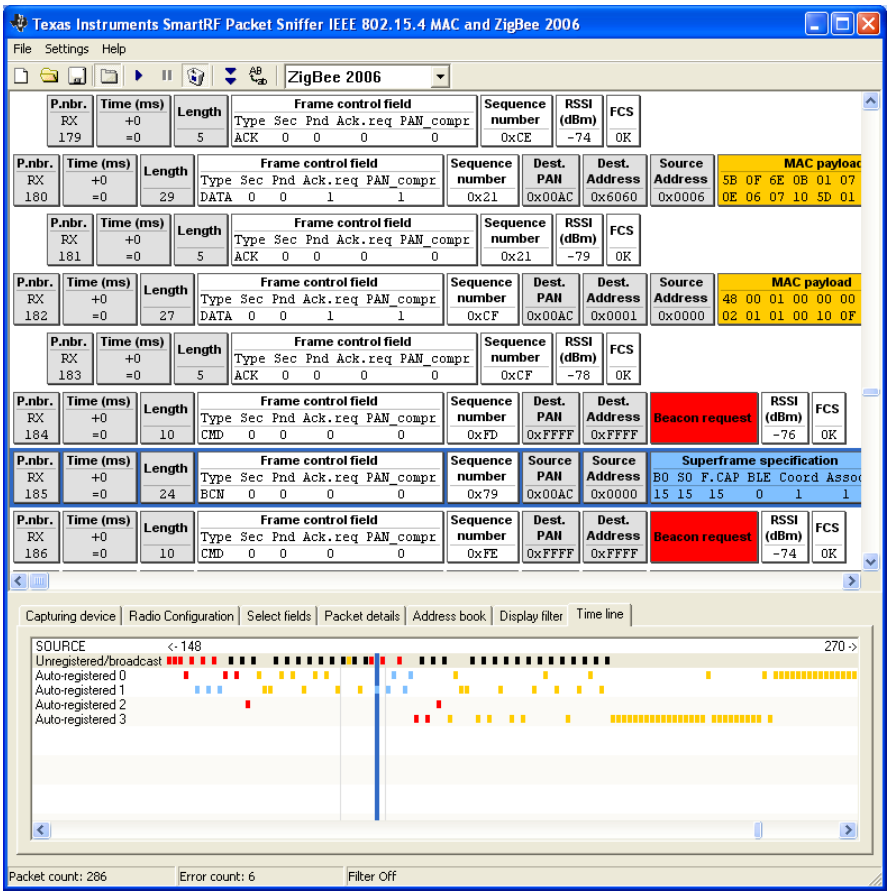
SmartRF™ Packet Sniffer User Manual



SWRU187F

Table of contents

1 Introduction

The SmartRF™ Packet Sniffer is a PC software application used to display and store RF packets captured with a listening RF HW node. Various RF protocols are supported. The Packet Sniffer filters and decodes packets and displays them in a convenient way, with options for filtering and storage to a binary file format.

The Packet Sniffer is installed separately from SmartRF® Studio, and must be downloaded from the Texas Instruments web site.

A shortcut for all supported signalling protocols will be placed on the Windows "Start menu" after the installation.

1.1 Protocols

The supported protocols can be seen in the Launch window when starting the packet sniffer. The following combinations of protocols and HW (RF Device) are supported:

Protocol	Version	Capture device	Can be used to capture packets from
Bluetooth® low energy	Bluetooth core spec 4.0	CC2540 USB Dongle CC2540EM+SmartRF05EB	CC2540 Bluetooth® low energy devices
ZigBee	2007/PRO 2006 2003	CC2531 USB Dongle CC2530EM+SmartRF04EB/SmartRF05EB CC2520EM+SmartRF05EB CC2430EM+SmartRF04EB/SmartRF05EB CC2431EM+SmartRF04EB/SmartRF05EB CC2430DB	CC2420 CC2430 CC2431 CC2480 CC2520 CC2530 CC2531
	2003	CC2420EM+CC2400EB	CC2420
RF4CE	ZigBee RF4CE 1.0.1	CC2531 USB Dongle CC2530EM+SmartRF04EB/SmartRF05EB CC2520EM+SmartRF05EB CC2430EM+SmartRF04EB/SmartRF05EB CC2431EM+SmartRF04EB/SmartRF05EB CC2430DB	CC2533 CC2530 CC2531
SimpliciTI	1.1.1 1.1.0 1.0.6 1.0.4 1.0.0	CC2531 USB Dongle CC2530EM+SmartRF04EB/SmartRF05EB CC2520EM+SmartRF05EB CC2430EM+SmartRF04EB/SmartRF05EB CC2431EM+SmartRF04EB/SmartRF05EB CC2430DB	CC2430 CC2431 CC2520 CC2530 CC2531
		CC1110EM[1]+SmartRF04EB/SmartRF05EB CC1111 USB Dongle CC1101EM+TrxEB CC110LEM+TrxEB CC113LEM+TrxEB	CC1100 CC1100E CC1101 CC110L CC115L CC1110 CC1111 CC430
		CC2510EM+SmartRF04EB/SmartRF05EB CC2511 USB Dongle	CC2500 CC2510 CC2511
		CC1120EM+TrxEB CC1121EM+TrxEB	CC1120 CC1121
Generic	Any	CC2531 USB Dongle CC2530EM+SmartRF04EB/SmartRF05EB CC2520EM+SmartRF05EB CC2430EM+SmartRF04EB/SmartRF05EB CC2431EM+SmartRF04EB/SmartRF05EB CC2430DB	CC2420 CC2430 CC2431 CC2480 CC2520 CC2530 CC2531 CC2533
		CC1110EM+SmartRF04EB/SmartRF05EB CC1111 USB Dongle	CC1100 CC1100E CC1101 CC110L CC115L CC1150 CC1110 CC1111 CC430
		CC2510EM+SmartRF04EB/SmartRF05EB	CC2500

[1] When sniffing in the sub-1 GHz frequency bands, you need hardware that supports the operating frequencies. Also note that CC1110 and CC1111 have limited support for some frequencies supported by CC1100E.

		CC2511 USB Dongle	CC2550 CC2510 CC2511
		CC1120EM+TrxEB CC1121EM+TrxEB	CC1120 CC1121

Table 1: Supported protocols

1.2 Hardware Platform

The packet sniffer can be used with different HW platforms. The following HW can be used:

- CC2400EB + CC2420EM
- CC2430DB
- SmartRF04EB + (CC2430EM, CC2530EM, CC1110EM or CC2510EM)
- SmartRF05EB + (CC2430EM, CC1110EM, CC2510EM, CC2520EM or CC2530).
- TrxEB + (CC1101EM, CC110LEM, CC113LEM, CC1120EM or CC1121EM)
- CC2531 USB Dongle.
- CC Debugger + SmartRFCCxx10TB
- CC2540 USB Dongle.

The applicable board must be connected to the PC through USB.

Figure 1: CC2400EB + CC2420

Note:
The Packet Sniffer started when selecting “IEEE802.15.4/ZigBee (CC2420)” will be different from the others. A different GUI application will be started. The most important difference is that the packets will only be stored in a RAM buffer. That means the GUI application will not be able to handle more packets when the buffer is full. See the user manual for “Packet Sniffer CC2420” for more details. The manual can be found under the “documentation” option of the start menu: start->Texas Instruments->Packet Sniffer->Documentation->Packet Sniffer CC2420 user manual

