Atmel AVR2104: RF4CE-EK Remote Control Evaluation Kit - User Guide

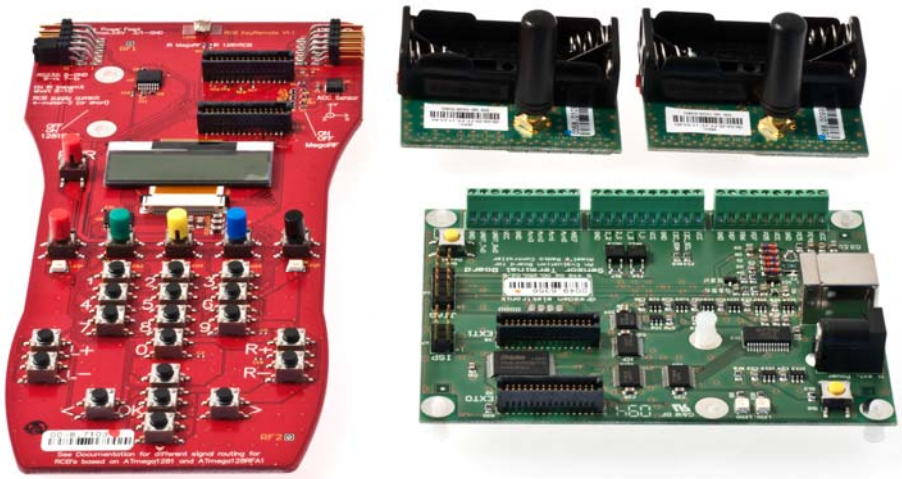
Features

- **Evaluation platform performance for the Atmel® ATmega128RFA1 using programmed performance test measurement application consisting of simple range measurement and packet error rate measurement**
- **Test various IEEE® 802.15.4 hardware embedded features**
- **Experiment with the enhanced feature set of ATmega128RFA1**
- **Verify the Atmel certified RF4CE software stack RF4Control**
- **Use serial boot loader functionality to flash new application software image into the MCU's program memory (for example, Atmel RF4CE Control software stack)**
- **Extendable to any IEEE 802.15.4 based wireless application development such as ZigBee®, 6LowPAN, ISA100, or any other proprietary network protocol**

1 Introduction

The Atmel RF4CE-EK is an evaluation kit that allows the setup and the operation of a simple ZigBee RF4CE network based on the ATmega128RFA1. With this kit the user can verify the performance of the chip and can work with the RF4Control software stack to build RF4CE remote control applications. The complete hardware contents of the kit are explained in Chapter 2.

Figure 1-1. RF4CE-EK hardware.



8-bit Atmel Microcontrollers

Application Note

Rev. 8391A-AVR-08/11

This document describes how to set up and run the pre-flashed Performance Test application included with the Atmel RF4CE Evaluation Kit. This initial test application is described in Chapter 3.

The Performance Test application includes a dedicated serial boot loader firmware which allows easy firmware update of the IEEE 802.15.4 [3] compliant Atmel ATmega128RFA1 SoC with the ZigBee RF4CE [1] certified stack Atmel RF4Control [2].

The use of this serial boot loader is covered in Chapter 4.

The document also explains set up and operation of the RF4Control example applications for RF4CE controller and target nodes using the reference design hardware. This primary operation of this RF4CE evaluation kit is explained in Chapter 5 of this document.

2 Hardware description

2.1 Kit contents

2 x RCB128RFA1 Radio Controller Board (RCB)
1 x RCB Sensor Terminal Board (STB)
1 x RCB Key Remote Control board (KEY_RC)
1 x RCB-BB RS232 cable
1 x USB cable
2 x 2.4GHz antenna
4 x AAA battery

2.1.1 The RCB128RFA1

The RCB128RFA1 is a small, battery-powered reference board that carries one Atmel ATmega128RFA1 System on Chip. It represents a system hardware reference for this microcontroller and contains a battery holder, SMA antenna connector and a 2x30 pin connector that allows the use of this board together with several different base boards to support various kinds of RF applications.

In this Atmel RF4CE-EK it serves the purpose of the RF communication interface module. With different applications loaded the board can be either used on the controller side as well as on the target side of an RF4CE network. The board can be also used to act as a simple RF performance evaluation module.

A detailed hardware description of this board can be found [12].

2.1.2 The RCB Key Remote Control board

The RCB Key Remote Control board is a remote control type base board that offers a typical remote control user interface like buttons, but also contains elements of more advanced remote controls like a display and LED's as event indicators.

It provides the standard 2x30 pin connector that is used on the Radio Controller Board platform and allows the connection of different types of RCB's to demonstrate different RF remote control applications and features.

This board has also external connectors for serial interface as well as a JTAG connector and can be used as development and programming base board for updating the firmware on the RCB's microcontroller.

A full description of the board is found in [6].

2.1.3 The Sensor Terminal board

The Sensor Terminal board is an interface board for the small form factor Radio Controller boards. It supports the 2x30 pin connector of the RCB platforms and provides several additional external interfaces like USB or GPIO's. It also provides a JTAG connector and therefore also allows the firmware update of RCB's plugged into the 2x30 pin connector.

It is used as interface board for enabling a user controllable RF4CE target application and for PC-based RF performance measurements with the RCB platforms.

A detailed hardware description of the Sensor Terminal Board is found in [10].

3 Performance Test Measurement Firmware Package

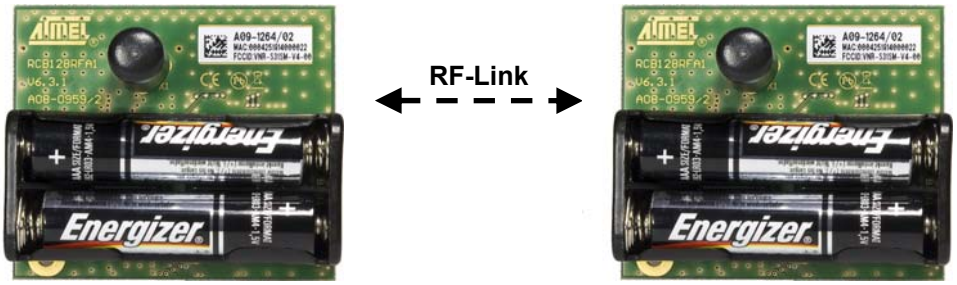
The main purpose of the Performance Test application is to offer a generic and simple to use tool to verify the RF performance of the Atmel ATmega128RFA1. This performance measurement firmware package is preprogrammed into every RCB128RFA1 contained in this kit. It can be operated in two different scenarios:

1. Simple Range Measurement – OP Mode 1.
   This mode is used to test simple transceiver performance when using only two RCB128RFA1. Each board is individually powered by batteries. The range measurement can be performed by simply setting both RCB's as communication peers and exchanging test frame data.
2. Packet Error Rate Measurement – OP Mode 2.
   This mode is used to perform more extensive transceiver tests. In this mode one RCB is operated stationary and it is connected to the host PC while the other is a mobile stand-alone RCB capable of performing basic range and terrain/environment survey measurements.

3.1 Simple Range Application

The RCB128RFA1 preprogrammed Performance Test Measurement application features an operation mode that is called Simple Range Measurement. After power up a RCB is in receive mode and ready to receive transmitted data packets from a transmitting node. One of the RCB's can be put into transmit mode to start the test. A successful transmit and receive operation is indicated by blinking LED's as the frames are transmitted and received by each board. This visual indication can be used for easy RF range measurements between two RCB128RFA1.

Figure 3-1. Simple Range Measurement Hardware Setup.



3.1.1 Application Setup

Insert two AAA batteries into each of the RCB128RFA1, connect the SMA stub antennas to the SMA connector of the board and apply power to both RCB's by turning on the power switch located on the top side of the board. The RCB's run a power-on check and indicate the successful completion by switching on LED D2.

3.1.2 Operational description

Pressing the T1 button on one RCB starts the Simple Range Measurement application. The application will then establish a dedicated RF link to the second RCB and after this first configuration phase it will automatically start the continuous transmission of data packets:

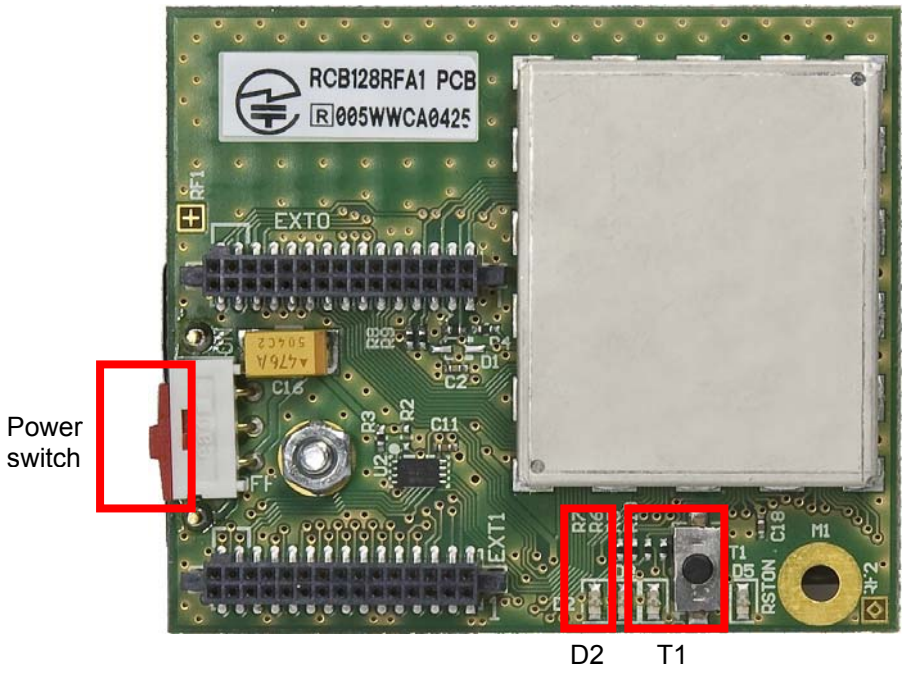
1. RCB initiates a configuration procedure by broadcasting frames on a predefined RF channel and is waiting for response from its peer RCB.
2. After a successful configuration, the initiating node is set as a transmitter node and its companion RCB as a receiver node.
3. Once the RF data link is established the transmitter node sends data frames to its communication peer.

