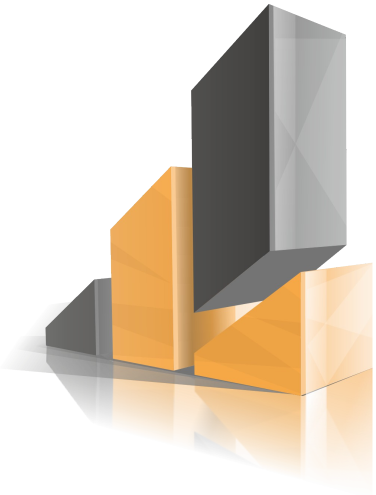
DSR

J,Skilled, *Trusted* Software Development

ZBOSS Sniffer User Manual



DSR Corporation
14818 W. Warren Ave., Suite 101
Lakewood, CO 80228, USA
Office: +1.720.962.9525 | Fax: +1.720.962.6625
www.dsr-company.com |www.dsr-wireless.com

Revision History

Date	Revision Type	Revision #	Comments	Initials
10/14/2013	*Major*	*1*	*Composed the document*	*AV*
02/07/2014	*Major*	*1.1*	*Merged with SubGHz sniffer description*	*AV*

CONTENTS

Revision History 2
Contents 3
INTRODUCTION 4
Document's Purpose 4
Intended Audience 4
Definitions, Acronyms, and Abbreviations 4
INSTALLATION 5
HARDWARE 6
ZBOSS Sniffer for 2,4 GHz 6
ZBOSS Sniffer for SubGHz 6
USAGE 7

Introduction

Document's Purpose

The document describes installation and use of ZBOSS Sniffer – open source cross-platform ZigBee packet sniffer, based on ZBOSS stack.

Intended Audience

Developers
QA Engineers
Users

Definitions, Acronyms, and Abbreviations

ZBOSS ZigBee Open Source stack http://zboss.dsr-wireless.com

Wireshark open source packet analyzer http://www.wireshark.org

TI (Texas Instruments) American company that designs and makes electronics http://www.ti.com

INSTALLATION

Installation of ZBOSS Sniffer is equal on both Windows and Linux platforms. After having got archive with the sniffer, unpack it to the preferred directory. Depending on the archive type (with binaries or sources), content may differ:

- in archives with binaries “gui” and “hex” and “doc” directories could be found; “gui” contains ZBOSS Sniffer GUI application, “hex” - flash images for sniffer devices and “docs” - this manual; in the archive for Windows operation systems a driver for sniffer devices is also placed;
- in archive with sources GUI and hardware sniffer sources are placed; GUI sources are provided with Qt Creator project and hardware sniffer sources are with IAR project; Those sources should be placed in “stack/devtools/sniffer” directory and IAR project should be included in the necessary ZBOSS IAR workspace (SubGHz or plain ZBOSS).

Sniffer is developed to work with Wireshark. To install Wireshark DSR build with many additional ZigBee dissectors can be used. Pure Wireshark without any plugins and downloaded from official site is suitable too. To work with Wireshark **WinPcap/LibPcap library must be installed**. Otherwise Wireshark will crash at the sniffer start.

HARDWARE

ZBOSS SNIFFER FOR 2,4 GHZ

Sniffer for 2,4 GHz is designed to work with TI SmartRf05EB via RS-232 and with CC2531 USB Dongle. There are two recommended ways to program hardware with ZBOSS sniffer:

- download prepared hex file for the target platform and load it to the device using SmartRf Flash Programmer;
- compile from ZBOSS sources with ZB_SNIFFER definition.

When using the first approach, perform the following steps:

1. start SmartRf Flash Programmer;
2. select “Program CCxxxx SoC” in the drop-down box;
3. check, that the correct SoC is detected (CC2530 for the evaluation board and CC2531 for the USB Dongle);
4. check “Erase, program and ”verify” radio button;
5. choose the hex image;
6. click “Perform actions”.

Notice, that CC2531 USB Dongle can be programmed only when connected to evaluation board. When programming USB Dongle evaluation module should be removed from the board and USB Dongle should be connected with the board via debug connector. If everything is correct SmartRf Flash programmer will detect SmartRf05Eb with CC2531 system on chip.

When compiling from sources, prepared IAR project “zboss_sniffer” can be used. It contains SmartRf05Eb and CC2531-Dongle configurations for each device. Notice, that for building sniffer ZB_SNIFFER and ZB_SNIFFER<packet transport>_TRACE should be defined. Possible packet transports are USB and SERIAL (ZB_SNIFFER_USB_TRACE for CC2531-Dongle and ZB_SNIFFER_SERIAL_TRACE for SmartRf05Eb). Programming the USB-Dongle with IAR doesn't differ from programming it with SmartRf Flash Programmer, that means that dongle should be connected to the board.

ZBOSS SNIFFER FOR SUBGHZ

Sniffer for SubGHz is designed to work with ZBOSS SubGHz kit: either with STM32F4 Discovery ot Olimex STM32-E407 boards. There are two recommended ways to program hardware:

- download prepared hex file for the target platform and load it to the device using STM32 ST-LINK Utility;
- compile from ZBOSS sources with ZB_SNIFFER definition.