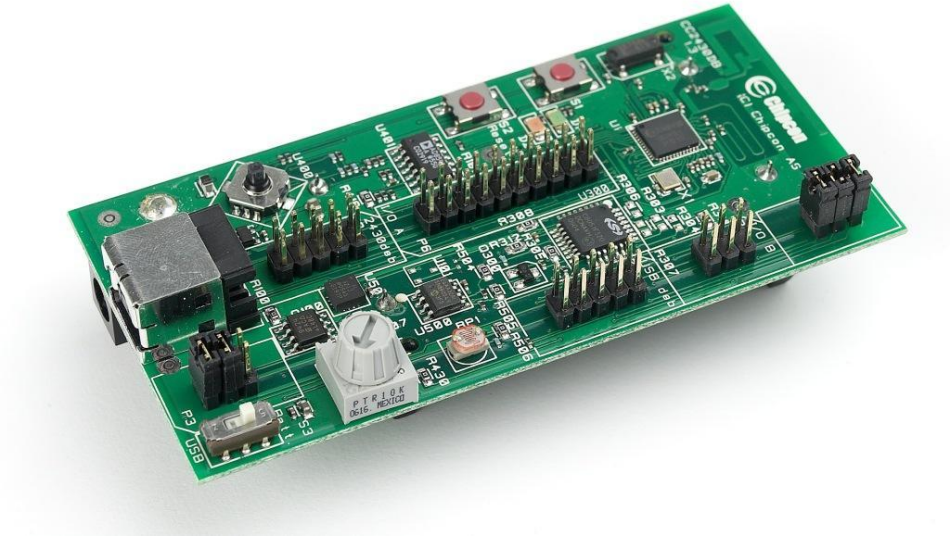


Figure 2: CC2430DB

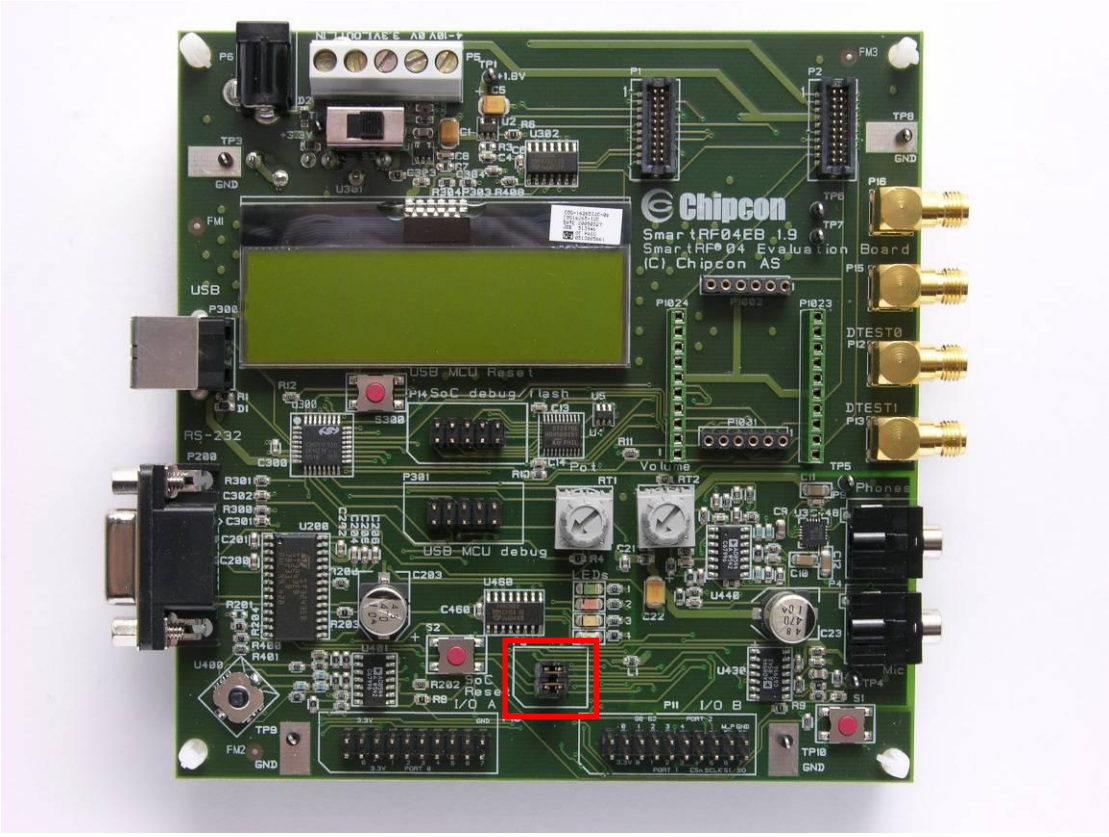


Figure 3: SmartRF04EB

Note:
Observe the jumpers on header P3. For revision A and B of CC2430 (register CHVER ≤ 0x02), the jumpers must be set in the horizontal direction (in parallel with the display) like in Figure 2. For newer revisions of CC2430, the jumpers should be set in the vertical direction (this is the default position).

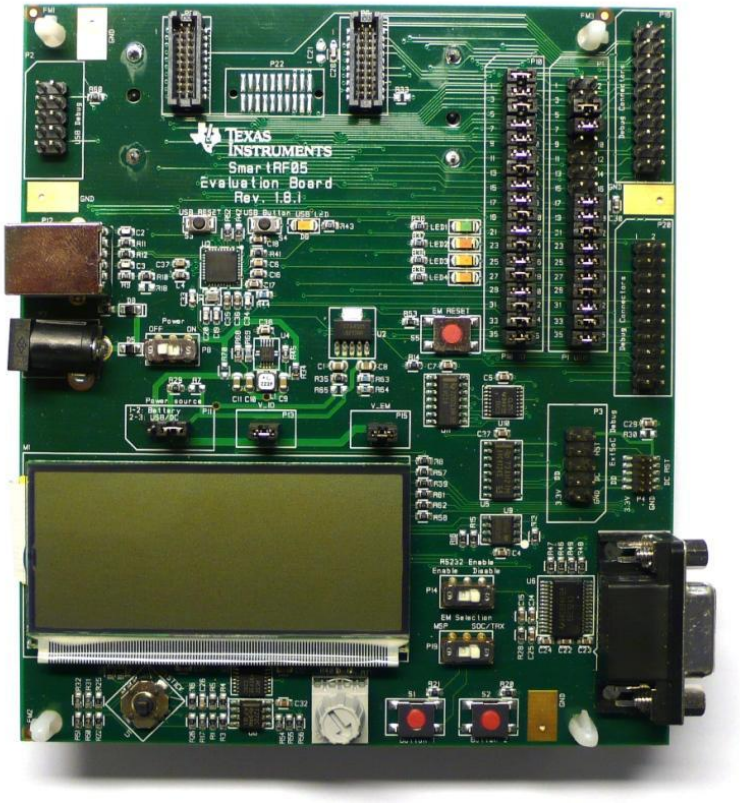


Figure 4: SmartRF05EB

Figure 5: CC2511/CC1111 USB Dongle

Both the CC2511 and the CC1111 USB Dongle can be used as capturing device with the packet sniffer. The Dongles must be pre-programmed with special firmware in order to work with the packet sniffer. After installation of the packet sniffer, the hex file to be programmed can be found on the following directory: *<installation directory>\General\Firmware\ sniffer_fw_ccxx11.hex*
The firmware can be programmed with the SmartRF Flash Programmer. To program the firmware on the dongle, it must be connected to SmartRF05EB or the CC Debugger via the debug connector. See user manual for the flash programmer for details on how to use the flash programmer.

Figure 6: CC2531 USB Dongle

The CC2531 USB Dongle must be pre-programmed with special firmware in order to work with the packet sniffer. After installation of the packet sniffer, the hex file to be programmed can be found on the following directory: *<installation directory>\General\Firmware\sniffer_fw_cc2531.hex*
The firmware can be programmed with the SmartRF Flash Programmer. To program the firmware on the CC2531 Dongle, it must be connected to SmartRF05EB or the CC Debugger via the debug connector. See user manual for the flash programmer for details on how to use the flash programmer.

Figure 7: CC Debugger + SmartRFCCxx10TB

Figure 8: CC2540 USB Dongle

The CC2540 USB Dongle must be pre-programmed with special firmware in order to work with the packet sniffer. After installation of the packet sniffer, the hex file to be programmed can be found on the following directory: *<installation directory>\General\Firmware\sniffer_fw_cc2540.hex*
The firmware can be programmed with the SmartRF Flash Programmer. To program the firmware on the CC2540 Dongle, it must be connected to SmartRF05EB or the CC Debugger via the debug connector. See user manual for the flash programmer for details on how to use the flash programmer.

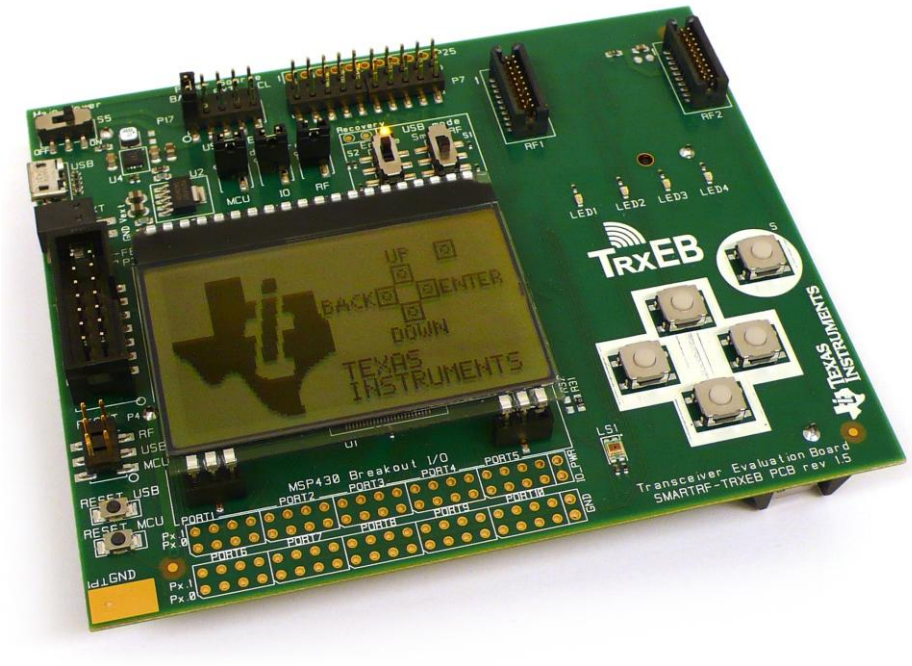


Figure 9, TrxEB

The USB MCU on the TrxEB board must be programmed with FW revision “0014” or newer. The revision can be seen in the SmartRF Flash Programmer or SmartRF Studio when the board is connected to the PC with a USB cable.

1.3 Data flow

On the PC side the packets will be stored in a disk buffer. The total amount of packets that can be stored depends on the packet size and the size of the hard disk. During operation the packets will be cached in a RAM buffer to improve the access time when a packet is to be displayed in the GUI.

Figure 10: Dataflow for the packet sniffer (SoC) below shows the data flow for the packet sniffer.

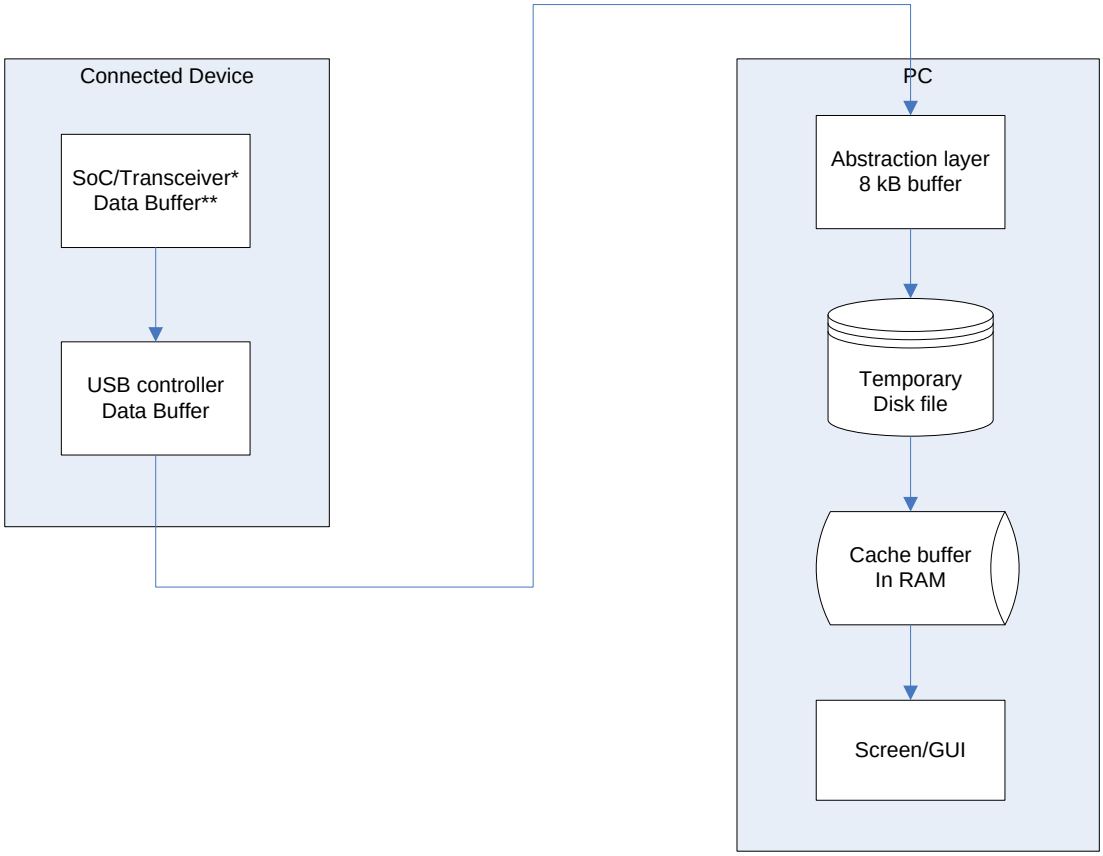


Figure 10: Dataflow for the packet sniffer (SoC)

* For the moment it is only the CC2520 transceiver that is supported by the packet sniffer.
** The buffer is only applicable for the System on Chip devices (SoC).