Each frame transmission is indicated by toggling of the LED's on the transmitting node. A successful frame reception is indicated by toggling of the LED's on the receiver node. The LED's on the receiver node stop toggling if no or no valid frames are received. This is an indication that the RF link between the two nodes is no longer operational. The frame transmission can be stopped at any time by pressing T1 button for a second time on the transmitter node.

NOTE

The node configuration (peer-to-peer configuration) gets lost when the power to an RCB is switched off or the board gets a reset. To restart the Simple Range Measurement both RCB's needs to be reset or power cycled and button T1 must be pressed on that RCB that should operate as the transmitter node to restart data communication between the nodes.

Figure 3-2. Power switch, D2 LED and T1 button on RCB.



3.1.3 Purpose of the Simple Range Measurement Test

Simple Range Measurement test is meant to be used to test basic RF link quality and range possibilities between two RF units when out in the field in a given environment.

To get an indication about the transmission range a user can start the transmission by pressing button T1 on the RCB that should operate as transmitter and use the second node as mobile receiver. Good RF link quality is indicated by continuously toggling LED's of the receiver node. As soon as the receiving node is moved too far away from the transmitting RCB or due to high attenuation structures between transmitter and receiver node, the LED's stop toggling.

The LED's on the transmitter RCB toggle always on successful data transmission and gives the user the visual indication that still data packets are transmitted. The nodes can be put in different places in order to get a fast indication whether a RF transmission is possible between these two locations. A typical scenario is that the transmitting RCB is placed on a fixed location while the receiving RCB is used to walk around and verify the RF range by observing the toggling LED's.

3.2 Packet Error Rate Measurement (PER)

The second operation mode of the Performance Test Measurement application is the Packet Error Rate measurement test. The test and its basic parameters are defined in the IEEE 802.15.4 standard [3]. The test can be easily performed by simply connecting one of the RCB nodes via the STB to the Host-PC running a Terminal application. This provides the user with a simple configuration interface to control the test parameters. Individual performance tests can be configured and executed by menu options in the Terminal window.

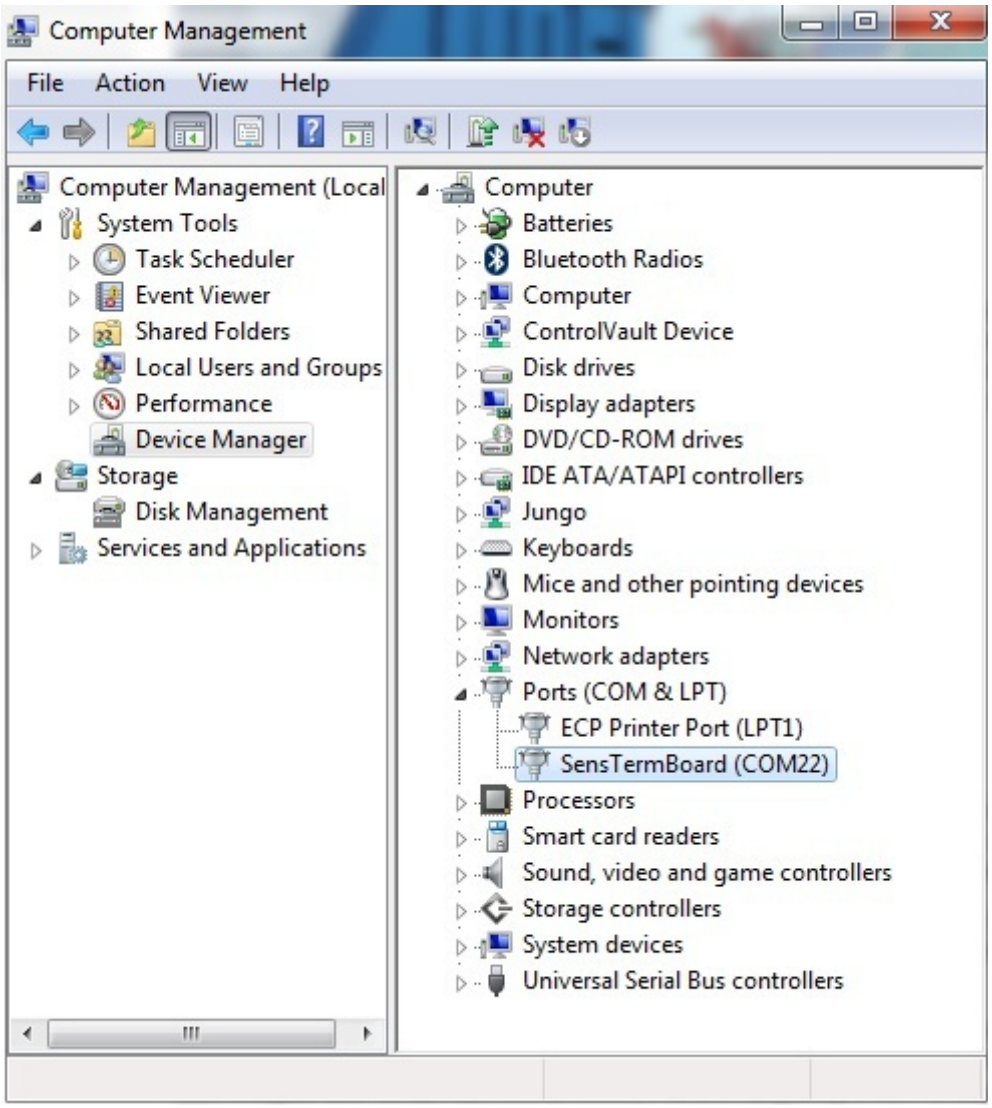
3.2.1 Setting and Configuring Kit Hardware

1. Insert two AAA batteries into a stand-alone RCB128RFA1. This RCB makes one peer node also referred to as a receiver node.
2. Mount the second RCB without batteries on top of STB by plugging it onto the EXT0 and EXT1 connectors. This RCB and STB board assembly makes a transmitter node. This node is used as stationary node and must be connected with the USB cable to the host PC running a Windows® XP or later operating system.
3. Download and install a USB driver (FT245RL) for the STB from [10].
4. Once the driver is downloaded unpack the driver archive to a folder on the host PC's hard-disk.
5. Connect USB cable to the STB. The New Hardware Installation Wizard will recognize the new board and will guide through the USB driver installation. When the Wizard is asking for the driver to install, navigate to the directory where the driver archive has been unpacked to.

A successful installation and USB enumeration with the PC is indicated by the power status LED (LED 2) on the STB.

6. Identify the new hardware in the Windows Device Manager. The assigned COM-port number is needed when configuring the Terminal application later. See Figure 3-3 for an example COM port assignment for the STB.

Figure 3-3. Windows Device Manager.



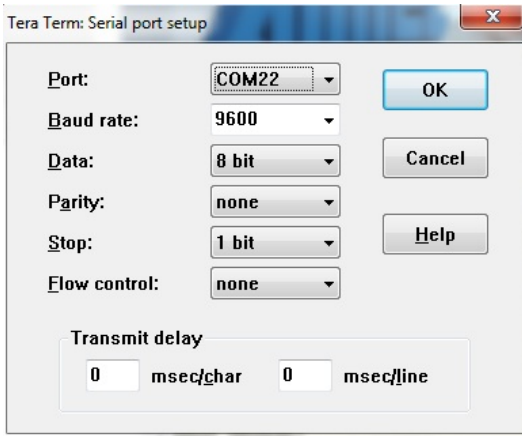
Once the Terminal application is launched it needs to be configured to communicate with the STB hardware via the corresponding COM port.

7. The Terminal program running on the Host-PC is used to control the application running on Atmel ATmega128RFA1 device of the STB-RCB board assembly.

   Set up the Terminal program parameters as below:

BAUD RATE:	9600
PARITY:	None
DATA BITS:	8
STOP BITS:	1
FLOW CONTROL:	Off

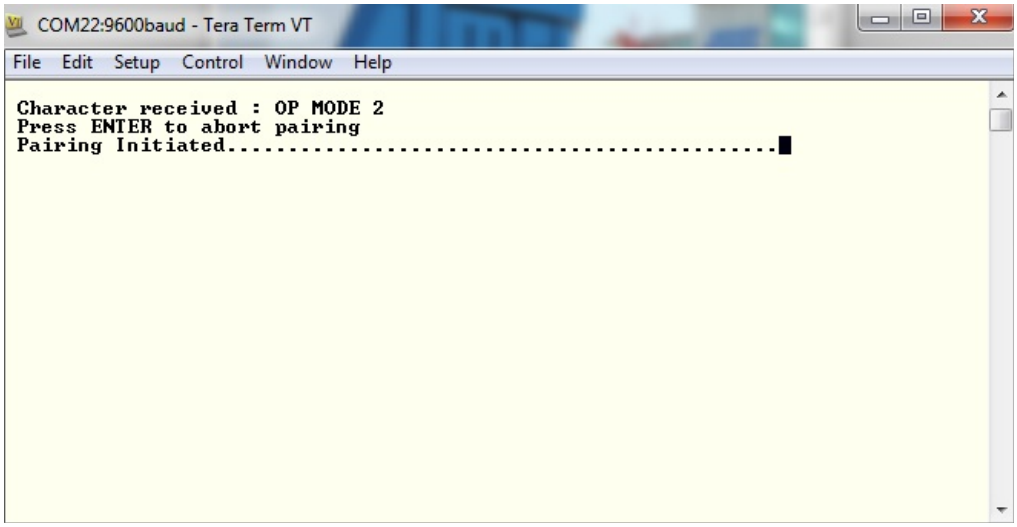
Figure 3-4. Terminal configuration.



Once the USB driver software is installed and the Terminal window is set up, a user can display the application menu by typing any character on the Host-PC's keyboard. The application menu is only displayed if the RF-link between both RCB nodes is established. This step is initiated as soon as the node powers up. It starts looking for the second RCB – this is shown in the Terminal window by means of a growing dotted line.

