RFbee V1.1 - Wireless Arduino compatible node

|  |
| --- |
| **Contents**   [[hide](javascript:toggleToc())]   * [1 Introduction](http://www.seeedstudio.com/wiki/index.php?title=RFbee_V1.1_-_Wireless_Arduino_compatible_node#Introduction) * [2 Features](http://www.seeedstudio.com/wiki/index.php?title=RFbee_V1.1_-_Wireless_Arduino_compatible_node#Features) * [3 Application Ideas](http://www.seeedstudio.com/wiki/index.php?title=RFbee_V1.1_-_Wireless_Arduino_compatible_node#Application_Ideas) * [4 Specification](http://www.seeedstudio.com/wiki/index.php?title=RFbee_V1.1_-_Wireless_Arduino_compatible_node#Specification)   + [4.1 Electrical Characteristics](http://www.seeedstudio.com/wiki/index.php?title=RFbee_V1.1_-_Wireless_Arduino_compatible_node#Electrical_Characteristics) * [5 Pin definition and Rating](http://www.seeedstudio.com/wiki/index.php?title=RFbee_V1.1_-_Wireless_Arduino_compatible_node#Pin_definition_and_Rating) * [6 Usage](http://www.seeedstudio.com/wiki/index.php?title=RFbee_V1.1_-_Wireless_Arduino_compatible_node#Usage)   + [6.1 Hardware Installation](http://www.seeedstudio.com/wiki/index.php?title=RFbee_V1.1_-_Wireless_Arduino_compatible_node#Hardware_Installation)     - [6.1.1 Figure 1:USB using UartSB](http://www.seeedstudio.com/wiki/index.php?title=RFbee_V1.1_-_Wireless_Arduino_compatible_node#Figure_1:USB_using_UartSB)     - [6.1.2 Figure 2:Seeduino via XbeeShield](http://www.seeedstudio.com/wiki/index.php?title=RFbee_V1.1_-_Wireless_Arduino_compatible_node#Figure_2:Seeduino_via_XbeeShield)     - [6.1.3 Any device using a Uart](http://www.seeedstudio.com/wiki/index.php?title=RFbee_V1.1_-_Wireless_Arduino_compatible_node#Any_device_using_a_Uart)   + [6.2 Example](http://www.seeedstudio.com/wiki/index.php?title=RFbee_V1.1_-_Wireless_Arduino_compatible_node#Example)     - [6.2.1 RFBee monitored by RF Explorer](http://www.seeedstudio.com/wiki/index.php?title=RFbee_V1.1_-_Wireless_Arduino_compatible_node#RFBee_monitored_by_RF_Explorer)       * [6.2.1.1 Requirements](http://www.seeedstudio.com/wiki/index.php?title=RFbee_V1.1_-_Wireless_Arduino_compatible_node#Requirements)       * [6.2.1.2 Set RFBee to work](http://www.seeedstudio.com/wiki/index.php?title=RFbee_V1.1_-_Wireless_Arduino_compatible_node#Set_RFBee_to_work) * [7 Support](http://www.seeedstudio.com/wiki/index.php?title=RFbee_V1.1_-_Wireless_Arduino_compatible_node#Support)   + [7.1 How to update firmware](http://www.seeedstudio.com/wiki/index.php?title=RFbee_V1.1_-_Wireless_Arduino_compatible_node#How_to_update_firmware) * [8 Version Tracker](http://www.seeedstudio.com/wiki/index.php?title=RFbee_V1.1_-_Wireless_Arduino_compatible_node#Version_Tracker) * [9 Resources](http://www.seeedstudio.com/wiki/index.php?title=RFbee_V1.1_-_Wireless_Arduino_compatible_node#Resources) * [10 How to buy](http://www.seeedstudio.com/wiki/index.php?title=RFbee_V1.1_-_Wireless_Arduino_compatible_node#How_to_buy) * [11 See Also](http://www.seeedstudio.com/wiki/index.php?title=RFbee_V1.1_-_Wireless_Arduino_compatible_node#See_Also) * [12 Licensing](http://www.seeedstudio.com/wiki/index.php?title=RFbee_V1.1_-_Wireless_Arduino_compatible_node#Licensing) * [13 External Links](http://www.seeedstudio.com/wiki/index.php?title=RFbee_V1.1_-_Wireless_Arduino_compatible_node#External_Links) |

Introduction

The RFBee is a RF module providing easy and flexible wireless data transmission between devices. It is based on a AVR Atmega168 working as a fully functional [Arduino](http://www.seeedstudio.com/wiki/Arduino) connected via SPI to a TI CC1101 RF transceiver.