If the PC application is not able to read the packets from the connected devices data buffer fast enough, an “Overflow” error will be given by the device and the packet sniffer will show the error on screen.

1.4 Software

The firmware on the SoC’s required to run the sniffer will be checked and loaded automatically, if needed, when the sniffer is started. This can be seen on the status bar in the lower left corner.

The same apply for the USB controller, but the user will be asked to do the update and the user has the possibility to reject the update. In that case it might be that the sniffer is not working properly.

The following operating systems are supported:

- Windows XP/Pro (32 bit)
- Windows Vista (32 bit)
- Windows Vista (64 bit)
- Windows 7 (32 bit)
- Windows7 (64 bit)

2 User Interface

2.1 Launch Window

To select between the different options for protocol and HW configuration, a launch window will be shown when starting the sniffer.

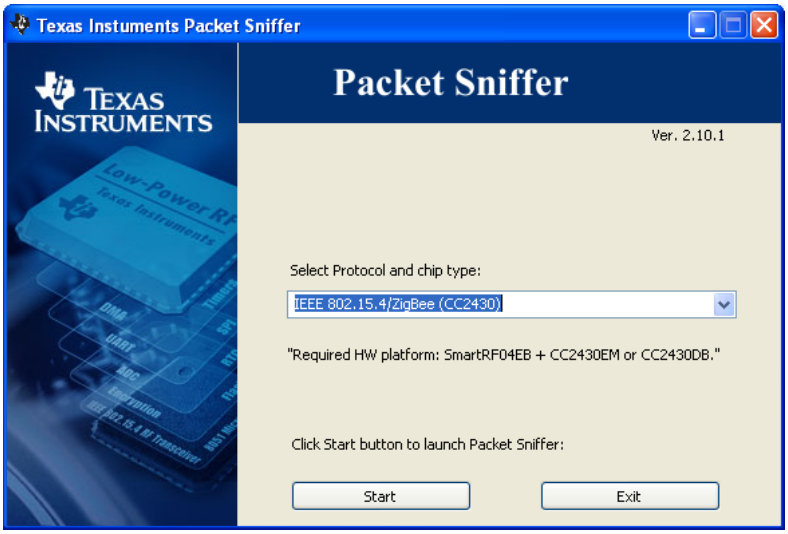


Figure 11: Screen shot of the packet sniffer launch window

To start the packet sniffer a combination of protocol and chip type should be selected. Then the start button should be clicked. If a packet sniffer session has been started and the launch window is closed, the sniffer session will still remain active and must be closed explicit if required.

2.2 Packet Sniffer window of an active session

The main window of the packet sniffer can be divided into two sections:

- At the top: A **packet list**, which displays the various fields of the decoded packets.
- At the bottom: The following seven tabs:
  - **Capturing device:** Selects which capturing device to use.
  - **Radio Configuration:** Input data to configure the radio of the capturing device. E.g. Channel number for IEEE 802.15.4 devices.
  - **Select fields:** Select which fields to display in the packet list.
  - **Packet details:** Displays additional packet details (e.g. raw data).
  - **Address book:** Contains all known nodes from the current session. Addresses can be registered automatically or manually, and they can be changed or deleted.
  - **Display filter:** Packet filtering with user defined filter conditions. A list of all fields which can be used to define the filter condition is given. From this list a filter condition can be defined by combining these fields with AND and OR operators.
  - **Time line:** Displays a large sequence of packets, about 20 times as many as in the packet list, and sorted by either source or destination addresses.

The packet sniffer screenshot in Figure 12 shows an example from the IEEE802.15.4/ZigBee protocol.

The status bar displays the total (unfiltered) number of captured packets, the number of packets with errors (checksum error and the number of occurrences of buffer overflow) and the status of the filter function. If filter is on, it will show the number of packets which have passed current filter conditions.

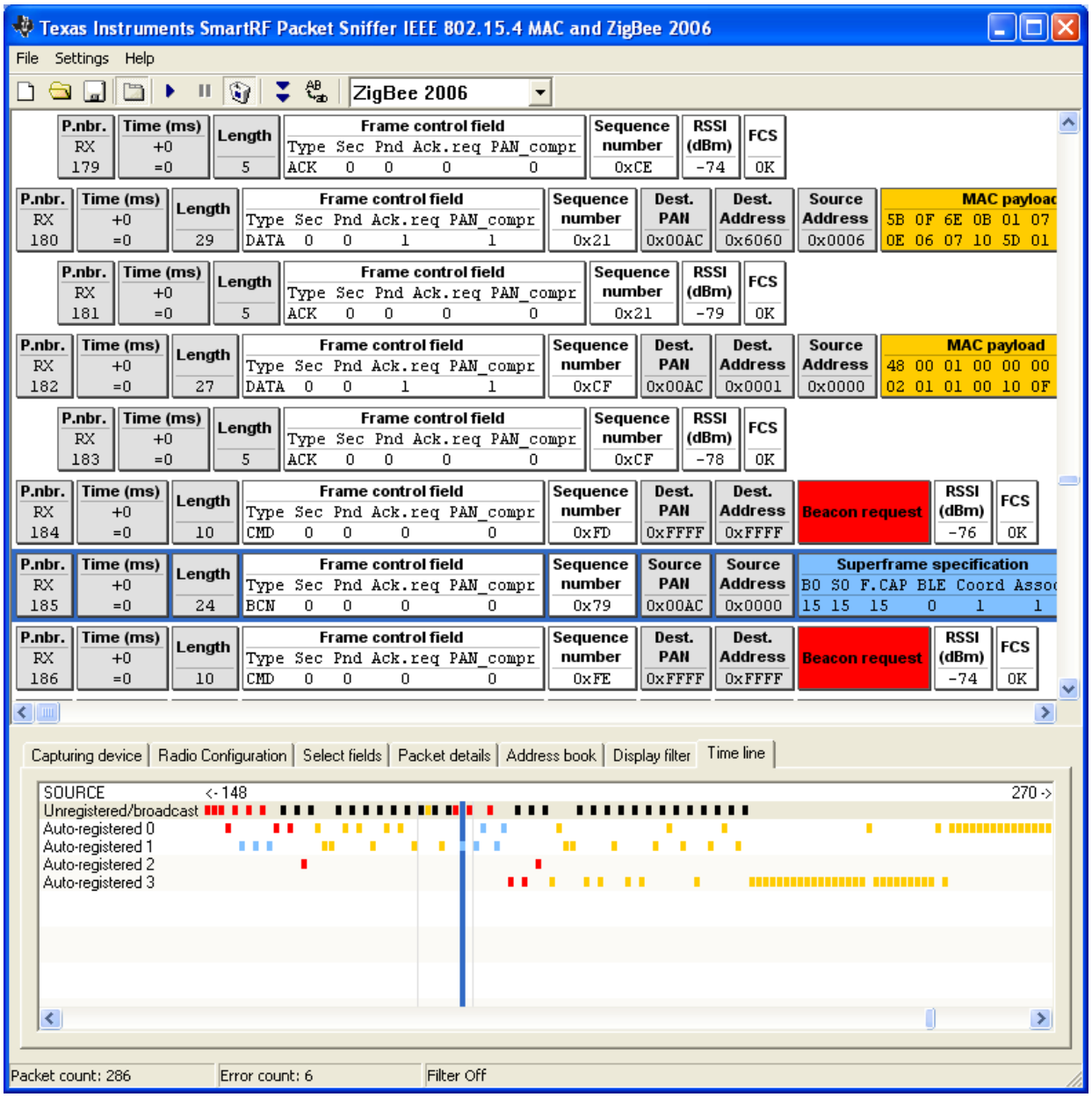


Figure 12: Packet sniffer screenshot from the IEEE8022.15.4/ZigBee protocols

2.3 Menus and Toolbars

Menu	Button	Key	Description
File→Reset...			Empties the packet buffer and the packet list.
File→Open data...			Load packet buffer from file
File→Save data...			Save packet buffer to file
			* Display the tabs at the bottom of the window.
		F5	Start the packet sniffer (does not empty the buffer)
		F6	Pause the packet sniffer
			* Delete all captured packets when starting
			Switch automatic scrolling on/off
			Switch between normal font or small font in the packet view window.
Settings → Cash buffer size…			* The size of the packet RAM buffer in megabytes.
Settings → Clock multiplier…			* A clock multiplier, which allows you to compensate for clock speed differences on the connected device and the hardware running the network application
Settings → Packet broadcast ...			* Configuration of packet broadcast. Enable/disable the feature, select broadcast address and UDP port number, select broadcast only
Help → About the PSD file format			Help on the file format used to save data.
Help→User Manual			Opens this document in your PDF file viewer
Help→Rev. History			Revision history (bug fixes, new features, etc.)

The application is closed by double-clicking in the top left corner, or single-clicking on the X-symbol in the top right corner.

Items marked with a star (*) are saved to Windows registry between each session.