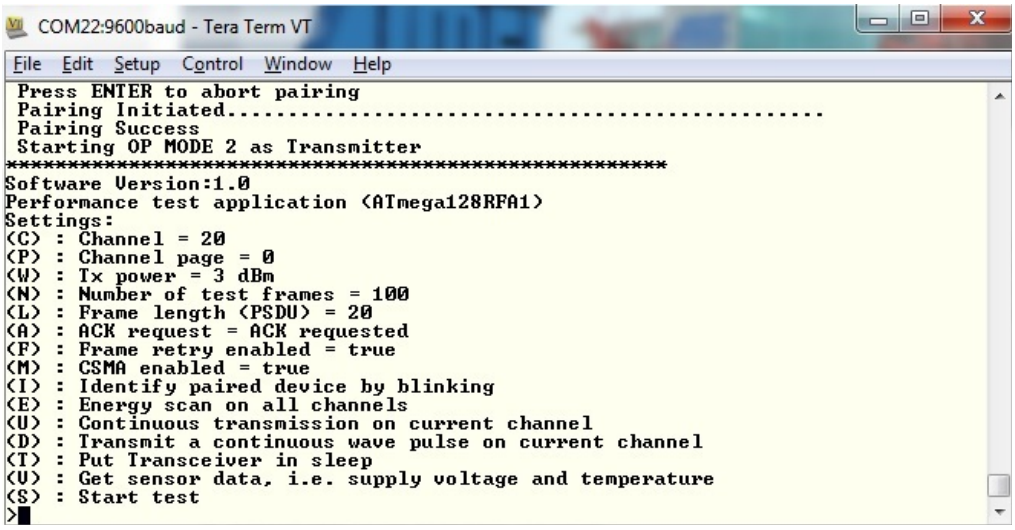
8. Switch on the second RCB to allow initial peer configuration.

Figure 3-5. Searching for peer device.



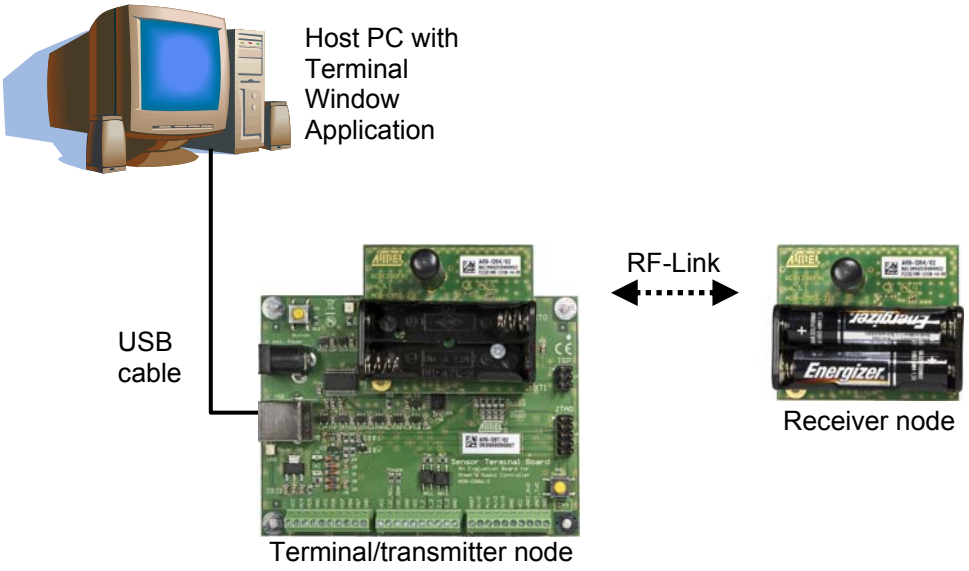
Once the second RCB is powered up the first node will recognize this and will establish a preconfigured link between those two nodes. The application will assign the device roles – the stationary STB-RCB setup will operate as the transmitter while the mobile node will act as a receiver node. After this step the command menu is displayed.

Figure 3-6. Successful configuration and command menu.



The complete setup will look like shown below.

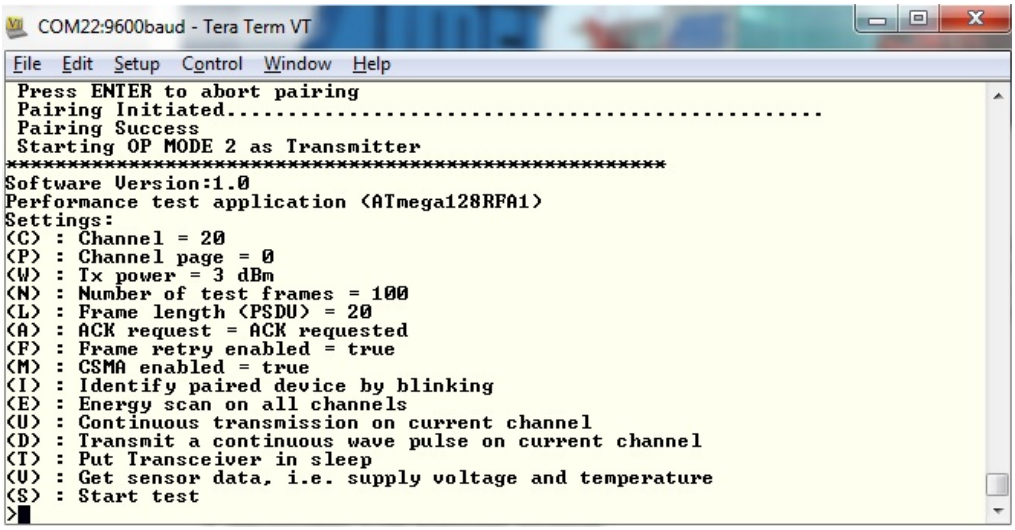
Figure 3-7. Complete setup.



3.2.2 Packet Error Rate Measurement Menu

The packet error rate measurement can be controlled using the menu options displayed in the Terminal window, see Figure 3-8. Configuration changes, single side tests and packet error rate measurements can be done by pressing the corresponding command key.

Figure 3-8. Packet Error Rate Measurement application menu.



3.2.3 Configuration parameters

The first entries in the Performance test application menu are basic parameters that can be modified to perform the PER test under different conditions like a different channel number or with different length of data packets. This includes all kinds of basic parameters required for basic RF performance tests. The list of all possible configuration parameters is found in Appendix, Table A-1.

3.2.4 Single side tests

Some of the command keys can initiate single side tests that are only using the stationary STB-RCB assembly. This includes Energy scan, Sleep mode or Continuous wave transmission tests. But since the two nodes perform an initial peer configuration sequence it is always required to have both RCB128RFA1 active – even for tests that will be performed with the stationary node only.

3.2.5 Start Test (Option S)

The menu option (S) starts a frame transmission to its peer device. Once this command key is used a basic PER test is performed with the configured test parameters as frame length, ACK request, retry setting, selected channel, and transmit output power. An example PER test sequence is described in Section 3.2.6.

3.2.6 A typical PER test sequence

Once the hardware is properly connected, the initial peer configuration sequence has been passed successfully and the Performance Test application menu is visible in the Terminal software the steps to initiate a basic PER test with a generic parameter set are as follows:

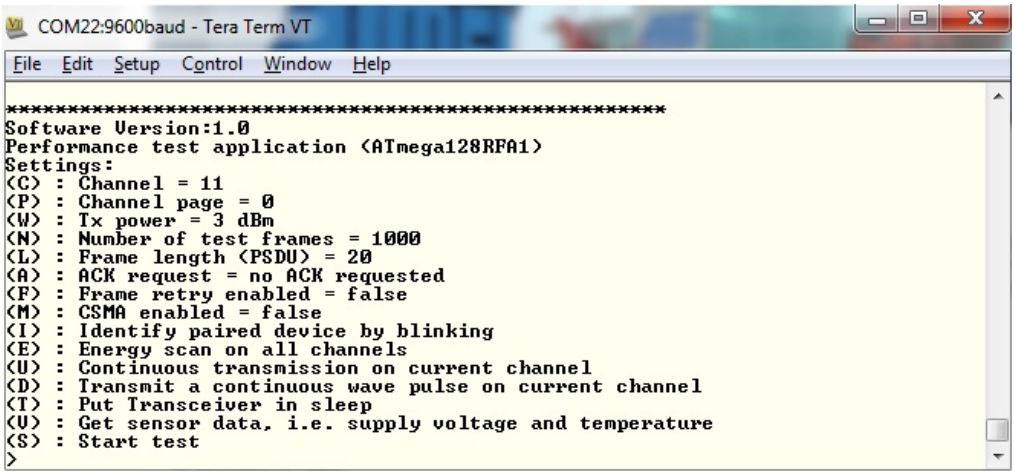
Parameter settings (parameters not listed here are left with their reset values):

1. Select channel: C.
   - Enter channel number, for example: 11
2. Select number of test frames: N.
   - Enter number of packets for this test, for example: 1000

3. Select frame length: L.
   - Set frame length to 20 bytes as this is the value defined by the IEEE 802.15.4 for a basic PER: 20
4. Select acknowledge (ACK) request option (enable/disable): A.
   - For the basic PER the acknowledgement should be disabled
5. Select frame retry option (enable/disable): F.
   - For the basic PER the frame retries should be disabled
6. Select CSMA option: M.
   - For the basic PER the CSMA should be disabled
7. Select paired device indication (blinking) option: I.
   - The LED's on the peer device toggle for a few seconds to enable the easy identification of the peer

The configured parameters should be reflected in the Terminal Menu as shown in Figure 3-9 below:

Figure 3-9. Parameter configuration for basic PER.



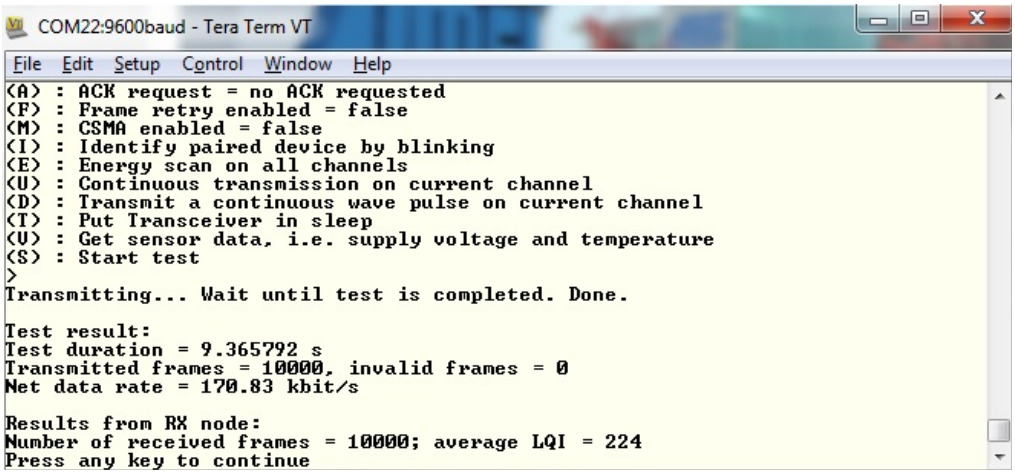
The second mobile receiving node should now be moved to a location where a regular packet error rate measurement should give more exact indication about the RF link quality between the stationary STB-RCB assembly and this second mobile node. Once this node is placed the test can be started.

