Radiocrafts
Embedded Wireless Solutions

RC2400-ZNM/RC2400HP-ZNM

ZigBee® PRO Network Module - User Manual

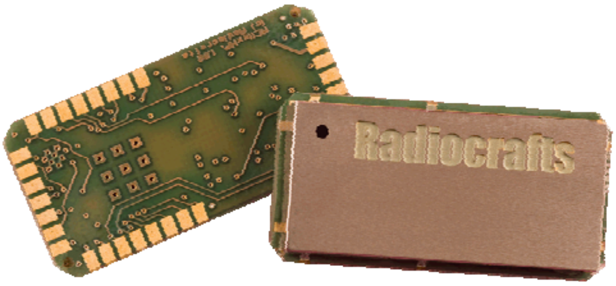


Table of contents

TABLE OF CONTENTS 1
INTRODUCTION 2
QUICK PRODUCT INTRODUCTION 2
ABOUT THIS DOCUMENT 2
PIN ASSIGNMENT 3
PIN DESCRIPTION 3
PIN CONFIGURATION 4
SERIAL COMMUNICATION 4
SPI INTERFACE 4
UART INTERFACE 4
GENERAL FRAME FORMAT 5
API COMMAND SET 6
STATES OF OPERATION 7
CONFIGURATION 7
OPERATION 9
PROGRAMMING INTERFACE 10
DOCUMENT REVISION HISTORY 11
DISCLAIMER 11
TRADEMARKS 11
LIFE SUPPORT POLICY 11
CONTACT INFORMATION 11

Introduction

This document includes or refers to all the needed information to develop solution with the RC2400-ZNP and RC2400HP-ZNM modules.

Quick Product Introduction

The ZNM series of modules are specially designed to meet the IEEE 802.15.4 standard and ZigBee PRO specification. It is preloaded with a ZigBee PRO compliant stack and offers an easy to use API via UART or SPI to an external processor. The external application processor can be of any type or brand and the development can be used with the tool and platform most convenient to the developer.