For the ZigBee and SimpliciTI protocol there are options to select the protocol version. This can be seen in the toolbar as a drop down list: 

. The selected version will be saved between each session.

Cash buffer size:

The cash buffer is a RAM buffer that is allocated to contain packets that is displayed by the packet sniffer. It is used to optimize the access time when the GUI asks for information to display a packet.

The cash buffer function tries to anticipate which packets that will be requested next and will try to load the buffer with these packets in the background.

Clock multiplier:

A clock multiplier, which allows you to compensate for clock speed differences on the connected device and the hardware running the network application (Synchronizing the clock on the sniffer device with the clock on the network devices).

Example:
Ensure that the time stamps are given in microseconds to get accurate numbers (default). Measure a known time interval (e.g. the distance between a few beacons for the IEEE 802.15.4 protocol). Divide the desired value by the real value, and enter this floating-point factor into this field. **Clock multiplier:**

2.4 Packet broadcast.

From version 2.14 of the General Packet Sniffer it is possible to stream packet data to other applications via a UDP-port. This feature can be configured from the *Settings -> Packet broadcast* menu. The feature can be disabled altogether, and it is possible to select broadcast *IP-address* and *UDP port*. The IP-address is restricted to the '*local'* interface to avoid inadvertently flooding the network with packets. Optionally the operator may choose to *broadcast data only*. This is useful when capturing data over time for storage as it avoid the problem of the buffers of the Packet Sniffer GUI application filling up.

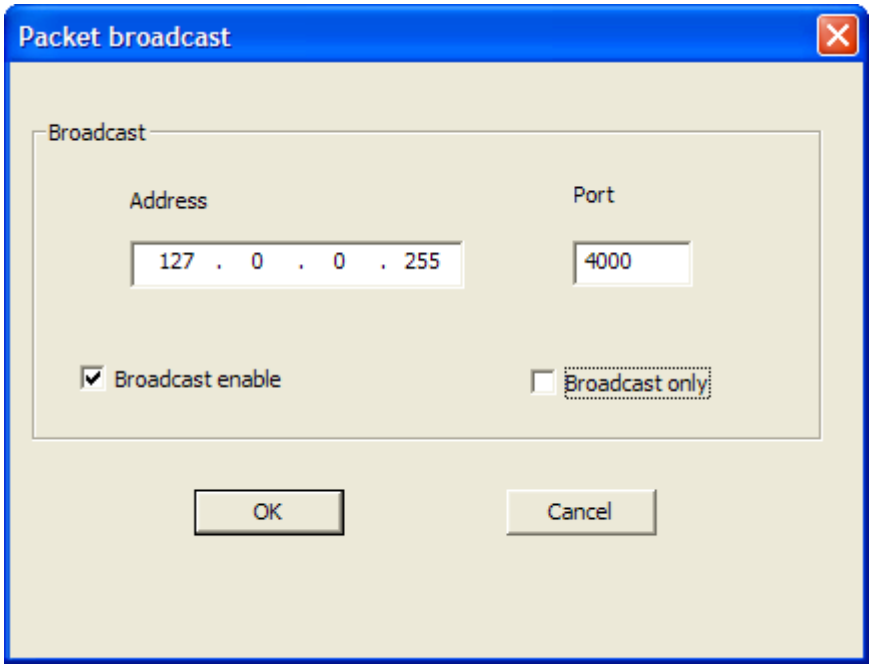


Figure 13: Packet broadcast setup

A PERL-script is provided as an example of how to use the broadcast data. It is located in <*installation directory*>*scripts****udp_receive.pl***.* It received data on port 4000 and displays the packets in hexadecimal format. Please note that the packet format is the same as the PSD-format described in chapter 5 *Format of packets saved to file* . To save a PSD-file, the application will have to pad the payload so it makes up the total number of bytes given in the <protocol>.plt file (Located in the plugin folder). The file contain "Packet_length_raw_data=n" where n is the buffer size excluding the packet info byte. E.g.: "Packet_length_raw_data=150" means that the total number of bytes for each packet should be 151.

P.nbr.	Time (us)	Length	Payload	RSSI (dBm)	LQI	FCS
	+0		61 88 00 07 20 61 19			
1	=0	16	60 19 04 CC C0 FF 01	-31	236	OK

Figure 14: Sample packet in Packet Sniffer GUI

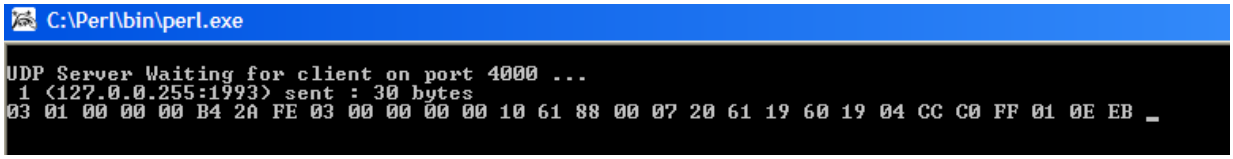


Figure 15: Sample packet as shown by *udp_receive.pl*

2.5 Capturing device

The Capturing device tab is used to select the required device.

Depending on the selected protocol, the applicable devices will be shown in the list. The list will automatically be updated when an applicable device is connected to a USB connector.

The Capturing device must be selected before the packet sniffer can be started (which is done by clicking the tool bar button, or hitting the F5 key).

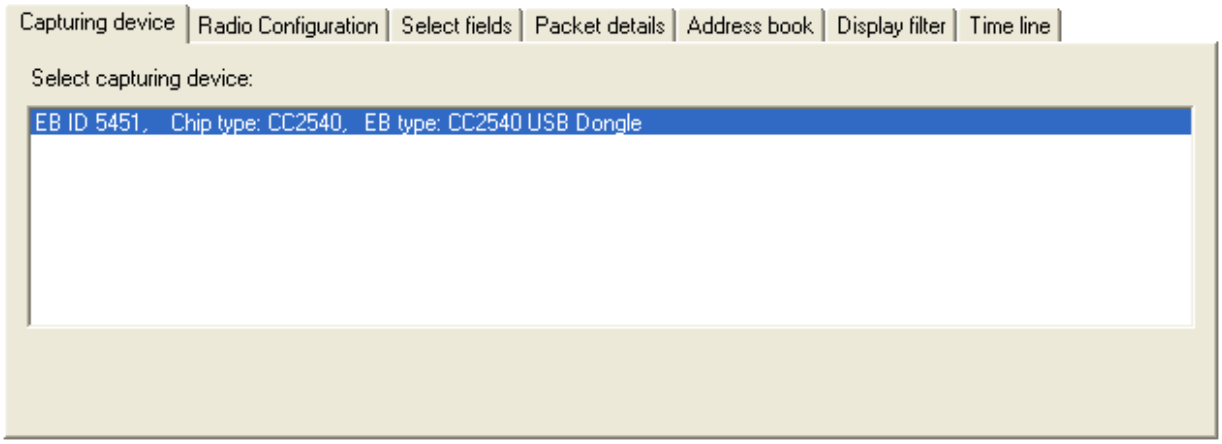
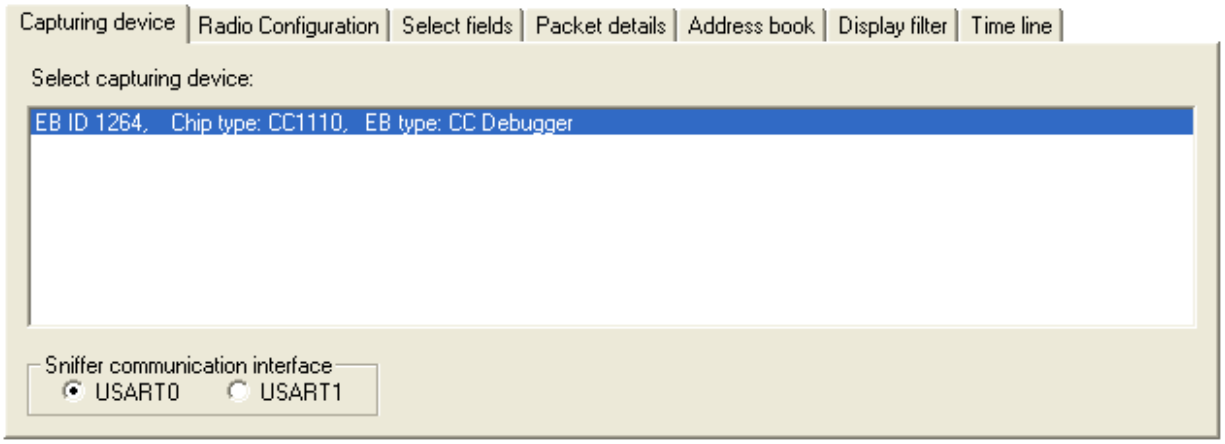


Figure 16: Capturing device

Sniffer communication interface:

If the EB type is CC Debugger and the chip type is CC1110 or CC2510, an additional option will be visible in the Capturing device panel. The sniffer communication interface must be selected.

The default value is USART0 and is applicable when CC Debugger is used together with the SmartRFCCxx10TB board. See figure below.



USART1 should be used for all other combinations with CC1110EM or CC2510EM.

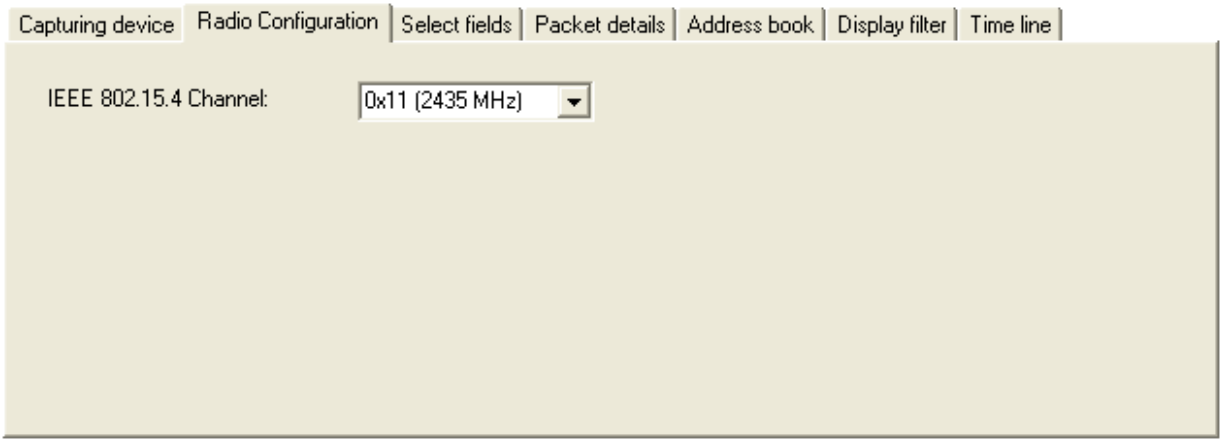
2.6 Radio Configuration

The Radio Configuration tab is used to select the parameter values required to configure the radio of the capturing device.

