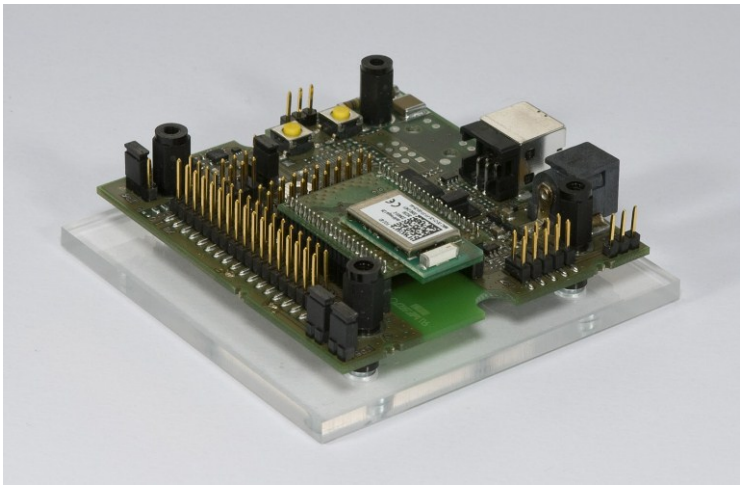

Figure 3: deRFnode-2TNP2 – board with radio module deRFmega128-22M00

Please Note: *The deRFlight is equipped with an USB connector. For the demo lighting application USB should not be used. Due to the power consumption of the high power RGB LEDs it is recommended to use the provided AC power adapter instead.*

4.1.2. Radio Module deRFmega128-22T00

The deRFlight is equipped with the radio module deRFmega128-22M00. The LGA board is soldered on a small adapter board to fit into the deRFnode socket. This pluggable module is called deRFmega128-22T00 and shown in Figure 4.

Technical details about the mini module itself can be found in the datasheet [3] and in the user manual [4]. The pinout of deRFmega128-22T00 is described in the adapter board datasheet [5].

Please note: *Care should be taken when the module is levered out of the deRFnode socket. Do not bend the connector pins nor scratch the small wires on the PCB.*

Figure 4: deRFmega128-22M00 mounted on an adapter board

4.1.3. Power LED board

The power LED overlay board (see Figure 5) is plugged on top of the deRFnode.

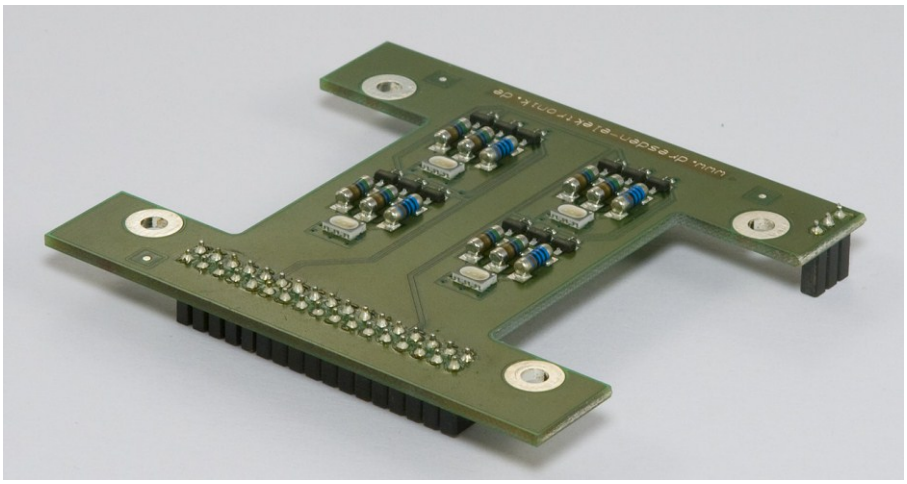

Figure 5: Quad RGB power LED board

Figure 6 shows the schematic circuit diagram of the LED board.
Please note: *The red LEDs on the deRFnode board shows the inverted state of the power LEDs different color channels.*