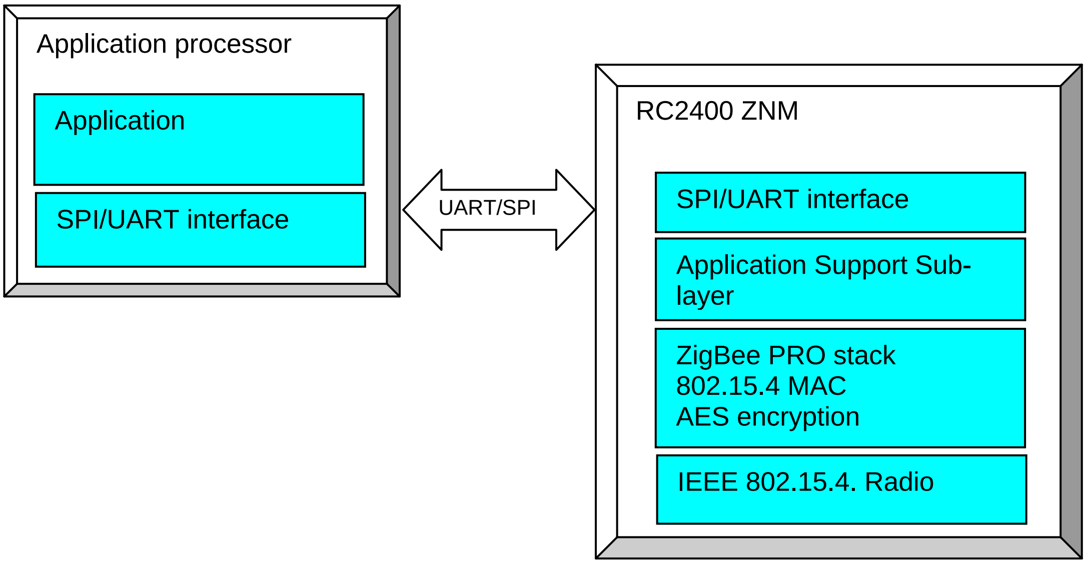


Figure 1 ZigBee Network Module concept

Using a pre-qualified module is the fastest way to make a ZigBee product and shortest time to market. Because it contains all the RF HW and MCU resources you need in a 100% RF tested and pre-qualified module shorten the qualification and approval process. No RF design or expertise is required to add powerful wireless networking to the product. As an option you can even get the module with integrated antenna. In the simplest case like a home light remote control you only need an external battery and a pushbutton.

About this document

This document is one part of the documentation for the module. See Datasheet for the electrical parameters, RF performance, footprint and PCB layout and regulatory information.

- RC2400/RC2400HP Data sheet
- RC2400/RC2400HP User Manual for developing FW on RC2400 platform

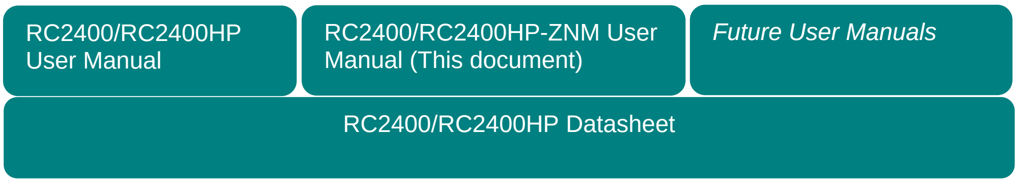
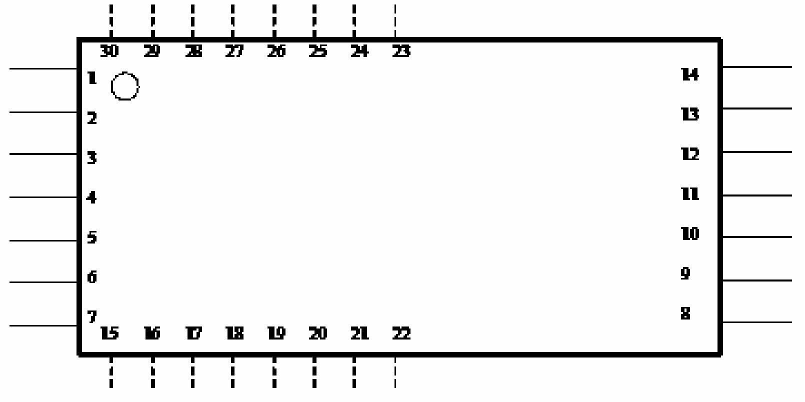

RC2400/RC2400HP User Manual
RC2400/RC2400HP-ZNM User Manual (This document)
Future User Manuals
RC2400/RC2400HP Datasheet

Figure 2 Document structure

Pin Assignment



Pin Description

I

Pin no	Pin name	Description
1	GND	System ground
2	CTS	UART Clear to Send / SPI SRDY
3	RTS	UART Request to Send.
4		
5	TXD	UART TX Data / SPI MRDY
6	RXD	UART RX Data
7	GND	System ground
8	GND	System ground
9	RF	RF I/O connection to antenna
10	GND	System ground
11	NC	Not Connected
12	Reset	RESET_N. Active Low
13	VCC	Supply voltage input. Internally regulated.
14	GND	System ground
15		GPIO
16	ZNM-Cfg0	ZnpCfg0 0 = 32 kHz RTC crystal 1= 32 kHz RC osc
17		GPIO
18	ZNM-Cfg1	ZnpCfg1 '0' = UART '1' = SPI
19	DD	Debug Data. Debug interface is used for programming.
20	DC	Debug Clock. Debug interface is used for programming.
21	GPIO	GPIO
22		**EN for RC2400HP**
23	32kHz_Q1	Internal 32 kHz oscillator. Do not connect.
24	32kHz_Q2	Internal 32 kHz oscillator. Do not connect.
25		SPI MI
26		SPI MO
27		SPI C
28		SPI SS
29		**PA_EN for RC2400HP**
30		GPIO with optional ADC input. LED Driver