1) Start test: S
   - This starts the PER test

During the test the text: "Transmitting … wait until test is completed." is displayed in the Terminal window while one LED on the receiver node is blinking fast.

After the test has been completed the test results are displayed in the Terminal window. The test result contains information like test duration, number of successfully transmitted frames, effective data rate, number of successfully received frames and average link quality for all data frames. An example test results is shown in Figure 3-10.

Figure 3-10. Packet Error Rate.



The PER test provides a statistical indication about the reliability of an RF link. For increased statistical significance it is recommended to use a higher number of packets (>10000) for this test and to repeat the test for one dedicated RF link between two points for more than one time.

4 Uploading Firmware Using On-chip Boot loader Program

The Atmel RF4CE-EK development kit devices are preprogrammed with a serial boot loader program that allows the update of the firmware in the RCB's via the serial interface of the Atmel ATmega128RFA1. The Atmel AVR® microcontroller has a separate boot loader program memory section where this serial boot loader firmware is located.

4.1 Functionality Overview

The boot loader firmware is capable of reprogramming (flashing) the device program memory with a new program application image. Due to that the application firmware can be easily upgraded without a separate programmer (for example, Atmel AVR JTAGICE mkII).

The RCB's must be mounted on top of the STB board which is connected with a USB cable to a host PC. A boot loader PC application is executed on the host. It establishes a serial connection link with the STB and is used to upload a new application memory binary file (.srec extension) into the device program application memory.

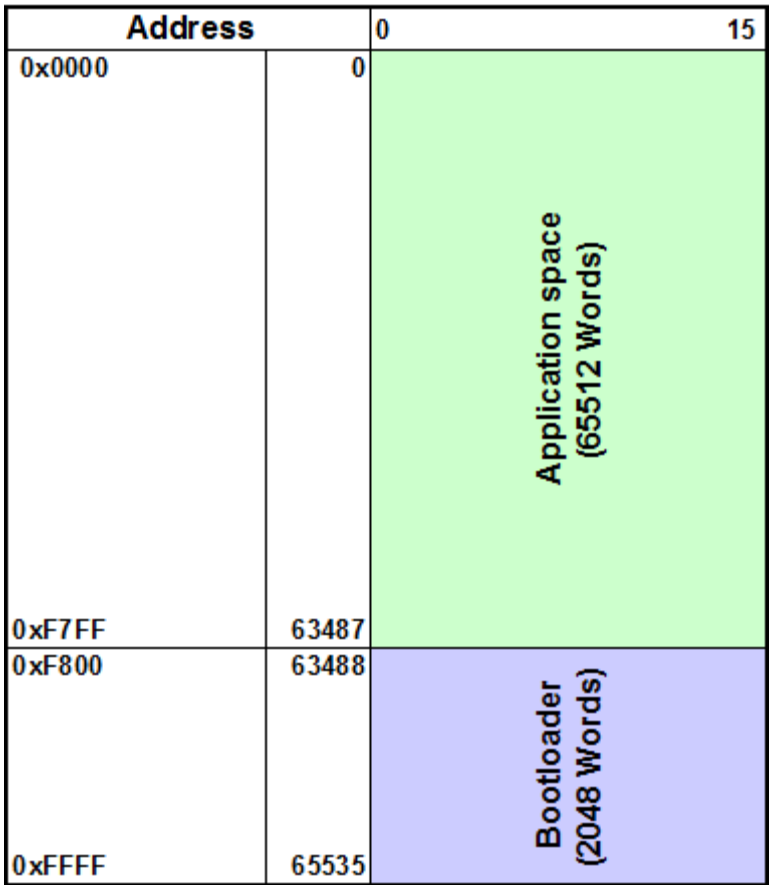
The boot loader firmware occupies 2048 words in the upper address space of the AVR microcontroller. The boot loader firmware is executed upon power up or device reset. The program execution starts in the boot loader memory section and initiates a handshaking procedure with the boot loader PC host application before it can begin with a new program upload.

ATmega128RFA1 program memory layout including its application and boot loader sections is shown in Figure 4-1.

After uploading a new device firmware into the application program memory the boot loader remains in its original program space location. If desired, it can be used again to update the existing application program anytime again.

If the PC boot loader application is not responding to this handshaking procedure the boot loader hands over the program execution after a timeout of ~500milisecond to the application that is actually loaded into the regular program memory.

Figure 4-1. Atmel ATmega128RFA1 Program Memory Partitioning.



4.2 Flashing RF4CE Image using the serial boot loader

The steps to use the boot loader firmware to update the software in the application space of the ATmega128RFA1 are as follows:

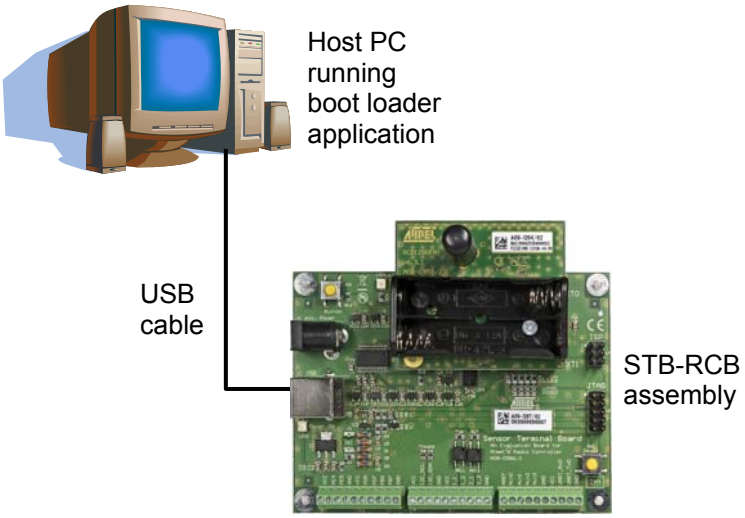
1. Install the boot loader GUI application on your host-PC.
The boot loader GUI application is found on the atmel.com website under the Atmel AVR2054 application note [11].

2. Connect one RCB that should get a new application firmware to the STB.
3. Connect the STB-RCB board assembly to the host PC using the USB cable.
If the STB board has never been connected to the host PC a USB software driver must be downloaded and installed for the STB [10]. The STB will get a COM port assigned by the operating system.

It is also possible to use UART pins of the ATmega128RFA1 which are accessible on the RCB Key Remote Control board to use this serial boot loader update method. For this purpose the kit contains a special serial cable that can be connected to a host PC's standard RS232 connector and connects on the other side to the exposed UART pins of the RCB128RFA1 which are accessible via connector EXT on the RCB Key Remote Control.

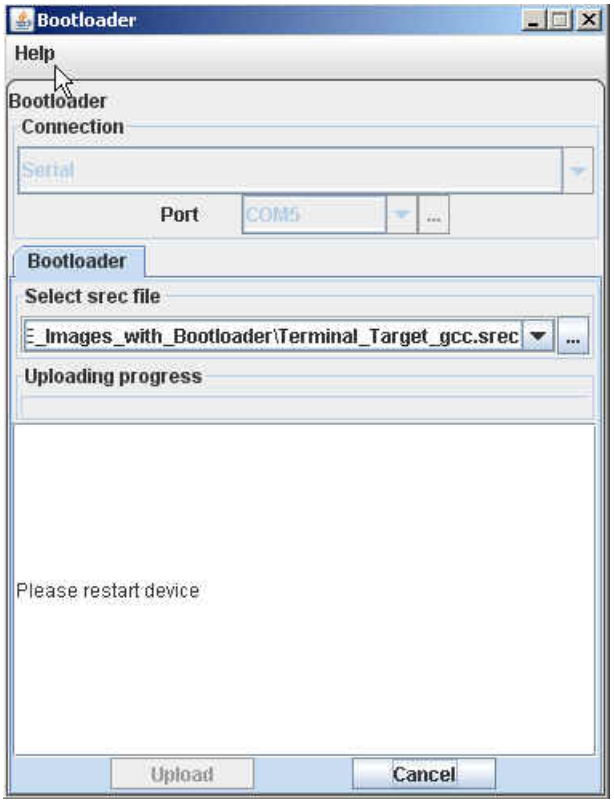
Throughout this document the boot loader operation based on the RCB-STB and the USB interface is the primary used firmware update method.

Figure 4-2. Application Software Upgrade with a Boot loader hardware setup.



4. Launch the serial boot loader PC software application on the host PC.
5. Select the source file to be uploaded (file_name.srec) using the PC boot loader application by either typing the source file path name or navigating to it using the navigation action button.
6. Once the source filename has been entered push the ‘Upload’ button to enter the source file name and begin uploading procedure.

Figure 4-3. Boot loader PC application GUI window.



7. When the message ‘Please restart device’ appears push the reset button on the STB. Now the boot loader firmware in the Atmel ATmega128RFA1 will initiate the handshake procedure and the new image upload begins. The progress indicator gives visual feedback about this upload procedure.
8. Upon upload completion the RCB is ready to execute the updated application.

Device fuses are not changed during the firmware upgrade using the serial boot loader.

9. When the firmware update for one RCB has been finished the board can be removed from the STB and the second RCB can be updated with a different application firmware.

For further details about the usage of the serial boot loader and the boot loader PC GUI refer to [11].

IMPORTANT

This application firmware update has to be done for the regular Atmel RF4CE operation of the kit. The exact steps to update one RCB to an RF4CE remote control device and the other RCB to its counterpart are explained in Section 5.2.1.

5 Running RF4CE example application

The primary usage scenario for the Atmel RF4CE evaluation kit allows the operation of a basic RF4CE network using a controller as well as a target device. The hardware provided in this kit is directly tailored to support this RF4CE network operation.

5.1 The RF4Control software package (AVR2102)

The Atmel certified RF4Control software stack provides example applications that demonstrate a reference implementation of RF4CE based remote controls and consumer electronic devices.