When using the first approach, perform the following steps for Olimex STM32-E407 board:

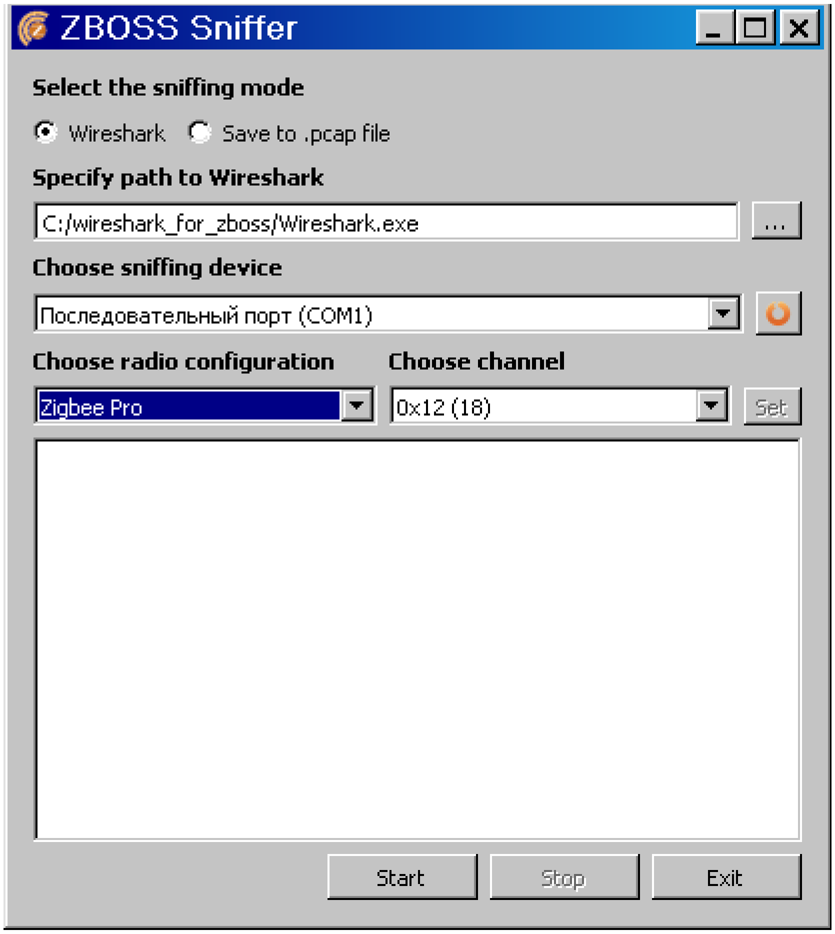
1. connect ST-LINK Debugger to the board via JTAG;
2. run STM32 ST-LINK Utility and click “Connect to the target”;
3. in the “Target” menu select “Program”;
4. choose hex image with firmware;
5. click “Start”;
6. after programming has finished, click “Disconnect from the target”;
7. reset device and check how it works.

Notice, that a process of programming STM32F4DISCOVERY with STM32 ST-LINK Utility doesn't differ from programming Olimex, except the fact that the DISCOVERY board already has built-in ST-LINK Debugger and there is no need to use external one. That's why Olimex programming instruction written above can be used starting from the second point.

When compiling from sources, prepared IAR project “zboss_sniffer” can be used. It contains Olimex and Discovery configurations. Notice, that ZB_SNIFFER should be defined and transport type selected. To build sniffer with UART define ZB_SNIFFER_UART and use M4_USB_SERIAL for USB transport.

USAGE

To start ZBOSS sniffer launch executable zboss_sniffer.exe on Windows platform or zboss_sniffer on Linux. The following window will appear:



ZBOSS Sniffer

Select the sniffing mode

● Wireshark ○ Save to .pcap file

Specify path to Wireshark

C:/wireshark_for_zboss/Wireshark.exe

Choose sniffing device

Последовательный порт (COM1)

Choose radio configuration: Zigbee Pro

Choose channel: 0x12 (18)

Sniffer needs some information to start capture:

1. Sniffing mode. There are two general options for sniffer available: sniffing to Wireshark and sniffing to .pcap file, which can be read by Wireshark later.
2. Path to the packet destination. Depending on sniffing mode, it should be path to Wireshark executable (when sniffing to Wireshark) or .pcap file (when sniffing to file). “Browse” button can be used to specify path in a typical for OS dialog form.
3. COM port which the sniffing device is connected to. If the required COM-port doesn't appear in the drop-down list, use “Update” button to refresh the list of active COM ports.
4. Radio configuration. Now ZigBee Pro for 2,4 GHz transceivers and SubGHz_EU2 for SubGHz transceivers on European frequencies are available.
5. Channel on which the capture will be performed.

After filling in all the required information, everything is ready to start packet capture. Click start (on start preferred paths to Wireshark and .pcap file and preferred channel are saved).

It takes up to 2 seconds to initialize the sniffer device, then depending on the sniffer mode, program behaves different ways. If sniffing to Wireshark, sniffer will look for it at the specified path and launch. If Wireshark hasn't been found, capture will be terminated. After launching Wireshark packets will be appearing in its window when device catches it. Notice, that for each new capture new Wireshark will be opened. If sniffing to file, after initializing device, capture is started and the only way to follow the process is to examine the sniffer's terminal output.

When capture is started, use buttons pause/resume and stop to manage it. The channel on which the capture is performed can be changed on-fly by clicking the “set” button near the channel configuration. When button pause is clicked, sniffer stops writing caught packets to destination. When button stop is clicked, sniffer closes the destination (Wireshark connection or pcap file) and becomes ready for a new capture. Notice, that Wireshark start, stop and restart controls can also be used to manage sniffer.