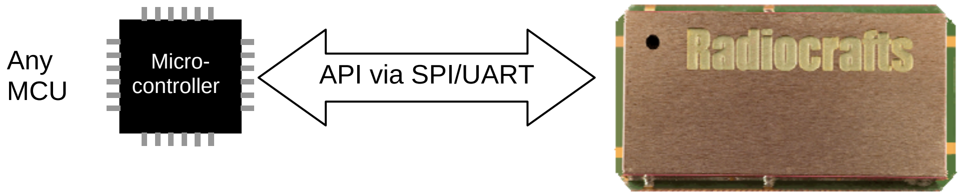
Pin configuration

There are two pins of RC2400 that polled by ZNM FW at startup, and determine how the module is configured.

RC2400/ RC2400 HP pin	Signal name	Result
16	ZNM_Cfg0	0 = 32 kHz RTC crystal 1= 32 kHz RC osc
18	ZNM_Cfg1 (Serial interface selection)	0' Low = UART '1'high = SPI

Serial Communication

Through a serial interface, either SPI or UART, the module/network can be configured and data can be sent and received.



SPI Interface

The SPI interface consists of these signals:

SO	-	Slave output
SI	-	Slave input
CS	-	SPI clock
SS	-	SPI Slave select
MRDY	-	Master ready
SRDY	-	Slave ready

The four upper signals are used for standard SPI operation with RC2400-ZNM as the slave. The MRDY and SRDY are used for power control/flow control. MRDY -> low indicates that the master has data to send and can be used to wake up the ZNM module from sleep. The module will reply with SRDY --> low when it is ready to receive data.

The SPI interface has the following characteristics:

- RC2400-ZNM is an SPI slave
- Max clock speed = 4 MHz
- Clock polarity on RC2400-ZNM = 0
- Clock phase on RC2400-ZNM = 0
- Bit order MSB first

UART Interface

The UART interface is implemented as DTE and consists of these signals

RX	-	RXD - data to module
TX	-	TXD - data from module
CTS	-	Input to module
RTS	-	Output from module

The setting for the UART is as follows:

UART Configuration	
Baud rate	115.2 kBaud
Data bits	8
Parity	Even
Stop bit	1
Flow control	RTS/CTS (implemented as DTE)

The frame format for the UART is as follows:

Start Of Frame(SOF)	Commands	Frame Check Sum- FCS (1 byte)
0xFE	General frame format	XOR of all bytes in General Data Format

General frame format

The general frame format for sending commands is as follow:

Length of data	Command ID		Data
1 byte	CMD0	CMD1	0-253 bytes
0xNN	0xNN	NN	0xNN NN ...

API command set

The set of API commands that can be sent via the UART/SPI interface can be divided into four categories:

- System commands
- Simple API (SAPI) commands
- AF commands
- ZDO commands

System commands are for controlling the HW device and include commands for resetting the module and utilizing resources within the module.

Simple API commands consist of only 10 commands which is the easiest way to build a complete application that does network creation and sending/receiving of data.

AF commands are commands for registering application and sending data with complete flexibility.

ZDO commands are commands for detailed control of ZigBee device operation regarding ZigBee Device Object. This includes binding devices, finding and matching descriptors.