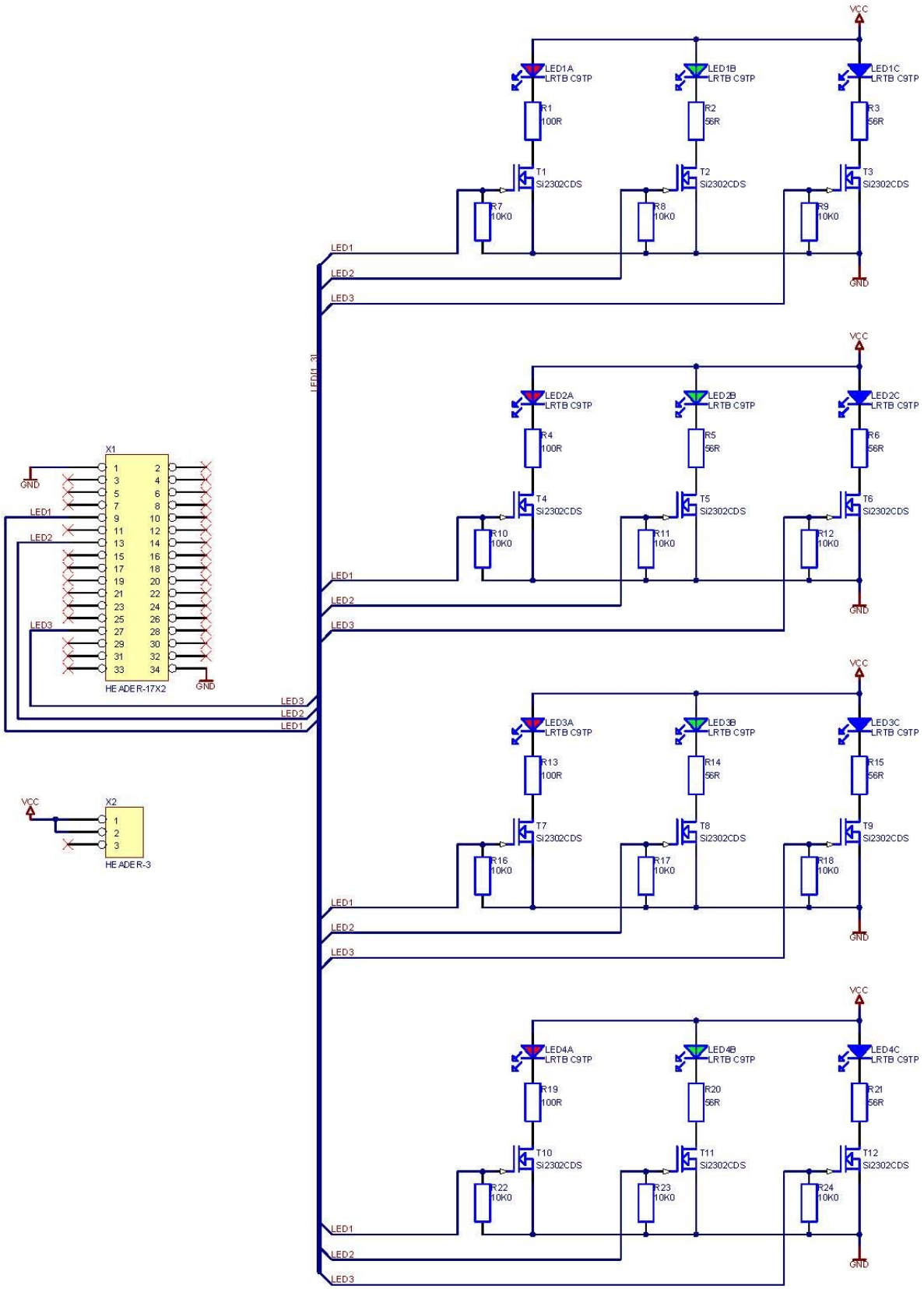

Figure 6: Schematic of power LED board

4.2. Key remote control

The key remote control consists of two parts. First one is the red colored board with buttons, display and some pin headers. On top of it works the radio module RCB128RFA1 V6.3.1. Following subsections describe the hardware in more detail.

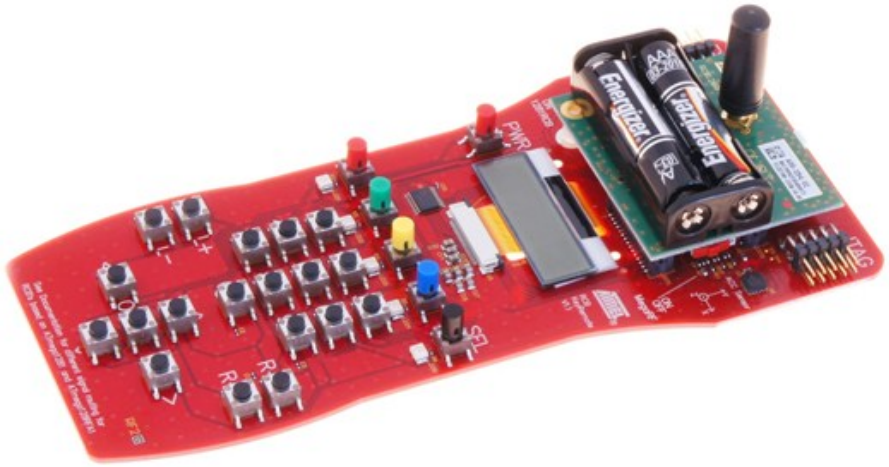

Figure 7: Key remote control with radio controller board RCB128RFA1 V6.3.1

4.2.1. Radio controller board RCB128RFA1 V6.3.1

The radio controller board RCB128RFA1 was originally designed to provide a reference design for the Atmel ATmega128RFA1 single-chip microcontroller and radio transceiver. The SoC integrates a powerful, 8-bit AVR RISC microcontroller, an IEEE 802.15.4-compliant transceiver, and additional peripheral features. The built-in radio transceiver supports the worldwide accessible 2.4 GHz ISM band.

In this development kit the RCB radio controller board is used as processing unit in conjunction with the key remote control. It is equipped with two AAA batteries and a 2.4 GHz stub antenna.

The radio controller board itself is described in detail in the Atmel datasheet [6].