The parameters depend on the selected capturing device.

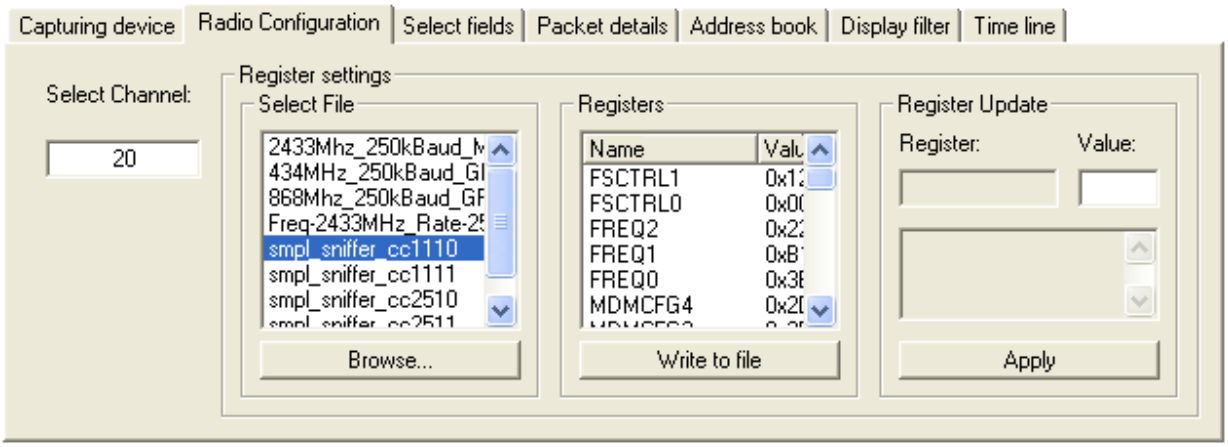
IEEE 802.15.4 devices:

For IEEE 802.15.4 devices, the required channel must be selected.



Proprietary devices:

These radios support a lot of programmable RF parameters.



The radio settings should be given in a text file. The file can be created by SmartRF® Studio. This makes it easy to get all the correct register settings calculated by SmartRF® Studio. See chapter 6 for further details.

After installation of the SmartRF™ packet sniffer, a default file will be available in the subdirectory of the applicable plugin.

The format of the file is shown in the example below:

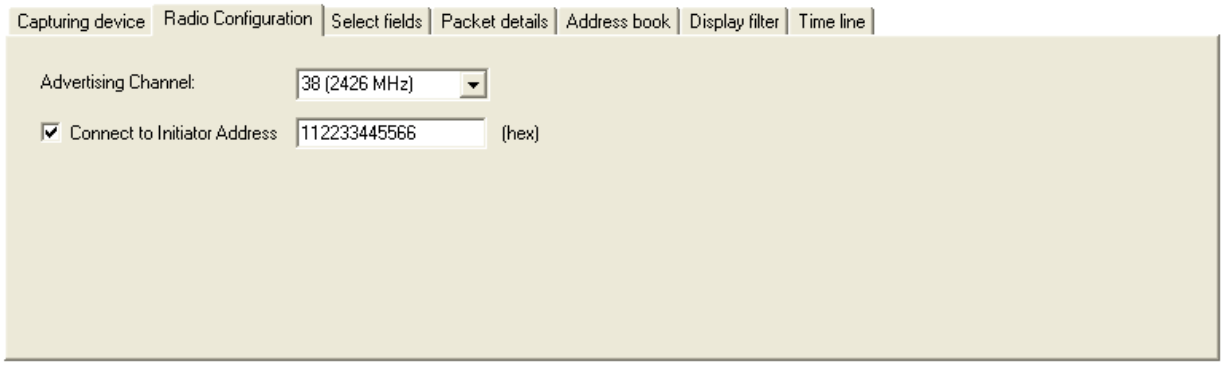
```
#Name    |Addr. |Val.|Description
#--------+------+----+-------------------------------------------------------------
PKTLEN   |0xDF02|0xFD|Packet length.
PKTCTRL1 |0xDF03|0x04|Packet automation control.
PKTCTRL0 |0xDF04|0x05|Packet automation control.
FSCTRL1  |0xDF07|0x07|Frequency synthesizer control.
FREQ2    |0xDF09|0x1C|Frequency control word, high byte.
FREQ1    |0xDF0A|0x80|Frequency control word, middle byte.
FREQ0    |0xDF0B|0x00|Frequency control word, low byte.
MDMCFG4  |0xDF0C|0x2D|Modem configuration.
MDMCFG3  |0xDF0D|0x3B|Modem configuration.
MDMCFG2  |0xDF0E|0x73|Modem configuration.
MDMCFG1  |0xDF0F|0x22|Modem configuration.
MDMCFG0  |0xDF10|0xF8|Modem configuration.
DEVIATN  |0xDF11|0x00|Modem deviation setting (when FSK modulation is enabled).
MCSM1    |0xDF13|0x0C|Main Radio Control State Machine configuration.
MCSM0    |0xDF14|0x10|Main Radio Control State Machine configuration.
FOCCFG   |0xDF15|0x1D|Frequency Offset Compensation Configuration.
BSCFG    |0xDF16|0x1C|Bit synchronization Configuration.
AGCCTRL2 |0xDF17|0xC7|AGC control.
AGCCTRL1 |0xDF18|0x00|AGC control.
AGCCTRL0 |0xDF19|0xB2|AGC control.
FREND1   |0xDF1A|0x56|Front end RX configuration.
FREND0   |0xDF1B|0x10|Front end RX configuration.
FSCAL3   |0xDF1C|0xA9|Frequency synthesizer calibration.
FSCAL2   |0xDF1D|0x0A|Frequency synthesizer calibration.
FSCAL1   |0xDF1E|0x00|Frequency synthesizer calibration.
FSCAL0   |0xDF1F|0x11|Frequency synthesizer calibration.
PA_TABLE0|0xDF2E|0xFF|PA Output Power Setting.
```

When the file is selected, the register values will be shown in the “Registers” frame.

To modify the register value, double click the register name. The register name will appear in the “Register update” frame. The value can be changed in the “Value” field. Click on “Apply” to use the new value. The changes can be seen in the “Register” frame. The new values can be written to file with a click on the “Write to file” button.

Bluetooth Low Energy devices:

For Bluetooth Low Energy devices, the Advertising channel must be selected.



The capture device can be configured to follow a data connection between a specific Bluetooth low energy master (initiator) and slave device. In the "Radio Configuration tab", click the checkbox next to the "Connect to Initiator Address" and write the address of the initiator (master) device. If this option is not selected, the capture device will start following the first data connection that appears on the current advertising channel

2.7 Select fields

The field selection tab can be used to select which fields to display and which to hide in the packet list. This feature is particularly useful for low-resolution screens (less than 1024x768). The fields are grouped in several colour coded categories.

The time stamp can be displayed in microseconds or milliseconds. The payload data can be displayed as hex-bytes or as plain-text. In plain-text format all non-printable characters will be replaced by a "*".

The Selected Fields list box gives the possibility to select predefined field groups. It is also possible to select “all” or “none”.

Each frame can be shown either with its LQI (Link Quality Indication, ranging from 0x00 to 0xFF) or RSSI (Received Signal Strength Indicator with an approximation to the actual RF level, in dBm). The LQI parameter is derived from the IEEE802.15.4/ZigBee protocol specification. The exact definition will depend on the used protocol.

The example below shows the fields defined for the SimpliciTI protocol.

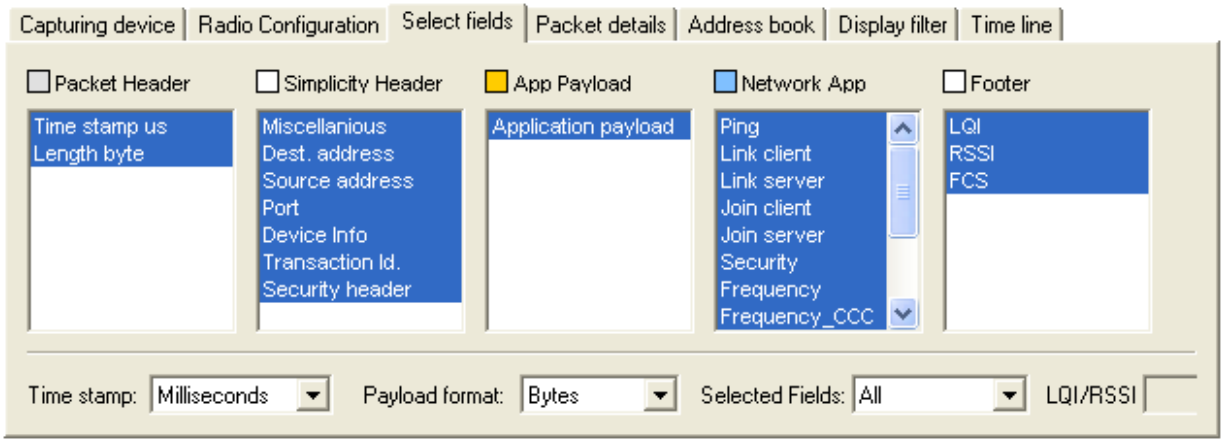


Figure 17: Select Fields panel

Tips:
Extended selection is used to operate the controls:

- To select a range of fields:
  - Click and drag over the fields that should be selected, or....
  - ... select the first field, hold down the "Shift" key, and select the last field.
- To select/unselect a single field:
  - Hold down the "Ctrl" key and click on the field to be toggled.