This RF4Control software package can be downloaded from the Atmel website:

http://www.atmel.com/rf4ce

The package contains the RF4CE network library, example applications as well as the RF4CE stack documentation. It includes example applications for the controller side and for the target side of a typical RF4CE network. The full description for the contents of this software package is found in [2].

5.2 Setting up a RF4CE demo

An RF4CE network consists of at least one target node and one controller node. These terms of the ZigBee RF4CE standard can be verified in [1].

A specific firmware has to be loaded into one RCB that should act as the controller in the RF4CE network and another firmware has to be loaded into the RCB that will act as the target side of the same network.

5.2.1 Updating the RCB firmware

The RCB's provided in this kit are preprogrammed with the Performance Test firmware as described in Chapter 3. In order to use the kit for RF4CE applications the firmware of both RCB has to be updated. The RCB128RFA1 can be either programmed via the serial boot loader process as described in Chapter 4 of this document or using a JTAG programmer as explained in Chapter 6.

The Atmel AVR2102 package contains example implementations for the target side and for the controller side for most of the provided applications. Step 4 in Section 5.2.2 needs to be repeated to update the firmware of both RCB that are supposed to be operated in the RF4CE network.

5.2.2 Updating the firmware of the controller RCB

1. Unpack the AVR2102 software package to a folder on the host-PC.
2. Connect one RCB with no batteries included to the STB and connect the STB to the host PC.

If required install the USB driver for the STB as described in Section 3.2.1.

3. Use the PC-based boot loader application for updating the RCB application firmware as described in Chapter 4.

4. Program the RCB with the Key Remote Controller example application from the AVR2102 software package:
   - **Applications**
     - **RF4CE_Examples**
       - **Key_Remote_Controller_Lib**
         - **ATMEGA128RFA1_RCB_6_3_KEY_RC**
           - **GCC**
             - **Key_Remote_Controller.srec**
5. Once the upload has been finished disconnect the RCB from the STB and insert the batteries again.
6. Plug the RCB into the RCB Key Remote Control board. The full controller side setup should look like shown in Figure 5-1.

Figure 5-1. Key remote board + RCB128RFA1.



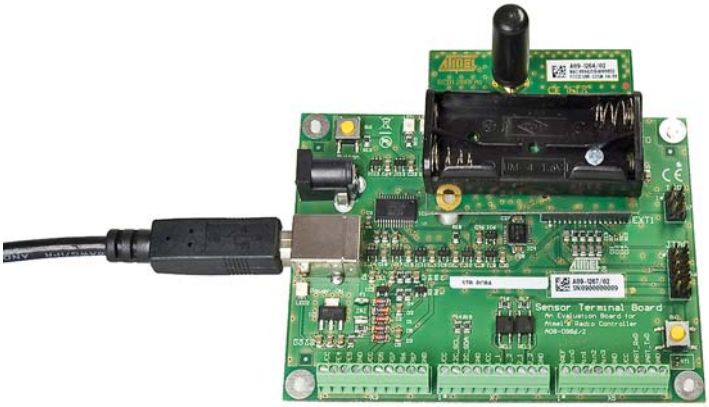
5.2.3 Updating the firmware of the target RCB

Repeat the steps 2, 3, and 4 from Section 5.2.2 with the second RCB but use instead the Key Remote Controller example application when updating the firmware for the RCB of the target side:

- **Applications**
  - **RF4CE_Examples**
    - **Terminal_Target_Lib**
      - **ATMEGA128RFA1_RCB_6_3_SENS_TERM_BOARD**
        - **GCC**
          - **Terminal_Target.srec**

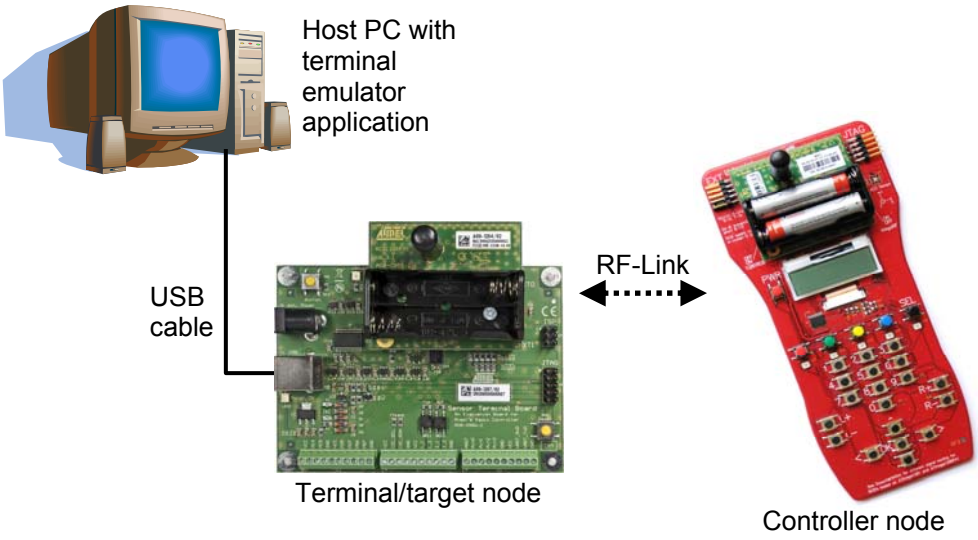
Leave the RCB-STB that has just been updated with the Terminal Target application connected to the host PC. The full target side setup should look like shown in Figure 5-2.

Figure 5-2. Sensor terminal board + RCB128RFA1.



The full setup of this basic Atmel RF4CE network where the target is controlled over the Terminal emulator application on the host PC is shown in Figure 5-3.

Figure 5-3. RF4CE network hardware setup.



5.3 Running the updated RF4CE demo

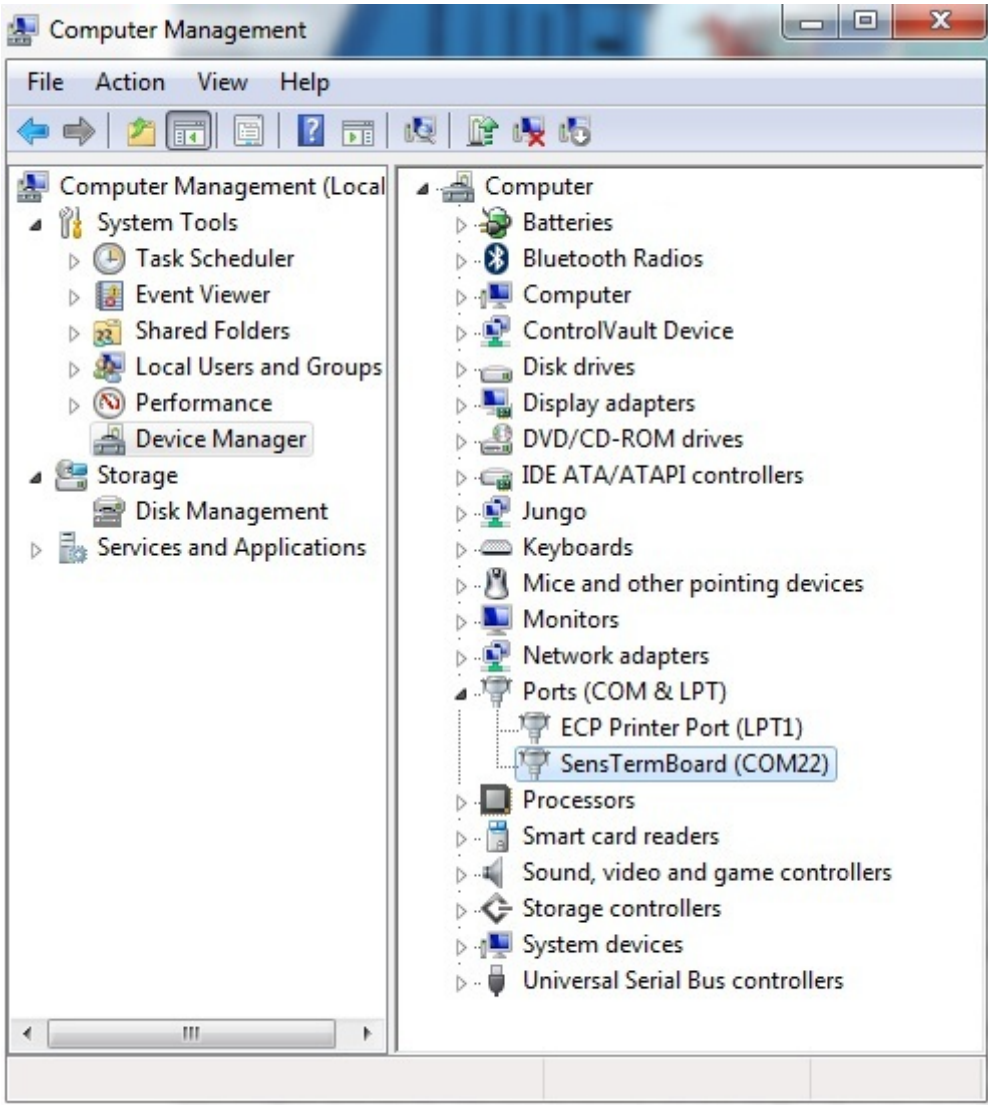
Once the RCB's have been updated with their respective RF4CE example application the following section describes the basic operation of this example network. The Atmel RF4Control software package contains also other application examples and details about the different applications can be found in [2].

The targets operation can be controlled by means of a standard Terminal emulator application. The basic installation of a Windows XP system already provides such tool with the HyperTerminal program, but any standard Terminal emulator software could be used as well.

To establish an RF4CE network now the Terminal Target application firmware that has been loaded to the RCB on the STB needs to be controlled via the Terminal emulator application.

When the STB has been connected to the host PC it has got a COM-Port number assigned by the operating system as shown in Figure 5-4.

Figure 5-4. Windows device manager.

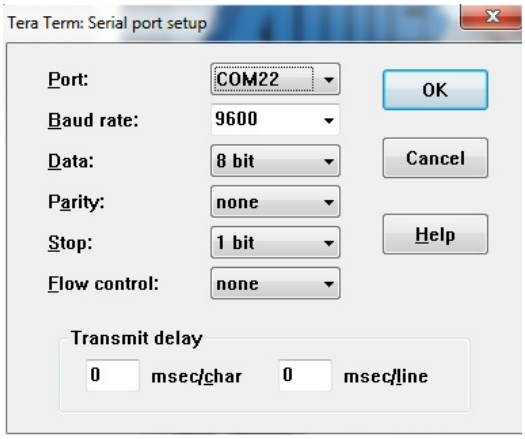


This assigned COM port needs to be opened in the Terminal emulator application.