4.2.2. RCB key remote control

The RCB key remote control is used with an Atmel RCB128RFA1 module in order to evaluate remote control applications. It is intended for further engineering, development, evaluation and demonstration purposes.

The Atmel application note [7] provides detailed hardware descriptions for the individual function blocks of the RCB key remote control board.

4.2.3. Connecting RCB and key remote control board

Unpack the key remote control board and the RCB. Insert 2x AAA batteries in right polarity. Mount the stub antenna onto the SMA connector. Then insert the radio module into the RCB connectors on the key remote control.

The key remote control and RCB hardware are both powered by the batteries on the RCB. The power supply switch on the RCB is used to connect or disconnect the battery power supply voltage. In order to use the battery supply, a jumper bridge must be connected at X1 between pin 2 and pin 4 as shown in Figure 8.

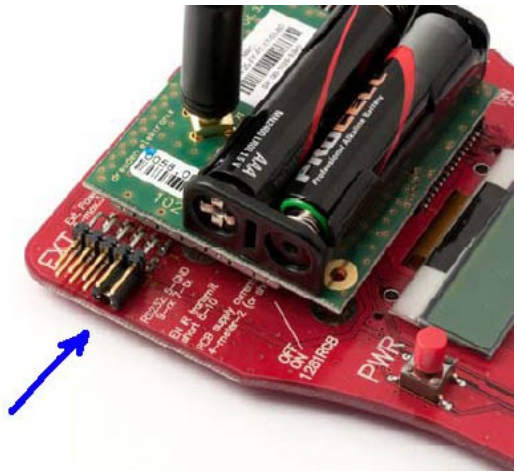

Figure 8: X1 connector with jumper bridge for battery supply

The power switch is located on the right hand side of the module. On and Off are marked on the bottom side of the board, see Figure 9.

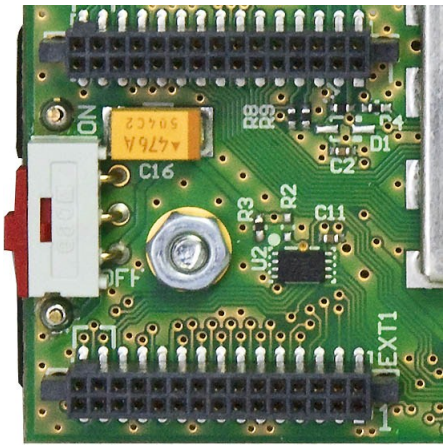

Figure 9: RCB128RFA1 V6.3.1 power switch

4.3. USB radio stick deRFusb-23E06 JTAG

The USB radio sticks deRFusb-23E06 JTAG contains a powerful Cortex-M3 microcontroller with 256 kBytes High-Speed Flash and a 2.4 GHz ISM band transceiver. The transceiver AT86RF231 is intended for ZigBee, IEEE 802.15.4, 6LoWPAN, RF4CE and proprietary ISM applications. Detailed information about hardware, software and device programming can be found in the datasheet [8] and user manual [9].

Figure 10: deRFusb-23E06 JTAG

5. Running the ZigBee Light Link demo application

The reference application involves color light devices and color scene controller devices, which are both defined in the ZigBee Light Link Profile specification [10]. Both types of devices are emulated by the deRFlights and the key remote control respectively.

The application demonstrates controlling color lights using a color scene controller. All main features of the ZigBee Light Link profile are implemented:

- A ZigBee Light Link network has no coordinator
- Touchlink commissioning – network parameters are transferred to a new light from a scene controller, while the scene controller is kept close to the light
- Lights are ZigBee routers to propagate messages across the network
- Multiple scene controllers may be used to control same or different sets of lights
- Application data and network parameters are saved to non-volatile memory to restore the state after reset, power off, etc.
- Light control functionality, including On/Off, Level (brightness), Color (Hue, Saturation), Groups, and Scenes.

5.1. Launching the demo

To launch the demo a color scene controller (key remote control) and one color light (deRFlight) are needed. More devices (lights or controllers) may be added as required.

- Assemble the key remote control as described in section 4.2.3. The device is pre-programmed with the demo application.
- Power on the device.
- Power on the deRFlight using the AC adapter. Please do not power the device using the USB connector. The power LEDs will draw more current than the USB interface can deliver.
- Perform a touchlink procedure between the color light and the color scene controller by holding the PWR button on the key remote control to form a network.
- Use color scene controller's buttons to control the light as described in section 5.3

5.2. Color light's function

The deRFlights are pre-flashed with the ZigBee Light Link demo application. Plug in the AC power adapter and connect it to the light. The controllers starts up; during that the LEDs may flicker somehow.

The deRFlight can now be controlled by the color scene controller, if the touchlink procedure was performed successfully.

Please note: A*ll settings like assigned network membership as well as light's on/off state, color, intensity and brightness are stored in non-volatile memory and recalled after light power up reset.*

5.2.1. Touchlink pairing with a controller device

For touchlink a color scene controller is brought closely (<10 cm) to a color light and the PWR button is pressed on the color scene controller for more than three seconds. The color scene remote sends an identify command to the color light, which blinks several times with

its LEDs. The touchlink procedure may be aborted at this moment, if the PWR button is released. To complete the procedure the key must be hold for additional three seconds.