**Model:**[**WLS126E1P**](http://www.seeedstudio.com/depot/rfbee-v11-wireless-arduino-compatible-node-p-614.html?cPath=139_140)



Features

* Range: Indoor/Urban: up to 50m; Outdoor line-of-sight: up to 120m;
* Receiver Sensitivity: -95dBm
* RF Data Transmission Rate: 4,800bps; 76,800bps
* Working Frequency : 868MHz & 915MHz
* Communication type: Point to Point, or Point to Multipoint.
* Easy-to-Use Serial Interface and rich extendable ports
* Easy-to-Use AT Command: Set working mode, Serial Baud Rate, etc.
* Open source Hardware and Firmware
* Socket compatible with the Xbee, so you can plug it into any Xbee socket as a quick replacement.

**Note:**

Only the Rx,Tx, VCC, GND pins are identical to the Xbee. RFbee’s will not communicate with Xbee’s, so RFbee’s need to be used on both ends of the wireless connection.

Application Ideas

* Powerful RF control
* Easy to implement of WSN (Wireless Sensor Network)

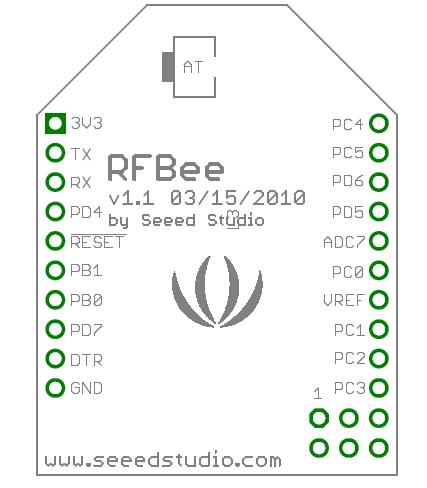
Specification

|  |  |
| --- | --- |
| **Specification** | **Value** |
| Microprocessor | Atmega168 |
| PCB size | 24.38mmx32.94mmx0.8mm |
| Indicators | No |
| Power supply | 3.3V |
| IO counts | 9 |
| ADC input | 7(6 multiplexing with IO) |
| Program interface | USB |
| Connectivity | Socket compatible with XBee |
| Communication Protocol | Uart(TTL) |
| Operating Frequency Band | ISM 868MHz & 915MHz |
| Outline Dimension | 24.38mmx32.94mmx15mm |

**Electrical Characteristics**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Specification** | **Min** | **Typ** | **Max** | **Unit** |
| Input voltage | 3.0 | 3.3 | 3.6 | VDC |
| Transmit Current |  | 34.5 |  | mA |
| Receive Current |  | 18.1 |  | mA |
| Idle Current |  | 5.2 |  | mA |
| Power-down Current |  | <0.3 |  | mA |
| Operating Temperature | -50 |  | 125 | °C |