1. Open this assigned COM port in a Terminal emulator, with the following settings:
   BAUD RATE: 9600
   PARITY: None
   DATA BITS: 8
   STOP BITS: 1
   FLOW CONTROL: Off

An example for this serial port configuration is shown in Figure 5-5.

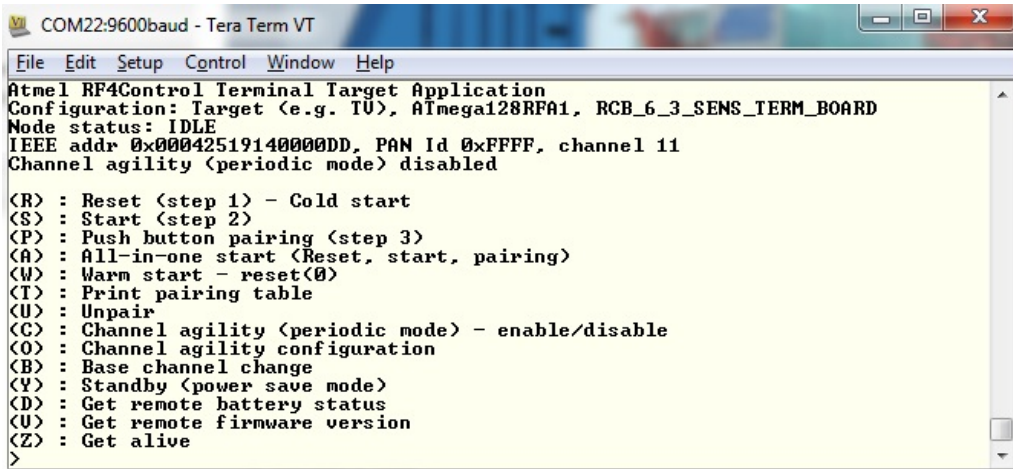
Figure 5-5. Serial port settings.



2. Once the connection to this COM-Port is established hit the <ENTER> key on the host PC's keyboard to get the Terminal Target application command interface displayed. See Figure 5-6.

The user interface of the Terminal Target application offers Atmel RF4CE protocol specific commands and control procedures to start and operate an RF4CE target node. The detailed operation of the Terminal Target application is explained in the Atmel AVR2102 RF4Control application note [7].

Figure 5-6. RF4Control Terminal Target application.



The Terminal Target application is now ready to be used. How to start an RF4CE network with this user interface is explained in Section 5.3.2.

5.3.1 The Key Remote Controller application

The Key Remote Controller application provides a full RF4CE compliant implementation of a controller node that uses the RCB Key Remote Control board from the RF4CE-EK to stimulate a RF4CE target. It can be used to understand and verify the role of the controller in an RF4CE network.

The RCB Key Remote Control board will be used to initiate RF4CE specific procedures like pairing and will be used for sending remote control commands to the target node of the RF4CE network.

5.3.2 Start the RF4CE network

In an Atmel RF4CE scenario the target node establishes the network by starting the network with a certain set of parameters. This can be done via the Terminal Target application.

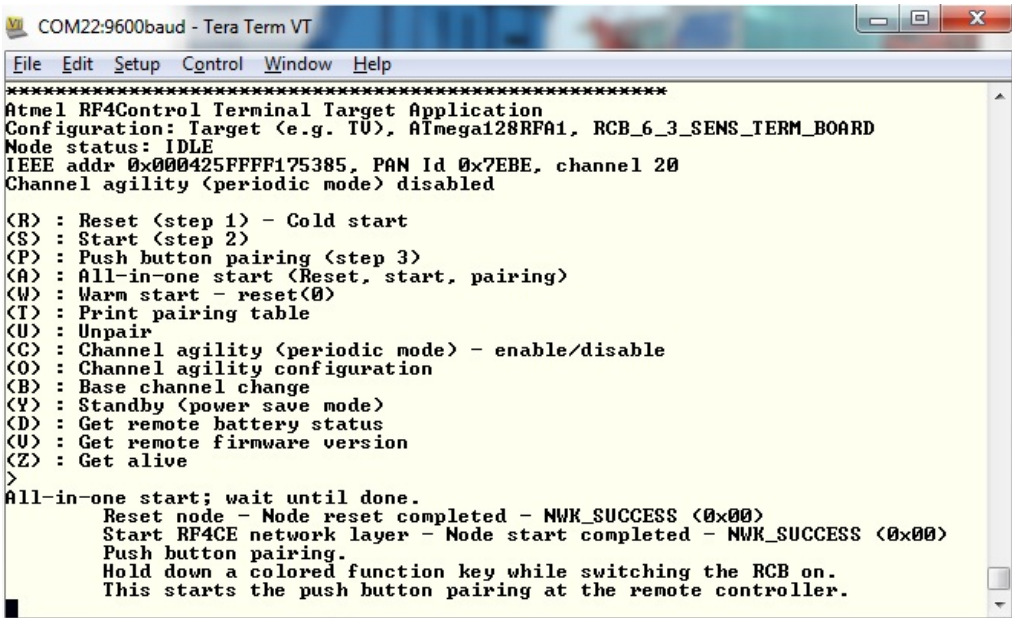
In the following example the target node is reset, the RF4CE network layer is started and the target is switched into push-button pairing mode [1]. Right after that the RCB Key Remote Control board is used to establish a secured link with the target by initiating the pairing sequence from the controller side. After a successful pairing the RCB Key Remote Control board can be used to send encrypted network messages like remote control key codes to the target and they will displayed in the Terminal application window.

The steps to run such example RF4CE network are as follows:

1. In the Atmel RF4Control Terminal Target application menu on the host-PC press: A.

This initiates the full sequence of target node reset, start of RF4CE network layer and initiating the push-button-pairing sequence.

Figure 5-7. RF4Control Terminal Target application push button pairing.

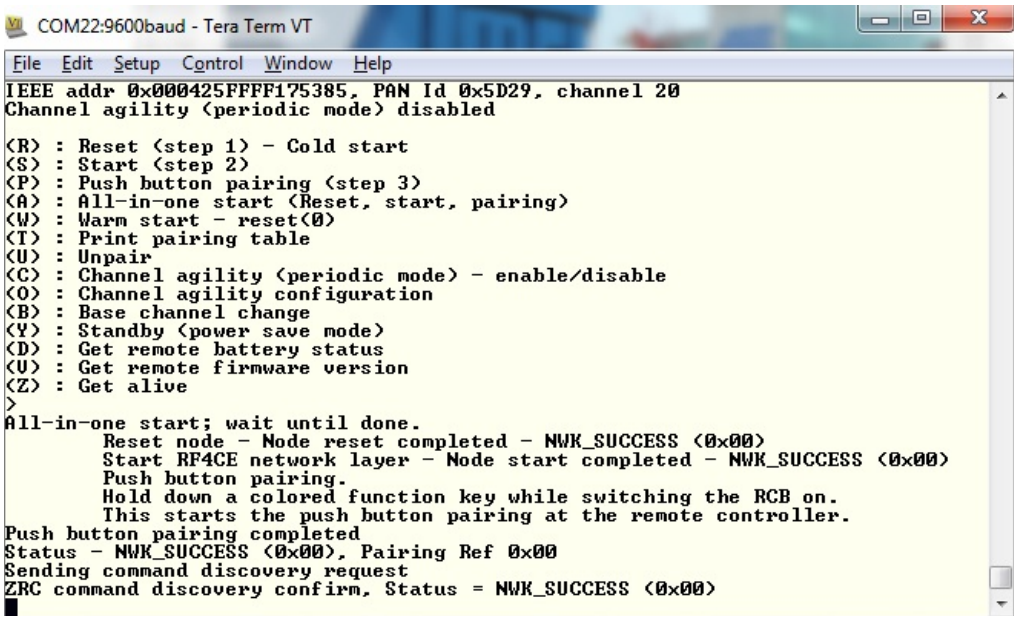


Now the push button pairing procedure has to be initiated also on the controller node in this RF4CE network. This step has to be performed before the pairing sequence on the Terminal Target application has timed out. The default timeout for this is 30 seconds.

2. Press & hold one of the four colored buttons (red, green, yellow, blue) on the RCB Key Remote Control board while the RCB is switched ON.

A successful pairing between the target in the controller is indicated in the Terminal emulator window with the status NWK_SUCCESS (0x00) as shown in Figure 5-8.

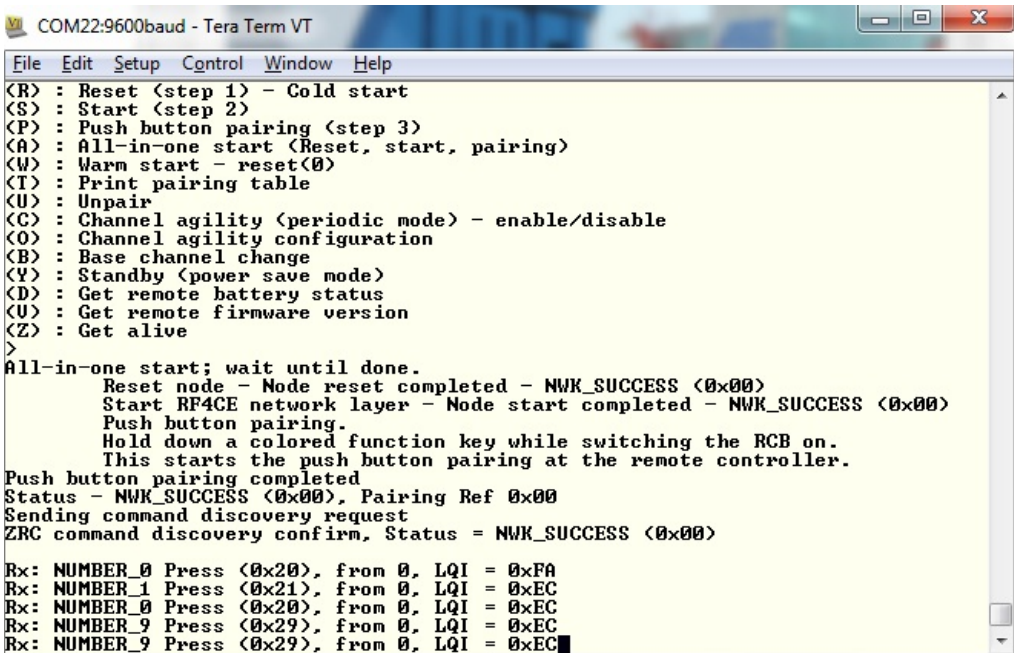
Figure 5-8. Atmel RF4Control Terminal Target application successful pairing.