The complete set of commands **to** the module is shown below:

System commands	Simple API commands	AF commands	ZDO commands
SYS_RESET_REQ	ZB_READ_CONFIGURATION	AF_REGISTER	ZDO_NWK_ADDR_REQ
SYS_VERSION	ZB_WRITE_CONFIGURATION	AF_DATA_REQUEST	ZDO_IEEE_ADDR_REQ
SYS_OSAL_NV_WRITE	ZB_APP_REGISTER_REQUEST		ZDO_NODE_DESC_REQ
SYS_OSAL_NV_READ	ZB_START_REQUEST		ZDO_SIMPLE_DESC_REQ
SYS_OSAL_START_TIMER	ZB_PERMIT_JOINING		ZDO_ACTIVE_EP_REQ
SYS_OSAL_STOP_TIMER	ZB_BIND_DEVICE		ZDO_MATCH_DESC_REQ
SYS_RANDOM	ZB_ALLOW_BIND		ZDO_USER_DESC_REQ
SYS_ADC_READ	ZB_SEND_DATA_RQUEST		ZDO_USER_DESC_SET
SYS_GPIO	ZB_GET_DEVICE_INFO		ZDO_END_DEVICE_ANNCE
	ZB_FIND_DEVICE_REQUEST		ZDO_END_DEVICE_BIND_REQ
			ZDO_BIND_REQ
			ZDO_UNBIND_REQ
			ZDO_MGMT_LQI_REQ
			ZDO_MGMT_LEAVE_REQ
			ZDO_MGMT_PERMIT_JOIN_REQ

For a complete overview of the command interface see *CC2530-ZNP Interface Specification*.

States of operation

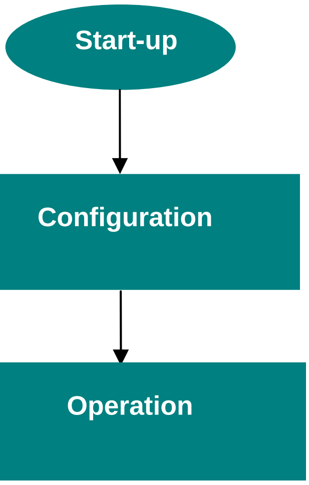


Figure 3 States of operation

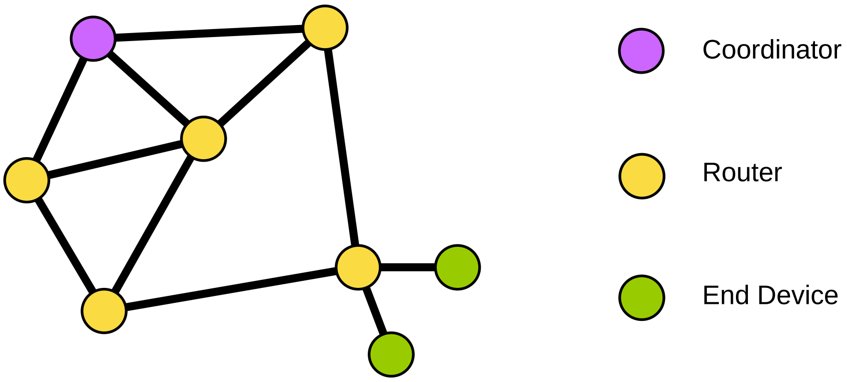
The module has three distinct phases of operation.

- Start-up: An transient phase I/O pins are checked to enable UART or SPI and whether 32kHz crystal oscillator is present. Automatically transition to Configuration state.

- Configuration: Set-up of the ZNM module. (see details below). A start command changes state to Operation

- Operation: The device active the RF part and Create/Joins network automatically.

Configuration



In a ZigBee network the devices have different roles. In a network you will always have 1 Coordinator and possible several Routers and End Devices.

- The ZigBee Coordinator is the root/master of the network and starts the network and later holds information on the network
- A ZigBee Router (Full Functional Device - FFD from IEEE 802.15.4) is an always-on device that including routing functionality.
- A ZigBee End Device (Reduced Functional Device - RFD from IEEE 802.15.4) is a device with no routing capabilities, but with sleep capability. Such a device can sleep most of the time and only poll the network at regular interval.

A ZigBee network is identified by a unique PAN-ID. This ID can be written to the module during configuration. Writing 0XFFFF to the PAN ID will make the Coordinator chose a random PAN-ID (after scan) and Routers/End Devices to join a random PAN.