Successfully paired lights emit green light as a response on touchlink procedure.
After that the color light can be controlled by commands sent from the color scene controller.

5.2.2. Reset deRFlight to factory new state

There are two ways to reset a color light device to the factory new state: sending a command from a color scene controller or using a button on the deRFnode board.

To trigger reset of a color light to factory new:

- Place a LED light in close proximity to key remote control.
- Hold **R+**, **R-** and **PWR** buttons altogether on the color scene controller for more than three seconds.
- LED light responds with turning on all lights followed by power down.

The second way to reset the light is to press the button 1 for more than 10 seconds (Button is shown in Figure 11).

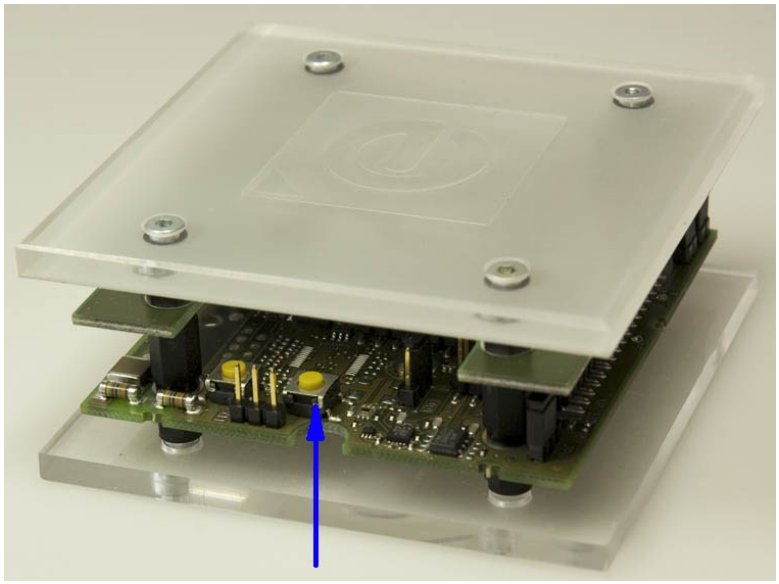

Figure 11: Button 1 on deRFlight

5.3. Color scene controller's function

Figure 12 shows the layout of the key remote control keyboard. All buttons are marked. The following subsections explain the different functionality of the scene controller in detail.

5.3.1. Main commands

Table 1 lists commonly used button functions for the color scene controller's commands. See the scheme of the key remote control board with all available commands on Figure 13 for complete reference.

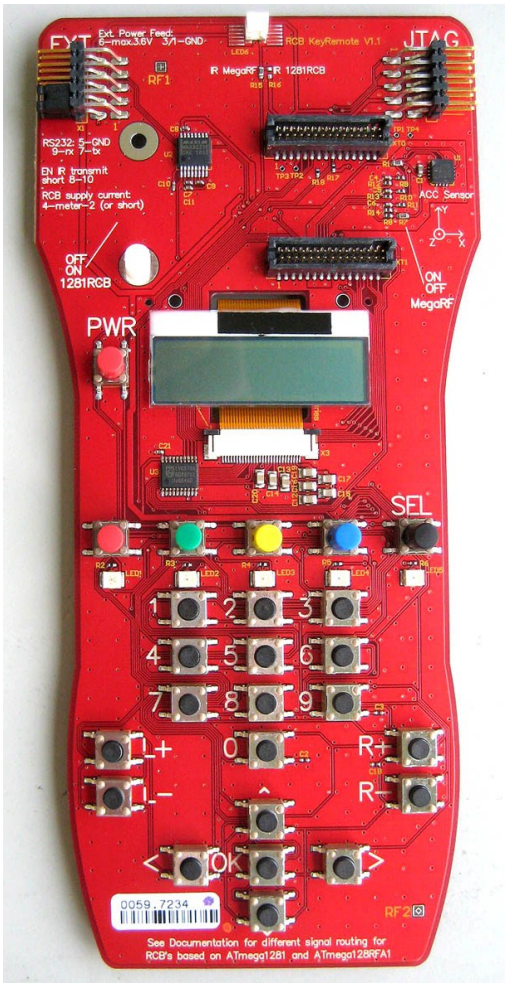

Figure 12: Key remote control base board without RCB

Table 1: Buttons for executing most frequent commands

Button(s)	Description
PWR	Press and hold for 3 seconds to perform touchlink
PWR & **R+**	Press and hold for 3 seconds to perform classical ZigBee scanning
PWR & **R-**	Press and hold for 3 seconds to reset device to factory new state
PWR & **R+** & **R-**	Reset a color light to the factory new state. The target color light should be selected using the SEL button; otherwise all color lights present in the network will be reset to the factory new state.
L+ / **L-**	Color light's on/off
Up / **Down**	Increase/decrease light level
Left / **Right**	Increase/decrease saturation
Colored buttons	Sets the corresponding color, turns light on, sets saturation and brightness to 100 %
SEL	Select the next bound device and requests it to identify itself. This allows sending unicast commands to a single device. Groupcast mode will be entered after selection of the last device.
1 / **2** / **3**	Store light scene if pressed for more than 3 seconds and recall light scene if pressed for less than 3 seconds
5	Sends an identify command to the currently selected light or group. Selected light(s) responds with blinking.
6	Start/stop commissioning mode
7 / **8** / **9**	Set minimum/middle/maximum light intensity

The **SEL** button may be used to select a color light. When this button is pressed and released, the next color light becomes active, which means that all commands initiated by the user are sent to this color light in a unicast manner. After selection of the last bound device the group is selected. This means that commands are sent to all bounded devices in a groupcast manner.