Now the controller node has been successfully paired with the target node and each button press on the RCB Key Remote Control board is shown in the Terminal emulator window.

3. Press a button on the RCB Key Remote Control board and verify that every button press is displayed in the Terminal Target window.

Figure 5-9. RF4Control Terminal Target application button press.



From the Terminal Target application the user can now initiate several other actions that demonstrate the functions of the RF4CE protocol. This includes initiating channel

agility [1] or using the bi-directional RF link between the controller and the target to get information from the controller to the target (like requesting the battery status of the controller).

The complete command reference for the Terminal Target application is found in [7].

6 Programming RCB using Atmel AVR JTAGICE mkII

Atmel ATmega128RFA1 device program (Flash) and data (EE) memories as well as the fuse configuration can be programmed and updated using a JTAG interface programmer (for example, AVR JTAGICE mkII [9]). The RCB device programming is possible when the RCB is mounted on top of KEY_RC board to make a KEY_RC-RCB board assembly or when mounted on top of STB to make an STB-RCB board assembly.

6.1 Prerequisites for JTAG programming

In order to use the JTAG programming and debugging capabilities of the Atmel RF4CE-EK the following prerequisites have to be met:

1. A JTAG capable hardware like the KEY-RC-RCB or the STB-RCB setup.
2. A JTAG programmer (like the AVR JTAGICE mkII).
3. JTAG programming software (like the Atmel programming IDE AVR Studio®).

The Atmel AVR Studio package version 4 can be downloaded from the atmel.com website:

http://www.atmel.com/dyn/products/tools_card.asp?tool_id=2725

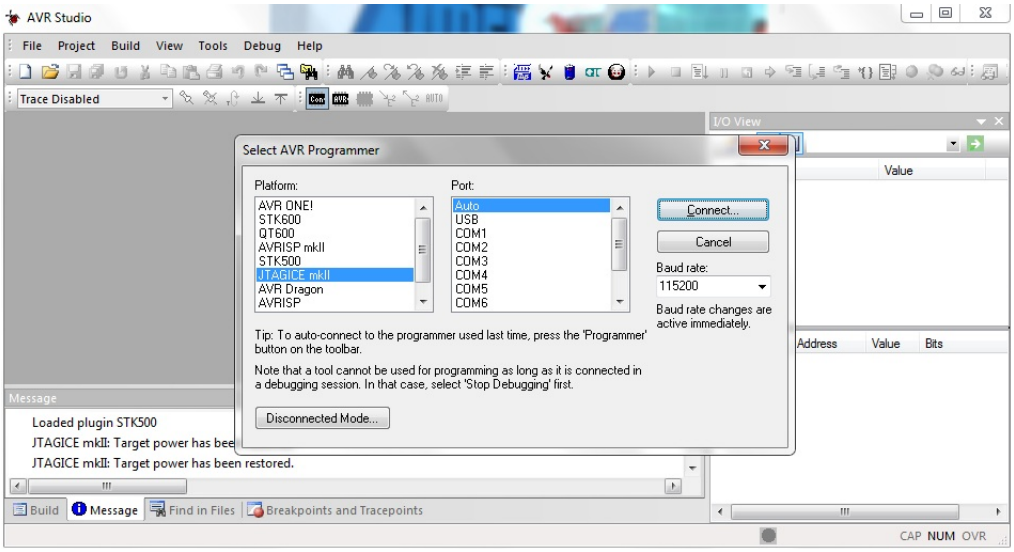
Once the AVR Studio is installed it allows the use of the AVR JTAGICE mkII to reprogram the RCB.

6.2 Programming the RCB in the KEY-RC setup

Follow the steps below:

1. Assemble KEY_RC-RCB (Controller) node.
2. Insert batteries into the controller nodes RCB128RFA1.
3. Connect the AVR JTAGICE mkII JTAG cable to the JTAG connector on the top of the RCB Key Remote Control board.
4. Switch the power supply on: (1) on the RCB and (2) on the AVR JTAGICE mkII programmer.
5. Launch AVR Studio IDE [8].
6. Click on the ‘Con’ button to display the ‘Connect’ dialog. Select AVR JTAGICE mkII as a platform and USB or Auto as a port name and click Connect. See Figure 6-2 as reference.

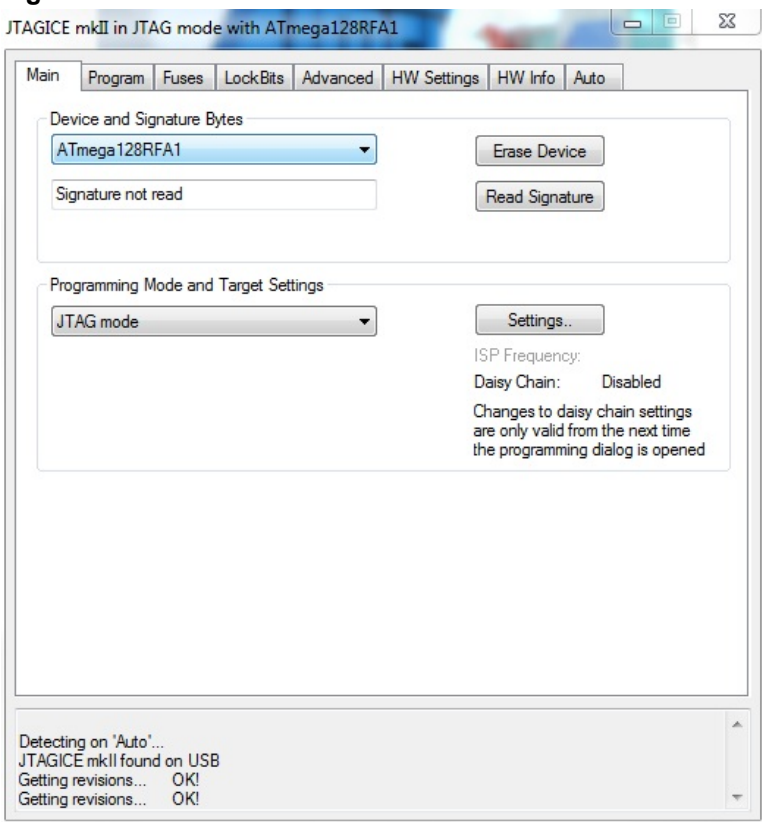
Figure 6-1. Atmel AVR Studio 4 Connect Dialog button.



Now the JTAG programming dialog appears.

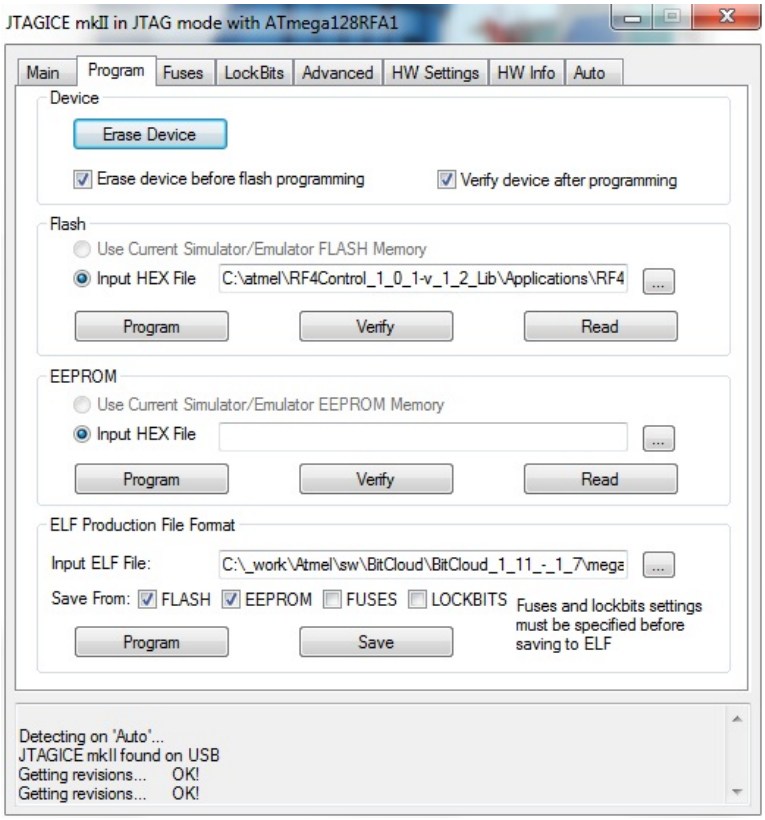
7. In the Main tab select Device: ATmega128RFA1, and Programming Mode and Target Settings: JTAG mode. See Figure 6-2.

Figure 6-2. AVR Studio 4 Device selection.



8. In the Program tab in the Flash section pick an Input .hex file and click Program. The device is programmed with a new application program image.

Figure 6-3. Atmel AVR Studio 4 JTAG programming dialog.



9. In the Fuses tab select the fuse configuration (see Appendix A) depending on the application.

6.3 Programming the RCB in the STB setup

Follow steps below:

1. Assemble STB-RCB (Terminal Target).
2. Connect USB cable to a free USB port of a PC or any other USB equipped host. This supplies power to the node (STB-RCB board assembly).
3. Connect the Atmel AVR JTAGICE mkII programmer cable to the JTAG connector on the STB.
4. Launch AVR Studio IDE.
5. Click on the ‘Con’ connect action button to display the ‘Connect’ dialog. Select AVR JTAGICE mkII as a platform and USB as a port name and click OK.
6. In the Main tab select Device: ATmega128RFA1, and Programming Mode and Target Settings: JTAG mode.
7. In the Fuses tab select the fuse configuration (see Appendix A) depending on application.
8. In the Program tab in the Program section provide .hex file path name and click program. The device is programmed with a new application program image. To program the Atmel RF4Control target node image please see the Atmel AVR2102 application note for a source file name (target_source_filename.hex).

6.4 Programming Performance Test application

The Performance Test Measurement application firmware is programmed into the RCB in the similar manner as it is done for RF4Control stack for the controller and target applications.