Pin definition and Rating

[](http://www.seeedstudio.com/wiki/File:RFBee-pin.jpg)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Pin** | **#** | **Pad Type** | **Description** | **Arduino Pin Number** |
| 3V3 | 1 | Supply input | VCC, +3.3V | - |
| TX | 2 | Output | Uart Tx port | 1(DIO) |
| RX | 3 | Input | Uart Rx port | 0(DIO) |
| PD4 | 4 | Input/Output | ATmega168 PD4 | 4(DIO) |
| !RESET | 5 | Input | ATmega168 Reset port |  |
| PB1 | 6 | Input/Output | ATmega168 PB1 | 9(DIO) |
| PB0 | 7 | Input/Output | ATmega168 PB0 | 8(DIO) |
| PD7 | 8 | Input/Output | ATmega168 PD7 | 7(DIO) |
| !DTR | 9 | Input | Used for programming the ATmega168 | - |
| GND | 10 | GND | GND | - |
| PC3 | 11 | Input/Output | ATmega168 PC3 | 3(Analog input)/17(DIO) |
| PC2 | 12 | Input/Output | ATmega168 PC2 | 2(Analog input)/16(DIO) |
| PC1 | 13 | Input/Output | ATmega168 PC1 | 1(Analog input)/15(DIO) |
| VREF | 14 | Input | ATmega168 AREF port | - |
| PC0 | 15 | Input/Output | ATmega168 PC0 | 0(Analog input)/14(DIO) |
| ADC7 | 16 | Input | ATmega168 ADC7 | 7(Analog input) |
| PD5 | 17 | Input/Output | ATmega168 PD5 | 5(DIO) |
| PD6 | 18 | Input/Output | ATmega168 PD6 | 6(DIO) |
| PC5 | 19 | Input/Output | ATmega168 PC5 | 5(Analog input)/19(DIO) |
| PC4 | 20 | Input/Output | ATmega168 PC4 | 4(Analog input)/18(DIO) |

Usage

**Hardware Installation**

The RFBee can be connected in various ways, e.g.:

* to a PC via USB using a UartSB device.
* to a Seeeduino (or Arduino) via an XbeeShield
* to any other device having a Uart port

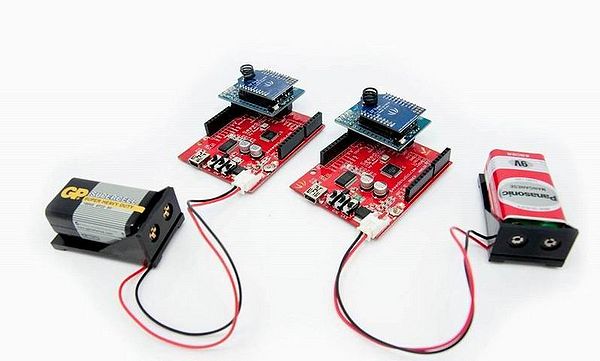
Note: the UartSB device and the XbeeShield are sold separately.

* UartSB: <http://www.seeedstudio.com/depot/uartsb-v22-a-much-more-powerful-usb-to-serial-converter-p-495.html>
* XbeeShield: <http://www.seeedstudio.com/depot/xbee%C3%82%C2%AE-shield-v11-by-seeedstudio-p-419.html>

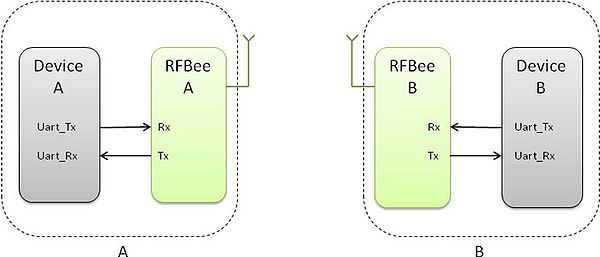
**Figure 1:USB using UartSB**

[](http://www.seeedstudio.com/wiki/File:RFBee-figure1.jpg)

**Figure 2:Seeduino via XbeeShield**

[](http://www.seeedstudio.com/wiki/File:RFBee-figure2.jpg)

**Any device using a Uart**

[](http://www.seeedstudio.com/wiki/File:RFBee-figure3.jpg)

**Example**

Here is a tutorial on how to use a RFBee transceiver from www.SeeedStudio.com with a RF Explorer Spectrum Analyzer from www.rf-explorer.com.

**RFBee monitored by RF Explorer**

The RFBee is a nice device which accepts some simple AT ASCII command strings to do some basic configuration. An ideal tool to experiment with digital RF transmission.