ZigBee utilises acknowledgement and retransmission on MAC layer. This means that each point-to-point will include this. But in addition an application end -to-end acknowledgement can be included.

ZigBee include a powerful AES128 encryption. The encryption key can be preconfigured in each device or it can be set in the coordinator and distributed to the rest of the network depending on the security requirements.

The output power needs to be set to comply with national regulation. See []

Configuration parameter	
ZCD_NV_STARTUP_OPTION	
ZCD_NV_LOGICAL_TYPE	Coordinator/Router/End Device
ZCD_NV_POLL_RATE	Setup for end device polling
ZCD_NV_QUEUED_POLL_RATE	
ZCD_NV_RESPONSE_POLL_RATE	
ZCD_NV_POLL_FAILURE_RETRIES	
ZCD_NV_INDIRECT_MSG_TIMEOUT	
ZCD_NV_APS_FRAME_RETRIES	Setup for application acknowledge and retransmission
ZCD_NV_APS_ACK_WAIT_TIMEOUT	
ZCD_NV_BINDING_TIME	
ZCD_NV_USER_DESCRIPTION	
ZCD_NV_PAN_ID	PAN-ID
ZCD_NV_CHANLIST	
ZCD_NV_PRECFGKEY	Setup for use of encryption
ZCD_NV_PRECFGKEY_ENABLE	
ZCD_NV_SECURITY_MODE	
ZCD_NV_BCAST_RETRIES	
ZCD_NV_PASSIVE_ACK_TIMEOUT	
ZCD_NV_BCAST_DELIVERY_TIME	
ZCD_NV_ROUTE_EXPIRY_TIME	
ZCD_NV_OUTPUT_POWER	

Before transition to Operation state the application must also be setup in the ZNM module. For each ZigBee application in the following parameters are needed.

- End Point
- Profile ID
- Device ID
- Input/output clusters (or input/output commands)

End point is the logical address given to an application as you can have several applications for one physical radio. (Same principle as USB/Bluetooth or UDP)

Profile ID identifies the profile the application follows. It might be an open profile or a manufacturer specific.

Device ID is used to identify which device within the profile is used.

A cluster is a set of attributes and/or commands in a server to provide a specific service to a client.

E.g. A on/off light will include a server cluster that include attribute OnOff (Boolean) and the following commands On, Off and Toggle. The cluster ID for On/off cluster is 0x0006.

A client to the on/off light can read the status (OnOff attribute) and send the commands in the cluster. The command ID for the given commands are

Command	Command ID
Off	0x00
On	0x01
Toggle	0x02
Reserved	0x03-0xFF

Operation

An important feature during ZigBee operation is **binding.** A binding is a logical connection for a given cluster between two end points in two different ZigBee devices

A binding is stored in a binding table and enables the use of indirect addressing. This means that the application does not specify the address of the receiving device, but simply specifies the binding to be used.

The command ZB_START_REQUEST starts the ZigBee stack within the RC2400 and the module enter operation state.

The module will automatically join or create a network based on the configuration parameters given above. The next step is to identify the devices to communicate with. This can be done in several different ways.

- Hard coded.

Application in external MCU have hard coded IEEE address to communicate to.
- Find device might be useful to make sure the device is in the network and recover short address
- Binding can then be done to desired end point