5.3.2. Light control command reference

Figure 13 shows the key remote control keyboard scheme and all commands that can be sent with the buttons from the color scene controller. Each button may be used to send up to 4 commands. What command is send by pressing a button depends on whether buttons **R+** and **R-** are also pressed or not. Table 2 shows the different combinations.

Table 2: Alternating a command by holding **R+** and **R-** buttons

Command	**R+ and R- button state**
first command	both buttons are not pressed
second command	**R+** pressed, **R-** not pressed
third command	**R+** not pressed, **R-** pressed
fourth command	both buttons pressed

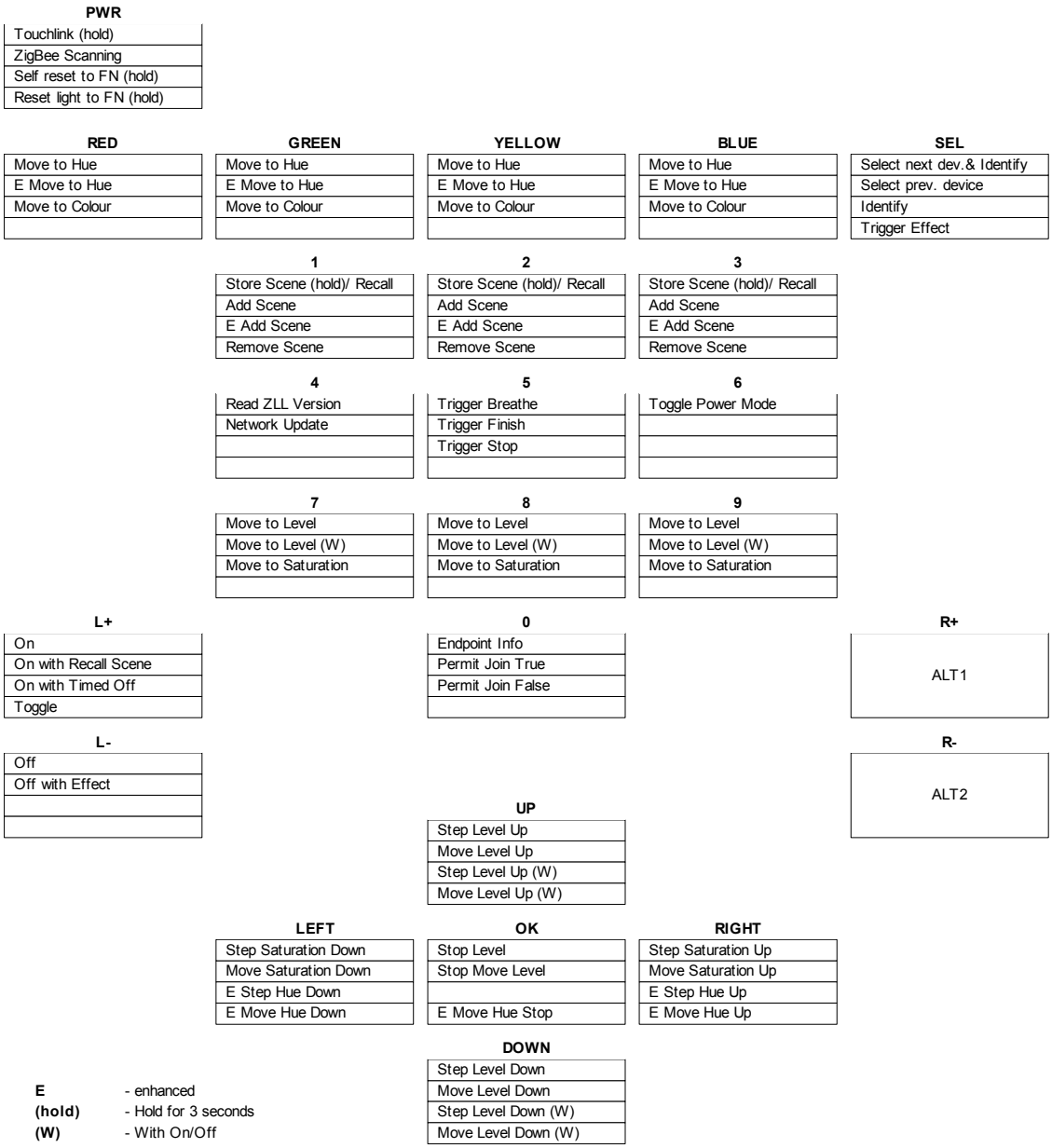

Figure 13: Color scene controller keyboard scheme

5.3.3. Status codes

The status of a key press event is shown in the lower right corner of the LCD display. Following codes are used:

Table 3: Key remote control status codes

Code	Description
0x00	Ok, command successfully executed
0xE9	No MAC ACK
0xA7	No APS ACK

5.3.4. Touchlink functionality to form a network

For touchlink a color scene controller is brought close to a new color light, which is not in the network yet. By pressing **PWR** button on a color scene controller, the user initiates a commissioning procedure, which goal is to transfer network parameters to the color light. At this moment the devices are not yet in the network, but the communication is possible, because it happens on the MAC level without a need for routing.