2.8 Packet details

By double-clicking on a packet in the packet list, additional details, as shown below, will be displayed. This example show details from the SimpliciTI protocol.

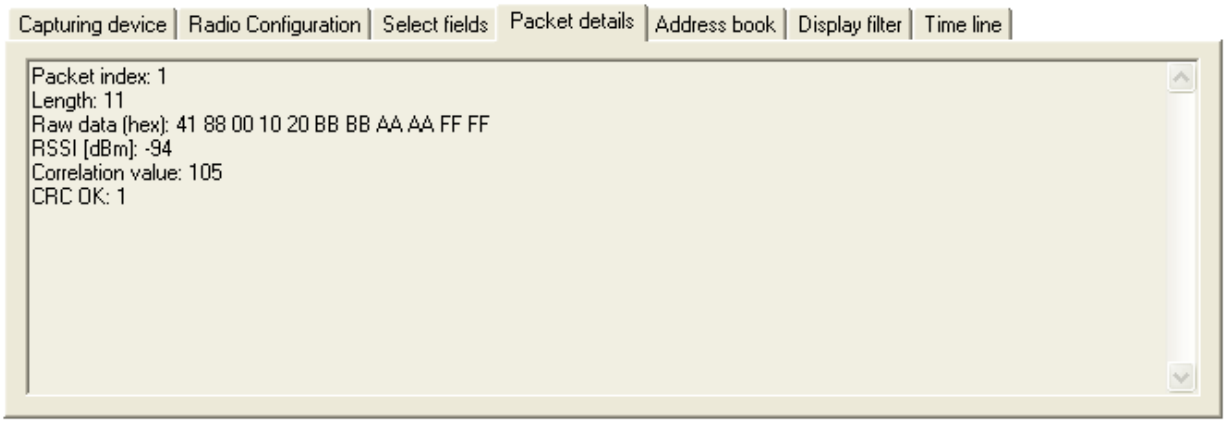


Figure 18: Packet details panel

The packet index shows the index for each captured packet, starting with index 1 for the first packet.

The RSSI value is read from the connected device and adjusted with a given offset value to get an approximate value in dBm. The correlation value is equal to the value read from the connected device. See the datasheet of the connected device for detailed information on the RSSI and correlation values.

2.9 Address book

The address book contains all known Node addresses from the most recent session. By selecting "Auto-register" (on by default), the packet sniffer will register all addresses automatically and add entries into the address book. The example below show the fields defined for the IEEE 802.15.4/ZigBee protocols.

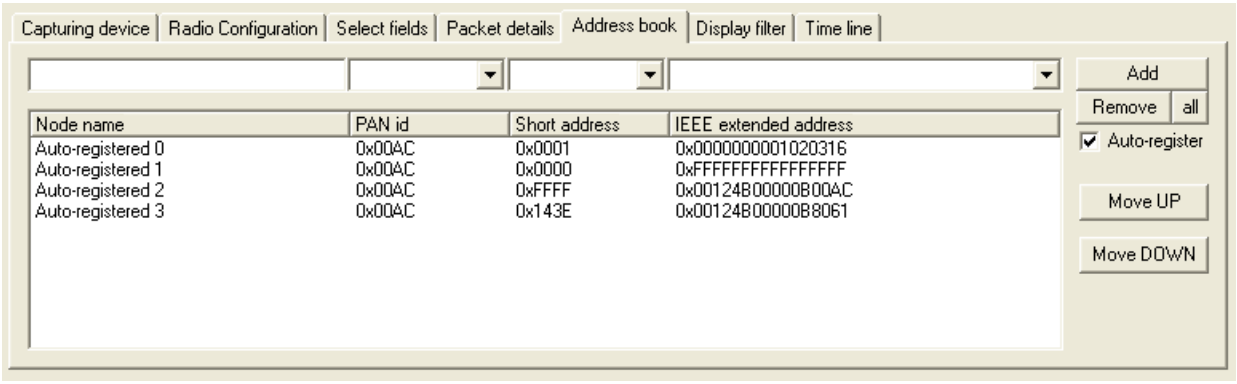


Figure 19: Address book panel

Nodes are added/replaced manually by clicking the "Add" button, or by pressing the "Enter" key while standing in one of the top fields.

Nodes can be removed by clicking the "Remove" button, or by pressing the "Delete" key while a node is selected in the address list.

Nodes can be moved up/down by using the rightmost buttons, or the "Alt + U" and "Alt + D" key combinations.

Depending on the protocol it may be required with manual editing of the fields in the address book to correct for address conflicts.

Below are examples where manual editing for the IEEE 802.15.4 protocol are given.

- There has been a PAN ID conflict.
- A device has left the network, and another device has been given an already used short address (the extended address will be replaced).
- Association response commands have not been detected.

Tips:
Fast editing of node names can be done using the following procedure:

1. Select the first auto-registered item in the address list
2. Hit "Enter" to copy the data and move to the node name field
3. Enter the new name
4. Hit "Enter" to replace the old entry and move back to the address list
5. Move one line down by using the down arrow.
6. Go to step 2

2.10 Display filter

The Display filter tab allows for filtering on all fields defined in the Field Name window.

A template is provided to ease the definition of the filter condition. The template will show the short name for each field. If the field has sub fields, the definition of all sub fields will be shown within brackets. Some fields will be dependent of other fields. This will also be shown in the template.

When the required field is selected, the template can be moved to the single line Filter condition window by pushing the First button or the And button. The “First” button will remove all existing conditions and set current template as the first condition. The “And” button will add current template to the existing conditions and the conditions will be evaluated with the AND operator. When the template has been moved to the single line Filter condition window, it must be modified to give the real values of the requested fields. The value is indicated with an “x” in the template. If the filter condition of a field with sub fields only requires evaluating the first sub field value, the sub field value can be given without brackets.

Example:

Figure 20 below shows an example from the IEEE 802.15.4/ZigBee protocols. The template of the Frame control field is:

FCF=(Type=x, Sec=x, Pnd=x, Ack_req=x, Intra_PAN=x)

If it is only required to test on the “Type” sub field, the condition can be simplified the following way:

FCF=BCN

Note: This is only possible for the first field in the definition of sub fields.

It is not required to fill in the values of all fields. If the filter condition only requires checking some of the sub fields, only those fields should be given.

Example:

Same as previous example, but this time it includes checking the “Type” and “Ack_req” values. This condition should be given as follows:

FCF=(Type=BCN, Ack_req=1)

When all conditions that should be evaluated with the AND operator are defined, the condition can be moved to the multi line Filter condition window with the Add button (“Enter” will give the same result). In this window several filter conditions can be added and all the conditions in “vertical” direction will be evaluated with the OR operator. To summarize: Conditions in the horizontal direction are evaluated with the AND operator. Conditions in vertical direction are evaluated with the OR operator.

To remove a line from the multi line filter condition, select the required line and click the Remove button. To remove all the lines from the multi line filter condition, click the All button.

The button Apply filter will activate the filter and the packet window will be redrawn with packets that comply with the given filter condition. The Turn off filter button should be used to disable the filter function. The packet window will be redrawn again and all packets will be visible. The filter function can be enabled and disabled while the sniffer is running.

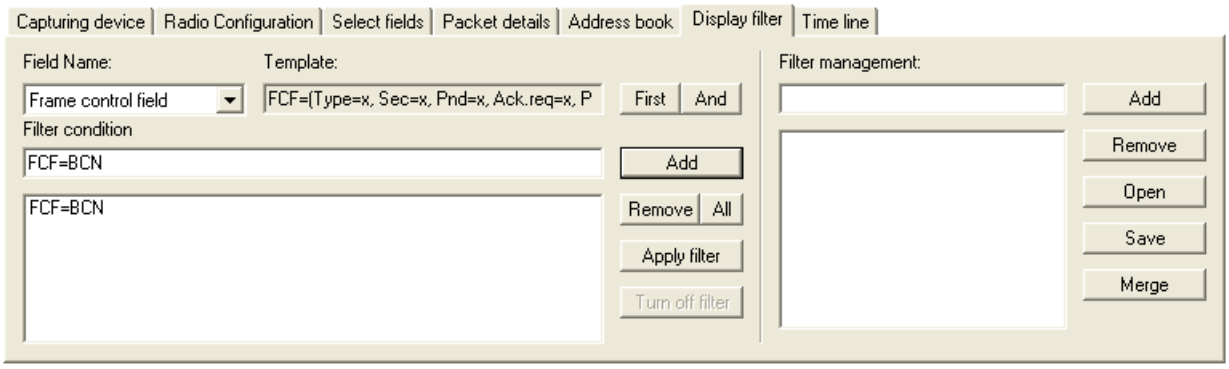


Figure 20: Display filter panel

Filter management:

On the right side of the panel, a Filter management function is provided. The filter management contains a database of defined filters. The database can be saved to a file and read from a file. The file is formatted as a plain text file and can be manually updated if required.

Below is an example of a filter database file. The filter name is given within square brackets [ ] and the filter conditions are given on the following lines. In this example the filter condition is: Dest. Address = 0x2430 OR 0x1749

```
[address]
DAD=0x2430
DAD=0x1749
```

To add a filter definition to the database, the Add button should be used. In order to add the filter to the database, a filter name must be given in the window to the left of the button. When the name is given and the “Add” button is pushed, the current filter definition (the multi line window of the filter condition) will be added to the filter database. The name of the filter will appear in the list of filters (Window below the filter name).

To remove a filter, select the required filter in the list of filters and push the Remove button.

The filter database can be read from a file with the Open button. To save the filter database to a file, use the Save button.

To add filter definitions from a file, without deleting the existing filters in the database, the Merge button should be used. This will open the given file and add the filters to the existing filter database. If the given filter name already exist in the filter database, the name will be modified with an additional digit at the end of the name.

To use a filter from the filter database, double click on the filter name and the filter condition will appear in the multi line filter condition window at the left side.

Note:
When packets are filtered out, the delta times shown in the Time fields still show the delta time to the previous packet captured, not the previous packet shown.

2.11 Time line

The time line displays all received packets, ordered horizontally by the time of reception and vertically by source or destination address. Selecting a packet from the time line will instantly be reflected in the packet list, and vice versa, thus allowing for efficient navigation in large collections of packets.