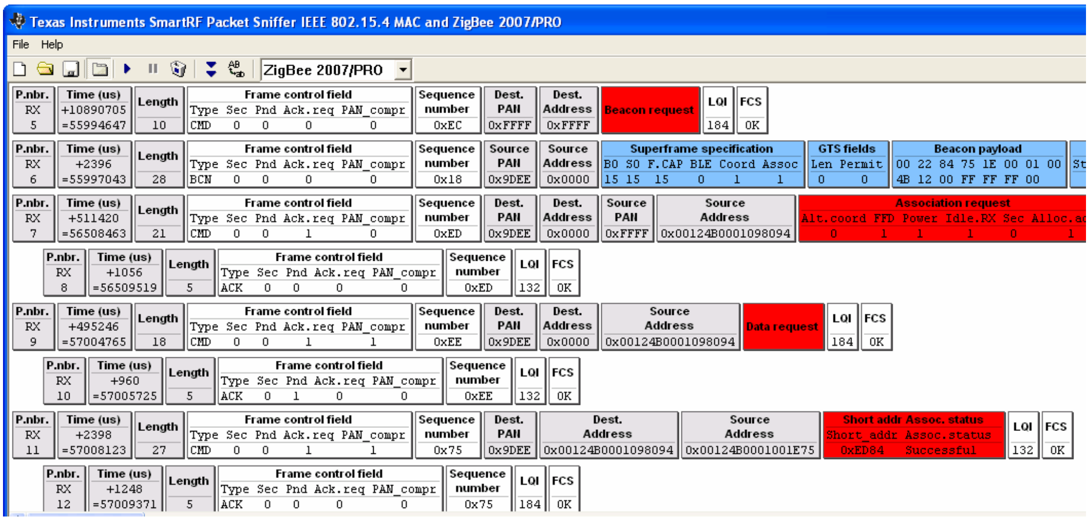
- Semi automatic. The ZigBee device can find appropriate devices with Match descriptor. If several possible devices exist, the binding procedure should include some sort of button push to identify which device to bind to.

Packet sniffer

For evaluating and testing a application on network level a packet sniffer is a useful tool. We recommend using.

- Texas Instruments Packet Sniffer (PC tool)
- CC-debugger
- RC2400DB / RC2400HP-DB

Optionally any other HW with RC2400 module + programming/debugging connector can be used as the physical sniffer.



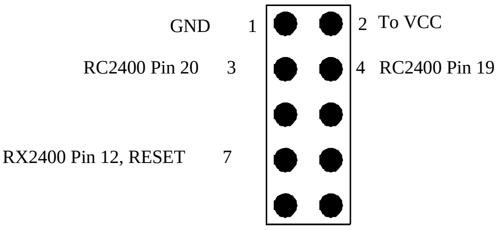
Texas Instruments SmartRF Packet Sniffer IEEE 802.15.4 MAC and ZigBee 2007/PRO
File
Help
ZigBee 2007/PRO

```
P.nbr. | Time (us)  | Length | Frame control field                 | Sequence | Dest.  | Dest.   | Beacon request | LQI | FCS
RX     | +10890705  |        | Type Sec Pnd Ack.req PAN_compr      | number   | PAN    | Address |                |     |
5      | =55994647  | 10     | CMD  0   0   0       0              | 0xEC     | 0xFFFF | 0xFFFF  |                | 184 | OK

P.nbr. | Time (us)  | Length | Frame control field                 | Sequence | Source | Source  | Superframe specification            | GTS fields | Beacon payload          | St
RX     | +2396      |        | Type Sec Pnd Ack.req PAN_compr      | number   | PAN    | Address | BO SO F.CAP BLE Coord Assoc         | Len Permit | 00 22 84 75 1E 00 01 00 |
6      | =55997043  | 28     | BCN  0   0   0       0              | 0x18     | 0x9DEE | 0x0000  | 15 15 15    0   1     1             | 0   0      | 4B 12 00 FF FF FF 00    |

P.nbr. | Time (us)  | Length | Frame control field                 | Sequence | Dest.  | Dest.   | Source | Source             | Association request
RX     | +511420    |        | Type Sec Pnd Ack.req PAN_compr      | number   | PAN    | Address | PAN    | Address            | Alt.coord FFD Power Idle.RX Sec Alloc.ad
7      | =56508463  | 21     | CMD  0   0   1       0              | 0xED     | 0x9DEE | 0x0000  | 0xFFFF | 0x00124B0001098094 | 0         1   1     1       0   1

    P.nbr. | Time (us)  | Length | Frame control field                 | Sequence | LQI | FCS
    RX     | +1056      |        | Type Sec Pnd Ack.req PAN_compr      | number   |     |
    8      | =56509519  | 5      | ACK  0   0   0       0              | 0xED     | 132 | OK

P.nbr. | Time (us)  | Length | Frame control field                 | Sequence | Dest.  | Dest.   | Source             | Data request | LQI | FCS
RX     | +495246    |        | Type Sec Pnd Ack.req PAN_compr      | number   | PAN    | Address | Address            |              |     |
9      | =57004765  | 18     | CMD  0   0   1       1              | 0xEE     | 0x9DEE | 0x0000  | 0x00124B0001098094 |              | 184 | OK

    P.nbr. | Time (us)  | Length | Frame control field                 | Sequence | LQI | FCS
    RX     | +960       |        | Type Sec Pnd Ack.req PAN_compr      | number   |     |
    10     | =57005725  | 5      | ACK  0   1   0       0              | 0xEE     | 132 | OK

P.nbr. | Time (us)  | Length | Frame control field                 | Sequence | Dest.  | Dest.              | Source             | Short addr Assoc. status | LQI | FCS
RX     | +2398      |        | Type Sec Pnd Ack.req PAN_compr      | number   | PAN    | Address            | Address            | Short_addr Assoc.status  |     |
11     | =57008123  | 27     | CMD  0   0   1       1              | 0x75     | 0x9DEE | 0x00124B0001098094 | 0x00124B0001001E75 | 0xED84     Successful    | 132 | OK

    P.nbr. | Time (us)  | Length | Frame control field                 | Sequence | LQI | FCS
    RX     | +1248      |        | Type Sec Pnd Ack.req PAN_compr      | number   |     |
    12     | =57009371  | 5      | ACK  0   0   0       0              | 0x75     | 184 | OK
```