The color light receives network parameters and starts the network as a router (the adjustments in the BitCloud stack allow this). Once the network is started, the color scene controller joins this network as an end device.

When the first color light is commissioned, the color scene controller may be brought to the second color light, which, receiving the network parameters, will not start the network once again, but will join the existing network as another router. The subsequent color lights are commissioned in the same way.

5.3.5. Reset key remote control to factory new state

To reset the key remote control to factory new state press and hold **R-** and **PWR** buttons altogether for more than three seconds.

5.4. Multiple color scene controllers in a network

A Light Link network may contain more than one color scene controller. Additional controller devices are added using touchlink, which happens between two color scene controllers:

- Bring a factory new color scene controller to a color scene controller that is in the network and hold the **PWR** button on both devices for more than 3 seconds.
- Perform touchlink between the new color scene controller and each of the color lights.
- Use any of the controller devices to manipulate color lights.

Please note: *If touchlink between controller devices is not performed pairing a new controller device with a color light that is already in the network it will cause the color light's leaving this network and form a new one.*

5.5. Supported Clusters

Table 4 lists clusters supported by the demo application for color light and color scene controller. Note that most of the clusters used by Light Link applications duplicate common clusters from ZigBee Cluster Library, but may be slightly different, and so applications should employ clusters specially defined for the ZigBee Light Link profile (header files for these clusters include Zll in their names).

Table 4: Clusters supported by the demo application (s – server, c – client)

Colour Light	**Colour Scene Controller**
Basic (s)	Basic (s)
Commissioning (s)	Commissioning (s)
OnOff (s)	Commissioning (c)
Level control (s)	OnOff (c)
Color control (s)	Level control (c)
Groups (s)	Groups (c)
Identify (s)	Identify (c)
Scenes (s)	Scenes (c)
Commissioning (c)	

6. BitCloud SDK

6.1. SDK's contents

The SDK includes the BitCloud implementation of ZigBee PRO protocol and ZigBee Cluster Library, the Light Link demo application and a set of pre-built firmware images and development tools.
The BitCloud stack included in this package contains the ZigBee Cluster Library implementation and is very similar to the BitCloud Profile Suite stack. However, a few enhancements are made to adjust to the behaviour needed for ZigBee Light Link profile. Table 5 describes the file structure of the software package supplied with the SDK.

Table 5: BitCloud ZigBee Light Link SDK's structure.

Name	Location
`/BitCloud/Components`	Header files and source code for some components (HAL,BSP, and drivers)
`/BitCloud/Components/ZCL/ZLL`	Header files with definitions of clusters used by Light Link applications, and ZLL's specific API definitions
`/BitCloud/lib`	BitCloud compiled libraries
`/Applications/ZLLDemo`	Demo application's sources and projects
`/Evaluation Tools/ZLLDemo`	Demo application's pre-compiled images
`/Documentation`	Empty, see `/Documentation` folder on the Kit CD-ROM

In addition to this document, the documentation includes the API Reference, which contains specification of functions and times available for the user, and the developer guides.

6.2. Supported platforms and tools

Supported platforms and development tools are described in Table 6.

Table 6: System requirements for BitCloud ZigBee Light Link SDK.

Parameter	Value	Note
MCU/RF	Atmel ATmega128RFA1 SoC	
deRFlight	deRFnode with deRFmega128-22T00 radio module and power LED overlay board	Additional deRFlights can be purchased by dresden elektronik ingenieurtechnik gmbh
Key remote control	RCB128RFA1 hosted on a Key Remote Control board for a color scene controller	Additional key remote controls can be purchased by dresden elektronik ingenieurtechnik gmbh
JTAG emulator	Atmel AVR JTAGICE mkII or Atmel JTAGICE 3	Required to upload firmware onto the boards and debug through JTAG
IDE	IAR Embedded Workbench® for AVR 6.11 (with IAR™ C/C++ Compiler for AVR v6.11.1.50453)	Required to develop applications using API and to upload firmware images through JTAG

7. Programming the devices

7.1. Tool chain installation

- Install IAR Embedded Workbench for AVR [7], if not already installed on your PC
- Add a Windows environment variable named IAR_AVR_HOME, and set its value to the IAR Embedded Workbench installation directory (for a default installation, it is `C:\Program Files\IAR Systems\Embedded Workbench 6.11`). To do this:
  - Go to `Control Panel > System > Advanced > Environment Variables`
  - Click `New` below the `System variables` list, and enter `Variable Name` and `Variable Value`

  This step is required if you plan to build embedded images using IAR Embedded Workbench from the command line.

More details on tool chains used with BitCloud may be found in [11] or [12].