The performance application source file 'Performance_test_evk_with_BT.hex' includes the performance firmware and the boot loader module which was originally provided with the Atmel RF4CE-EK kit [14].

7 References

[1] ZigBee RF4CE Specification Version 1.00
http://www.zigbee.org
094945r00ZB_RF4CE-Specification.pdf

[2] Atmel RF4Control – ZigBee RF4CE Certified Platform
http://www.atmel.com/dyn/products/tools_card_mcu.asp?tool_id=4712

[3] IEEE Standard 802.15.4TM-2006: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs)
http://standards.ieee.org/getieee802/download/802.15.4-2006.pdf

[4] Atmel Atmega128RFA1; Microcontroller with Low Power 2.4GHz Transceiver for ZigBee™ and IEEE 802.15.4™
http://www.atmel.com/dyn/products/product_card.asp?part_id=4692

[5] AVR2025: IEEE 802.15.4 MAC Software Package for AVR Z-Link
http://www.atmel.com/dyn/products/tools_card.asp?tool_id=4675&category_id=163&family_id=676&subfamily_id=2124

[6] AVR2037: RCB Key Remote Control - Hardware User Manual
http://www.atmel.com/dyn/resources/prod_documents/doc8356.pdf

[7] AVR2102: RF4Control – User Guide
http://www.atmel.com/dyn/products/tools_card.asp?tool_id=4712

[8] Atmel AVR Studio 4, IDE for writing and debugging AVR applications
http://www.atmel.com/dyn/products/tools_card.asp?tool_id=2725

[9] Atmel AVR JTAGICE mkII
http://www.atmel.com/dyn/products/tools_card.asp?tool_id=3353

[10] Sensor Terminal Board, Dresden Elektronik GmbH
http://www.dresden-elektronik.de/shop/prod75.html

[11] AVR2054: Serial Boot loader User Manual
http://www.atmel.com/dyn/products/tools_card.asp?tool_id=4495&category_id=163&family_id=676&subfamily_id=2124

[12] AVR2044: RCB128RFA1 – Hardware User Manual
http://www.atmel.com/dyn/resources/prod_documents/doc8339.pdf

[13] Minimalist GNU for Windows
http://www.mingw.org or http://sourceforge.net/projects/mingw/

[14] RF4CE Remote Control Evaluation Kit documentation
http://www.atmel.com/dyn/products/tools_card.asp?tools_id=4835

Appendix A - Performance Test Measurement Application

A.1 ATmega128RFA1 fuses – no boot loader configuration

Performance Application Atmel ATmegaRFA1 Fuse Settings:

Fuse	Description
BODLEVEL	Brown-out detection at VCC=1.8V
OCDEN	en
JTAGEN	en
SPIEN	en
WDTON	dis
EESAVE	en
BOOTSZ	Boot flash=4096
BOOTRST	dis
CKDIV8	dis
CKOUT	dis
SUT_CKSEL	Int. RC Osc. 6CK + 65ms

FUSES OVERVIEW:

EXTENDED	0xFE
HIGH	0x11
LOW	0xE2

A.2 ATmega128RFA1 fuses –boot loader configuration

Performance Application with Boot loader ATmegaRFA1 Fuse Settings:

Fuse	Description
BODLEVEL	Brown-out detection at VCC=1.8V
OCDEN	en
JTAGEN	en
SPIEN	en
WDTON	dis
EESAVE	en
BOOTSZ	Boot flash=2048
BOOTRST	en
CKDIV8	en
CKOUT	dis
SUT_CKSEL	Int. RC Osc. 6CK + 65ms

FUSES OVERVIEW (with boot loader):

EXTENDED	0xFE
HIGH	0x92
LOW	0x42

A.3 List of user interface commands & options

Once the devices have finished the initial peer configuration phase the user interface on the transmitter side allows performing several tests. These can be configured and initiated by pressing the corresponding command key in the Terminal program.

NOTE

Commands are listed in order of the appearance in the user interface and not in alphabetical order.

Table A-1. User interface commands.

Command key	Parameter & possible options
C	Channel [11 … 26]
P	Channel page: 0 – 250 kbps, OQPSK Modulation (IEEE 802.15.4) 2 – 500 kbps, OQPSK Modulation (proprietary) 16 – 1000 kbps, OQPSK Modulation (proprietary) 17 – 2000 kbps, OQPSK Modulation (proprietary)
W	Transmit Power [-17dBm … +3dBm]
N	Number of test frames [1 … 2^32]
L	Frame length (PSDU) [11 … 127]
A	Enable/Disable ACK request
F	Enable/Disable Frame retry
M	Enable/Disable CSMA
I	Identify peer device by LED toggle
E	Energy scan (duration [0 … 14])
U	Continuous transmission on current channel, Mode: C – Continuous wave on channel (-0.5MHz) P – PRBS transmission on channel
D	Transmit continuous wave pulse on channel (-0.5MHz)
T	Put device into sleep mode Note: Once the device is in sleep mode the user interface only accepts the ‘T’ command to wake up the device
V	Get sensor data (voltage, temperature)
S	Start test Note: The transmitter starts to send the number of configured packets (N-command), with the configured length (L-command) on the configured channel (C-command) with the configured transmit power (W-command)

EVALUATION BOARD/KIT IMPORTANT NOTICE

This evaluation board/kit is intended for use for **FURTHER ENGINEERING, DEVELOPMENT, DEMONSTRATION, OR EVALUATION PURPOSES ONLY**. It is not a finished product and may not (yet) comply with some or any technical or legal requirements that are applicable to finished products, including, without limitation, directives regarding electromagnetic compatibility, recycling (WEEE), FCC, CE or UL (except as may be otherwise noted on the board/kit). Atmel supplied this board/kit "AS IS," without any warranties, with all faults, at the buyer's and further users' sole risk. The user assumes all responsibility and liability for proper and safe handling of the goods. Further, the user indemnifies Atmel from all claims arising from the handling or use of the goods. Due to the open construction of the product, it is the user's responsibility to take any and all appropriate precautions with regard to electrostatic discharge and any other technical or legal concerns.

EXCEPT TO THE EXTENT OF THE INDEMNITY SET FORTH ABOVE, NEITHER USER NOR ATMEL SHALL BE LIABLE TO EACH OTHER FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES.

No license is granted under any patent right or other intellectual property right of Atmel covering or relating to any machine, process, or combination in which such Atmel products or services might be or are used.

Mailing Address: Atmel Corporation, 2325 Orchard Parkway, San Jose, CA 95131
Copyright © 2009, Atmel Corporation.

8 Table of contents

Features *1*
1 Introduction *1*
2 Hardware description *3*
2.1 Kit contents 3
2.1.1 The RCB128RFA1 3
2.1.2 The RCB Key Remote Control board 3
2.1.3 The Sensor Terminal board 3
3 Performance Test Measurement Firmware Package *4*
3.1 Simple Range Application 4
3.1.1 Application Setup 4
3.1.2 Operational description 4
3.1.3 Purpose of the Simple Range Measurement Test 5
3.2 Packet Error Rate Measurement (PER) 6
3.2.1 Setting and Configuring Kit Hardware 6
3.2.2 Packet Error Rate Measurement Menu 9
3.2.3 Configuration parameters 10
3.2.4 Single side tests 10
3.2.5 Start Test (Option S) 10
3.2.6 A typical PER test sequence 10
4 Uploading Firmware Using On-chip Boot loader Program *13*
4.1 Functionality Overview 13
4.2 Flashing RF4CE Image using the serial boot loader 14
5 Running RF4CE example application *17*
5.1 The RF4Control software package (AVR2102) 17
5.2 Setting up a RF4CE demo 17
5.2.1 Updating the RCB firmware 17
5.2.2 Updating the firmware of the controller RCB 17
5.2.3 Updating the firmware of the target RCB 18
5.3 Running the updated RF4CE demo 19
5.3.1 The Key Remote Controller application 21
5.3.2 Start the RF4CE network 22
6 Programming RCB using Atmel AVR JTAGICE mkII *25*
6.1 Prerequisites for JTAG programming 25
6.2 Programming the RCB in the KEY-RC setup 25
6.3 Programming the RCB in the STB setup 27
6.4 Programming Performance Test application 28
7 References *29*
Appendix A - Performance Test Measurement Application *30*
A.1 ATmega128RFA1 fuses – no boot loader configuration 30
A.2 ATmega128RFA1 fuses –boot loader configuration 30

A.3 List of user interface commands & options 31
EVALUATION BOARD/KIT IMPORTANT NOTICE 32
8 Table of contents 33



Atmel Corporation
2325 Orchard Parkway
San Jose, CA 95131
USA
Tel: (+1)(408) 441-0311
Fax: (+1)(408) 487-2600
www.atmel.com

Atmel Asia Limited
Unit 01-5 & 16, 19F
BEA Tower, Milennium City 5
418 Kwun Tong Road
Kwun Tong, Kowloon
HONG KONG
Tel: (+852) 2245-6100
Fax: (+852) 2722-1369

Atmel Munich GmbH
Business Campus
Parkring 4
D-85748 Garching b. Munich
GERMANY
Tel: (+49) 89-31970-0
Fax: (+49) 89-3194621

Atmel Japan
9F, Tonetsu Shinkawa Bldg.
1-24-8 Shinkawa
Chou-ku, Tokyo 104-0033
JAPAN
Tel: (+81) 3523-3551
Fax: (+81) 3523-7581

© 2011 Atmel Corporation. All rights reserved.

Atmel®, Atmel logo and combinations thereof, AVR®, AVR Studio®, and others 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. Other terms and product names may be trademarks of others.

Disclaimer: The information in this document is provided in connection with Atmel products. No license, express or implied, by estoppel or otherwise, to any intellectual property right is granted by this document or in connection with the sale of Atmel products. **EXCEPT AS SET FORTH IN THE ATMEL TERMS AND CONDITIONS OF SALES LOCATED ON THE ATMEL WEBSITE, ATMEL ASSUMES NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO ITS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT SHALL ATMEL BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, SPECIAL OR INCIDENTAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS AND PROFITS, BUSINESS INTERRUPTION, OR LOSS OF INFORMATION) ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF ATMEL HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.** Atmel makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and product descriptions at any time without notice. Atmel does not make any commitment to update the information contained herein. Unless specifically provided otherwise, Atmel products are not suitable for, and shall not be used in, automotive applications. Atmel products are not intended, authorized, or warranted for use as components in applications intended to support or sustain life.

8391A-AVR-08/11