Figure 4 Screenshot from packet sniffer

Programming Interface

For update firmware (e.g. custom variant requirements) to the module it is required to include a 2x5 pins programming connector to the module programming pins. The connector should be a 2.54 mm pitch pin-row (same pitch in both directions), SMD or through-hole version, with the following connections:



Document Revision History

Document Revision	Changes
1.0	First release

Disclaimer

Radiocrafts AS believes the information contained herein is correct and accurate at the time of this printing. However, Radiocrafts AS reserves the right to make changes to this product without notice. Radiocrafts AS does not assume any responsibility for the use of the described product; neither does it convey any license under its patent rights, or the rights of others. The latest updates are available at the Radiocrafts website or by contacting Radiocrafts directly.

As far as possible, major changes of product specifications and functionality, will be stated in product specific Errata Notes published at the Radiocrafts website. Customers are encouraged to check regularly for the most recent updates on products and support tools.

Trademarks

RC232™ is a trademark of Radiocrafts AS. The RC232™ Embedded RF Protocol is used in a range of products from Radiocrafts. The protocol handles host communication, data buffering, error check, addressing and broadcasting. It supports point-to-point, point-to-multipoint and peer-to-peer network topologies.

All other trademarks, registered trademarks and product names are the sole property of their respective owners.

Life Support Policy

This Radiocrafts product is not designed for use in life support appliances, devices, or other systems where malfunction can reasonably be expected to result in significant personal injury to the user, or as a critical component in any life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness. Radiocrafts AS customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Radiocrafts AS for any damages resulting from any improper use or sale.

Contact Information

Web site: www.radiocrafts.com
Email: radiocrafts@radiocrafts.com

Address:
Radiocrafts AS
Sandakerveien 64
NO-0484 OSLO
NORWAY

Tel:	+47 4000 5195
Fax:	+47 22 71 29 15
E-mail:	sales@radiocrafts.com support@radiocrafts.com