7.2. Fuse bit settings

The recommended fuse bits settings are shown in Table 7.

Table 7: Fuse bits for Atmel ATmega128RFA1.

Option	Required value (for 8MHz)	Required value (for 16MHz)
BODLEVEL	1V8	1V8
OCDEN	Enabled	Enabled
JTAGEN	Enabled	Enabled
SPIEN	Enabled	Enabled
WDTON	Disabled	Disabled
EESAVE	Enabled	Enabled
BOOTSZ	Boot Flash size=2048 words start address=$F800	Boot Flash size=2048 words start address=$F800
BOOTRST	Disabled	Disabled
CKDIV8	Enabled	Enabled
CKOUT	Disabled	Disabled
SUT_CKSEL	Int. RC osc.; Startup time: 6 CK + 65ms	Transceiver Oscillator; Startup time 16K CK + 65ms
Resulting bytes		
Ext	0xFE	0xFE
High	0x13	0x13
Low	0x62	0x77

- Note 1: the demo application is already pre-flashed into the devices. The pre-built application images are located in the `Evaluation Tools` directory.
- Note 2: The EESAVE fuse should be enabled to prevent deletion of EEPROM stored MAC address during chip erase.

7.3. Programming with IAR Embedded Workbench

7.3.1. Precompiled images

When using IAR Embedded Workbench to program precompiled images provided with the SDK, the user first needs to create a project containing the precompiled image.

1. Start IAR Embedded Workbench for AVR
2. Select `File > New > Workspace`
3. Select `Project > Create New Project…`
4. In the `Create New Project` dialog, select `Externally build executable` in `Project templates:`
5. Select a name for the project, and click `Save`
6. Follow the instructions in `readme.txt`
7. Once the project is set up, select `Project > Options`
8. In the `General options` category, set `Processor Configuration` to `-cpu=xm256a3, ATmega128RFA1`
9. Click `OK`
10. Select `JTAGICE mkII > Fuse Handler` (`JTAGICE 3 > Fuse Handler`)
11. Click `Read Fuses`, and make sure that the device fuses are set as specified in section 7.2
12. If fuses are set incorrectly, select the correct fuse settings, and click `Program fuses`
13. Select `Project > Download and debug`
14. Once the debugging session has started, click `Stop debugging`

The image is now installed on the board.

7.3.2. Application workspace

1. Double-click on a workspace (for example, `ZLLDemo.eww`), and select `JTAGICE mkII > Fuse Handler` (`JTAGICE 3 > Fuse Handler`)
2. Click `Read Fuses`, and make sure that the device fuses are set as specified in section 7.2
3. If fuses are set incorrectly, select the correct fuse settings, and click `Program fuses`
4. Select the desired application configuration from the drop-down box in the `Workspace` pane
5. Select `Project > Download and debug`
6. Once the debugging session has started, click `Stop debugging`

The image is now installed on the board.

8. deCONZ

8.1. Purpose

ZigBee is a technology which offers a powerful solution to a wide range of low-power, low-cost wireless sensor network applications. Some popular application profiles are Home Automation, Smart Energy or Light Link; beside them and other public profiles ZigBee PRO provides the possibility to easily develop special purpose applications.

In many stages of a product development process it is necessary to interact with the devices in order to verify their correct operation. To achieve this in an efficient way extra PC tools are often built around the related application first for the developer and later for deployment, for operation and for maintenance. The deCONZ application from dresden elektronik is a powerful graphical tool addressing all those stages.

The deCONZ software provides comprehensive monitoring, control and commissioning capabilities based on the ZigBee PRO specification. The application core is kept completely generic and is therefore not limited to a specific application profile. All ZigBee application specifics like devices, profiles and clusters are described in XML files. Based on this information, the deCONZ application can generate a full functional graphical user interface for each device and any application.

The main applications for the deCONZ software are:
- Operate ZigBee PRO networks
- Device application monitoring & control
- Create/remove bindings between devices
- Commissioning

8.2. USB radio stick installation

The development kit contains an USB radio stick deRFusb-23E06. It is intended to be the gateway between the PC application deCONZ and the wireless IEEE 802.15.4 network.

Please insert the USB stick into the PC. The system asks for a driver to install. The USB driver is provided on the CD-ROM in the folder `<drive letter>:\Driver`.

After successful installation a new COM device should appear in the Windows Device Manager as shown in Figure 14.