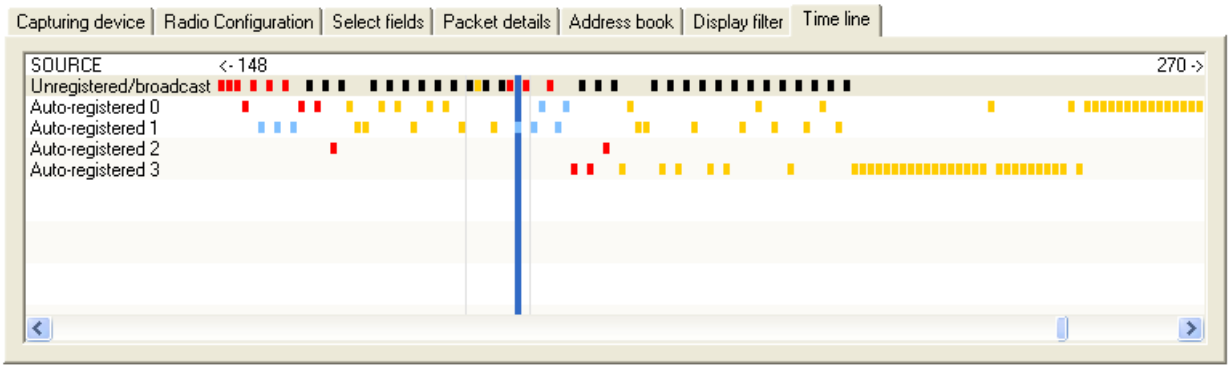


Figure 21: Time line panel

Double-click in the left section of the time line to switch between destination and source.
Packets are selected by clicking and/or holding down the left mouse button.
The time line can be scrolled clicking and holding down the right mouse button (drag).

3 Encrypted Payload

Decryption of encrypted data is only supported by the Bluetooth low energy packet parser.

3.1 How to use the decryption feature

1. Copy the file “ltk.txt” with the Long-term key (LTK) to the root directory “c:\”.
   The file can be found in the “BLE” plugin directory.
   Typically: C:\Program Files\Texas Instruments\Packet Sniffer\General\Plugin\ble

2. Modify "ltk.txt" with the right ltk. The format is MSO-LSO.
   E.g. If the ltk=0x00112233445566778899AABBCCDDEEFF
   The file will have to be "00112233445566778899AABBCCDDEEFF" as its very first line

3. Run the sniffer like normal. Encrypted packets will be decrypted and flagged as "Encryption Enabled". Payload and MIC will be displayed on the GUI.

3.2 Limitations

Decryption is supported with the following limitations:

1. The decryption will fail if one or more packets are sent but failed to be captured by the capturing device. The decryption algorithm depends on the timing, packet counters (one for each side) and the direction of the packet. There are algorithms in the parser to determine these parameters but they can’t capture all the scenarios where one or more packets missing.

4 Know Issues

4.1 Bluetooth Low Energy

The capture device currently ignores the connection timeout parameter for an active connection. This means that the sniffer will not know that a connection between two BLE devices is "down" if no new packets are received for the duration of the connection timeout. The reason this is not supported by the sniffer, is to remedy the case where the sniffer follows a data connection between two remote devices and thus is likely to lose a number of packets for a period of time that exceeds the connection timeout.

When the actual connection is terminated due to a connection timeout, the sniffer must be stopped (click the pause/stop icon) and restarted (click the play icon) in order to follow a new connection.

5 Format of packets saved to file

The figure below describe the packet format for packets saved to a Packet Sniffer Data(PSD) file. The number of bytes is given for each field.

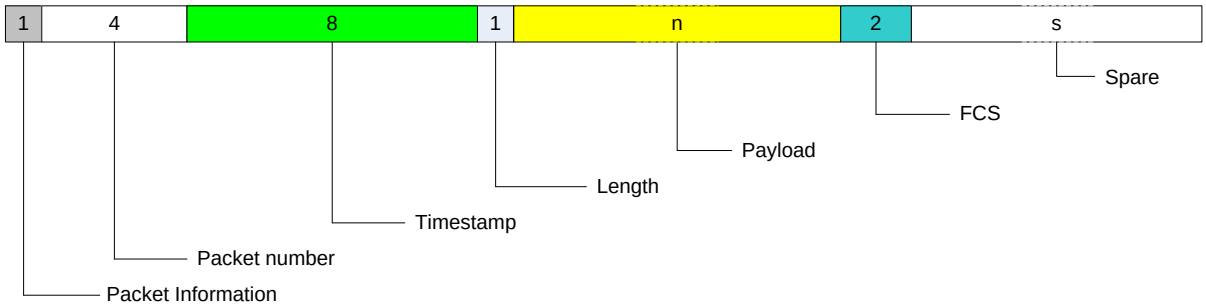


Figure 22: Packet format in PSD file

Packet Information:
New field introduced from version 2.3.0. Contains information used by the packet sniffer to read the data correctly.
Bit 0: Length includes FCS.
Bit 1: Correlation used.
Bit 2: Incomplete packet.
Bit 3-7: Not used.

Timestamp:
64 bit counter value. To calculate the time in microseconds this value must be divided by a number depending on the clock speed used to drive the counter tics on the target. (E.g. CC2430EM -> 32, CCxx10 -> 26, SmartRF05EB + CC2520EM -> 24). The timestamp on the first packet will be used as offset value for all packets. That means that packet number 1 will be shown in the packet sniffer with time = 0.

Length:
The length will or will not include the FCS field depending on Bit 0 in the Packet information.

Payload:
Packet Information Bit 0 = 0 → n = Length
Packet Information Bit 0 = 1 → n = Length – 2

FCS:
The checksum of the frame has been replaced by the radio chip in the following way:

BYTE 1: RSSI and if Correlation used, this byte is also used to calculate the LQI value.
BYTE 2: Bit 7: Indicate CRC OK or not.
Bit 6-0: If Correlation used: Correlation value.
If Correlation not used: LQI.

See data sheet for the applicable chip for further details.

Spare:
The number of spare bytes depends on the total amount of bytes used by the packet sniffer to save the packet. The number of bytes depends on the protocol and can bee seen from the description of the packet format under the help menu.

6 Exporting register settings from SmartRF™ Studio

SmartRF™ Studio and its user manual can be downloaded from the Texas Instruments web site. See the SmartRF™ Studio user manual for more details.

When the correct device has been selected, a list of preferred register settings will be shown in the "Normal view" tab. After selecting the preferred register settings and optionally changing any of the register values, choose "File/Export CCxxxx code…" from the menu to start the code export. This will open the following window:

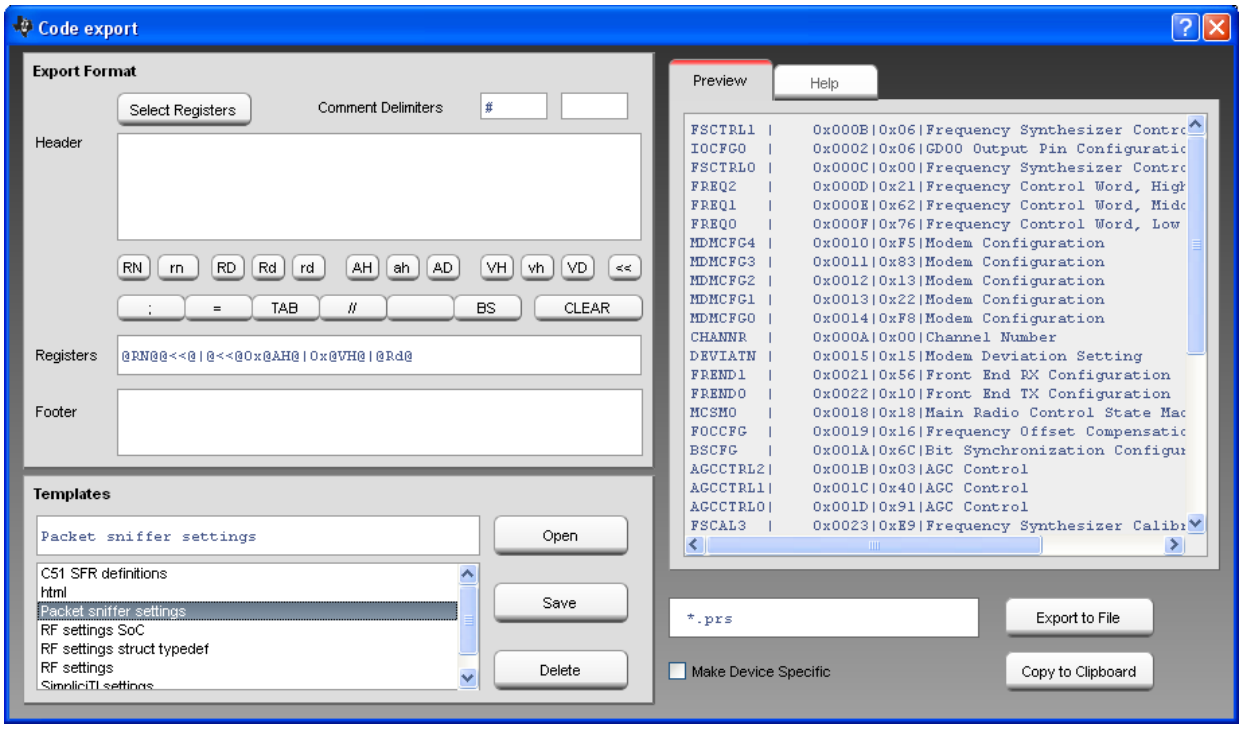


Figure 23: SmartRF Studio Code Export

Select "Packet sniffer settings". The register settings with correct formatting can be seen in the "Preview" tab to the right. Select "Export to file" to save the settings to file.