This example code is in the public domain and available [here](http://micro.arocholl.com/download/RFBeeTutorial/Test_RFBee.pde) and tested in Arduino IDE v0022.

|  |  |  |
| --- | --- | --- |
| [RFBee-Exam1.jpg](http://www.seeedstudio.com/wiki/File:RFBee-Exam1.jpg) | [RFBee-Exam2.jpg](http://www.seeedstudio.com/wiki/File:RFBee-Exam2.jpg) | [RFBee-Exam3.jpg](http://www.seeedstudio.com/wiki/File:RFBee-Exam3.jpg) |

**Requirements**

You may use a Seeeduino Stalker v02b to host the RFBee. Just plug the RFBee into the XBee slot. You also need a way to connect your Stalker to your PC to upload the sketch, I personally used a CP2102 USB bridge, you can use other USB bridges including the UartSBee recommended by Seeed.

Alternatively, you can do the same with an Arduino compatible board but will have to connect the RFBee with RX/TX of the CPU using a XBee 2mm connector, which may not be easy to get at your local shop.

Finally, you will need two jumper cables to connect Stalker port 2 to GND and port 3 and GND. We will use them as simple switches to configure the RFBee in different ways. Feel free to check the trivial sketch code to better understand how it works, and make your own changes.

Some familiarity with RFExplorer and RFBee user manual is required, otherwise some steps may not be obvious to you.

**Set RFBee to work**

After you upload the script in the Stalker, power off the unit completely so both ATMegas (Stalker's and RFBee's) resets at the same time and you start from a clean state.

Power the unit back, it will start transmission after Stalker led blinks 6 times.

RFExplorer will show received power and frequency. Play with antenna orientation till you get the best power response. In this tutorial we will use RFBee in 915Mhz, but equivalent results will be obtained in the 868Mhz. Take a look in the sketch code and uncomment the couple of lines coded for 868Mhz if your RF Explorer is in that band.

**More details, please visit** [micro.arocholl.com](http://micro.arocholl.com/index.php?option=com_content&view=article&id=53:tutorial-how-to-use-rf-explorer-to-monitor-a-rfbee&catid=40:article&Itemid=61).

Support

**How to update firmware**

You can update the RFbee firmware using the Arduino IDE using the procedure below. This procedure assumes the use of UartSB as this is the easiest way to connect a RFBee to a PC, see the section on Hardware installation for different ways to connect.

1. Connect your RFBee to UartSB, move switches to XBee and 3.3v, then connect it to your computer through a USB cable.
2. Download the source code of the RFBee firmware into your Arduino sketch folder
3. Open the Arduino IDE and open the RFBee\_vx\_x project. Then select Tools->Board->Arduino Pro or Pro Mini (3.3v, 8MHz) w/ATmega168. Choose the correct serial port from the Tools menu. You can now upload your RFBee firmware.
4. Reapply the configuration changes in the RFBee if they got lost during the update.
5. You could add or modify the firmware upon your demands, as the RFBee is able to work standalone as an Arduino.

>>RFBee firmware: <http://code.google.com/p/rfbee/downloads/list>

If you have other questions or other better design ideas, you can go to our [forum](http://www.seeedstudio.com/forum) or [wish](http://wish.seeedstudio.com/) to discuss.

Version Tracker

|  |  |  |
| --- | --- | --- |
| **Revision** | **Descriptions** | **Release** |
| v1.0 | Initial release | Mar 05, 2010 |
| v1.1 | revise release | Aug 27, 2010 |

Resources

* [RFBee User Manual](http://www.seeedstudio.com/depot/datasheet/RFBee%20User%20Manual%20v1.1.pdf)
* [RFbee firmware for Adruino 1.0](http://www.seeedstudio.com/wiki/File:RFbee_for_arduino1.0.zip)
* [RFbee firmware for Arduino 1.0(new)](http://www.seeedstudio.com/wiki/File:RFBee20121224.zip)
* [Schematic and layout in EAGLE file](http://www.seeedstudio.com/depot/datasheet/RFBee%20hardware%20v1.1.zip)
* [Forum](http://www.seeedstudio.com/forum/viewtopic.php?f=10&t=682&sid=7a9b1bed4f9fd10a9b1003ca1e48e756)