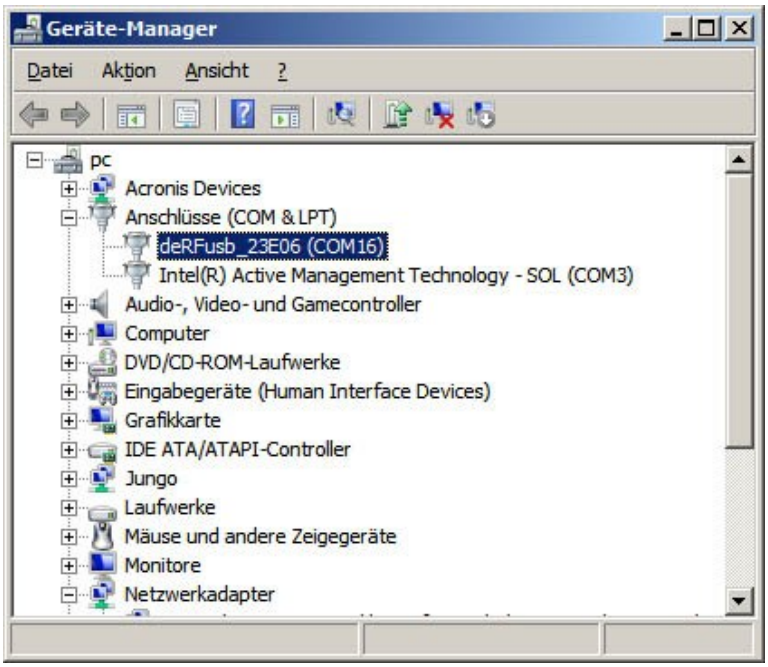

Figure 14: USB radio stick as new COM device in Windows Device Manager

8.3. deCONZ setup and operation

The installation file of the deCONZ software is on the CD-ROM. The setup program can be found at `<drive letter>:\Packages\deCONZ\deCONZ_Setup_Win32_V1_06.exe`.

Install it and then follow the instructions of the deCONZ Quick Start Guide delivered with the kit hardware. It describes the basic usage of the deCONZ software.

The deCONZ software itself will be extended and improved continuously by dresden elektronik ingenieurtechnik gmbh. Therefor please find the latest revision of the user manual online at http://www.dresden-elektronik.de.

9. References

[1] Datasheet deRFnode-2TNP2,
URL: http://www.dresden-elektronik.de/funktechnik/uploads/media/deRFnode-2TNP2-DBT-en.pdf

[2] User Manual deRFnode-2TNP2
URL: http://www.dresden-elektronik.de/funktechnik/uploads/media/deRFnode_deRFgateway-BHB-en.pdf

[3] Datasheet Radio modules deRFmega128-22M00 | 22M10,
URL: http://www.dresden-elektronik.de/funktechnik/uploads/media/deRFmega128-22M00_22M10-DBT-en.pdf

[4] User Manual Radio Modules deRFmega128-22M00 | 22M10 | 22M12
URL: http://www.dresden-elektronik.de/funktechnik/uploads/media/deRFmega128-22M00_22M10_22M12-BHB-en.pdf

[5] Datasheet Adapter Board deRFholder 22T00 | 22T02,
URL: http://www.dresden-elektronik.de/funktechnik/uploads/media/deRFholder-22T00_22T02-DBT-en.pdf

[6] Atmel AVR2044: RCB128RFA1 – Hardware User Manual
URL:http://www.atmel.com/Images/doc8339.pdf

[7] Atmel AVR2037: RCB Key Remote Control - Hardware User Manual
URL:http://www.atmel.com/Images/doc8356.pdf

[8] Datasheet USB Radio Stick deRFusb 23E00 | 23E06 JTAG
URL: http://www.dresden-elektronik.de/funktechnik/uploads/media/deRFusb-23E00_23E06_JTAG-DBT-en.pdf

[9] User Manual USB Radio Sticks
URL: http://www.dresden-elektronik.de/funktechnik/uploads/media/deRFusb-BHB-en.pdf

[10] 11-0037-10 ZigBee Light Link Profile: Specification Version 1.0

[11] Atmel AVR2052: BitCloud Quick Start Guide
URL: http://www.atmel.com/Images/doc8200.pdf

[12] Atmel AVR2055: BitCloud Profile Suite Quick Start Guide
URL: http://www.atmel.com/Images/doc8348.pdf

[13] ZigBee Alliance
URL: http://www.zigbee.org/Specifications/ZigBee/NetworkTopology.aspx

dresden elektronik ingenieurtechnik gmbh
Enno-Heidebroek-Straße 12
01237 Dresden
GERMANY

Phone +49 351 - 31850 0
Fax +49 351 - 31850 10
www.dresden-elektronik.de
E-Mail: wireless@dresden-elektronik.de

Trademarks and acknowledgements

- IEEE 802.15.4™ is a trademark of the Institute of Electrical and Electronics Engineers (IEEE).
- ZigBee® is a registered trademark of the ZigBee Alliance.
- Atmel® Atmel logo and BitCloud® are registered trademarks or trademarks of Atmel Corporation or its subsidiaries.
- Windows® and others are registered trademarks or trademarks of Microsoft Corporation in U.S. and or other countries.

All trademarks are registered by their respective owners in certain countries only.
Other brands and their products are trademarks or registered trademarks of their respective holders and should be noted as such.

Disclaimer

This note is provided as-is and is subject to change without notice. Except to the extent prohibited by law, dresden elektronik ingenieurtechnik gmbh makes no express or implied warranty of any kind with regard to this guide, and specifically disclaims the implied warranties and conditions of merchantability and fitness for a particular purpose. dresden elektronik ingenieurtechnik gmbh shall not be liable for any errors or incidental or consequential damage in connection with the furnishing, performance or use of this guide.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or any means electronic or mechanical, including photocopying and recording, for any purpose other than the purchaser's personal use, without the written permission of dresden elektronik ingenieurtechnik gmbh.