7 Help

The packet sniffer provides help through so-called tool-tips. By moving the cursor over a field (e.g. a button or a text field) and holding it in the same position for about half a second, the text will appear in a yellow box slightly below the cursor:

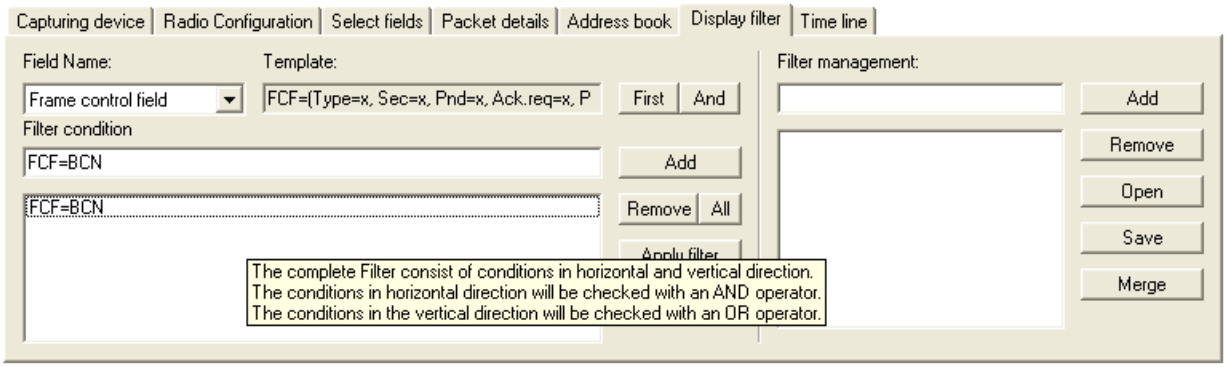


Figure 24: Display filter panel with the tool-tip shown

Keyboard and mouse button events will cause the tool-tip to disappear, or not be displayed at all.

8 Troubleshooting

This section contains some troubleshooting tips that should be used if the packet sniffer does not function as expected. Execute the steps one by one until the problem is solved.

A) The evaluation board is not detected (it does not appear in the list box in the Setup tab)

Using smartRF04EB + CC2430EM

- Make sure that the USB cable is connected and that a CC2430EM is mounted.
- Check that the jumper between I_OUT and I_IN is connected.
- Check that the jumpers on P3 are mounted correctly (see the CC2430DK user manual).
- Press the "Reset" button.

Using 2430DB

- Make sure that the USB cable is connected and power switch is in USB position.
- Check that the jumper on P3 pin 1-2 is mounted
- Check that all tree jumpers on P5 are mounted
- Press the "Reset" button.

B) When pressing the “start button”, the following message is given: "Not able to Start Sniffer. Try upgrade of USB firmware".

- Check if latest Firmware version of the USB controller is used. This should be version 0037 or later. This can be seen by using the SmartRF Flash Programmer from Texas Instruments. The figure below show an example from the flash programmer. The column “EB firmware rev” will show the version. The Flash Programmer can be download from the Texas Instruments web site.

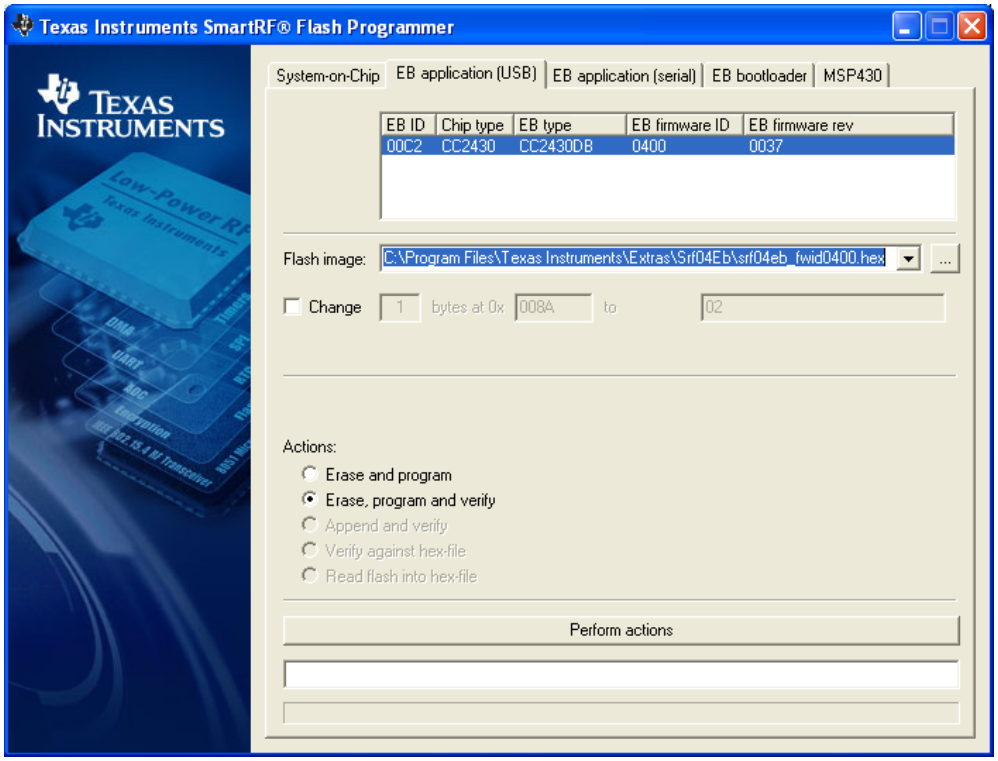


Figure 25: SmartRF Flash Programmer

C) When pressing the "start button", the sniffer stops immediately (the start button is not grayed out)

- Disconnect the USB cable from the SmartRF04EB or CC2430DB board, and plug it back in.
- Press the "Reset" button on the board.
- Disconnect the power cable from all evaluation boards and install the latest version of the packet sniffer.
- Reboot the computer.

D) The program does not respond

- Press the "Reset" button on the connected Evaluation Board (EB).

E) The packets are not decoded correctly

- Packets with an FCS failure will probably not be parsed correctly (FCS = ERR).
- Make sure that the packet really is correctly formatted (compare the fields with the raw data in the packet details tab).

F) Weird packets appear in the packet sniffer when not transmitting anything

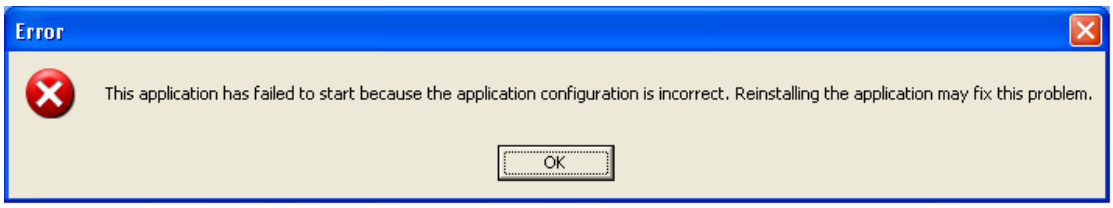
- CC2430 will try receiving packets down to the RF noise floor. Sometimes it will also decode packets which are decoded from noise only. These will appear in the packet sniffer. To avoid this, enable FCS filtering in the toolbar.

G) The packet sniffer stays “idle” and does not receive any packets after “start button” has been pressed.

- Check that correct channel is used in the setup panel.
- Check the jumper settings, see section 1.2 in this document. If a SmartRF04EB board with an CC2430EM module is used, make sure that the jumpers are set correct depending on the chip revision. Revision A and B (register CHVER ≤ 0x02), horizontal. Later revisions, vertical.

H) Error message when trying to start the packet sniffer.

If an error message about missing msvcp80.dll appears, or the error message shown below appears when attempting to start the application, you may be required to install an additional package from Microsoft.



The package contains some additional runtime components needed by applications developed with Visual C++. To resolve this problem, download the file vcredist_x86.exe from the URL below and install the package.

http://www.microsoft.com/Downloads/details.aspx?FamilyID=32bc1bee-a3f9-4c13-9c99-220b62a191ee&displaylang=en

9 General Information

9.1 Document History

Revision	Date	Description/Changes
F	07/07/2011	Added new HW platforms: - TrxEB + one of the following EM's: CC1101, CC110L, CC113L, CC1120 or CC1121
E	14/03/2011	Description of Packet Broadcast added.
D	22/10/2010	The table with combinations of protocols+HW has been updated. A picture of the HW platform for CC2420 is added.
C	25/08/2010	Document updated with information about capturing device used for the Bluetooth Low Energy protocol. Screen shots have been updated.
B	03/11/2009	SmartRFCCxx10TB added as possible capturing device.
A	19/02/2009	Description for SmartRF05EB added.
-	03/09/2008	First revision.

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

TI products are not authorized for use in safety-critical applications (such as life support) where a failure of the TI product would reasonably be expected to cause severe personal injury or death, unless officers of the parties have executed an agreement specifically governing such use. Buyers represent that they have all necessary expertise in the safety and regulatory ramifications of their applications, and acknowledge and agree that they are solely responsible for all legal, regulatory and safety-related requirements concerning their products and any use of TI products in such safety-critical applications, notwithstanding any applications-related information or support that may be provided by TI. Further, Buyers must fully indemnify TI and its representatives against any damages arising out of the use of TI products in such safety-critical applications.

TI products are neither designed nor intended for use in military/aerospace applications or environments unless the TI products are specifically designated by TI as military-grade or "enhanced plastic." Only products designated by TI as military-grade meet military specifications. Buyers acknowledge and agree that any such use of TI products which TI has not designated as military-grade is solely at the Buyer's risk, and that they are solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI products are neither designed nor intended for use in automotive applications or environments unless the specific TI products are designated by TI as compliant with ISO/TS 16949 requirements. Buyers acknowledge and agree that, if they use any non-designated products in automotive applications, TI will not be responsible for any failure to meet such requirements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products		**Applications**	
Amplifiers	amplifier.ti.com	Audio	www.ti.com/audio
Data Converters	dataconverter.ti.com	Automotive	www.ti.com/automotive
DSP	dsp.ti.com	Broadband	www.ti.com/broadband
Clocks and Timers	www.ti.com/clocks	Digital Control	www.ti.com/digitalcontrol
Interface	interface.ti.com	Medical	www.ti.com/medical
Logic	logic.ti.com	Military	www.ti.com/military
Power Mgmt	power.ti.com	Optical Networking	www.ti.com/opticalnetwork
Microcontrollers	microcontroller.ti.com	Security	www.ti.com/security
RFID	www.ti-rfid.com	Telephony	www.ti.com/telephony
RF/IF and ZigBee® Solutions	www.ti.com/lprf	Video & Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265