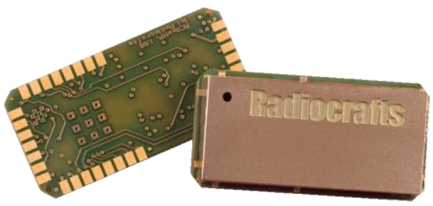
Radiocrafts
Embedded Wireless Solutions

RC24xx-ZNM/RC24xxHP-ZNM

ZNM Network Module - User Manual



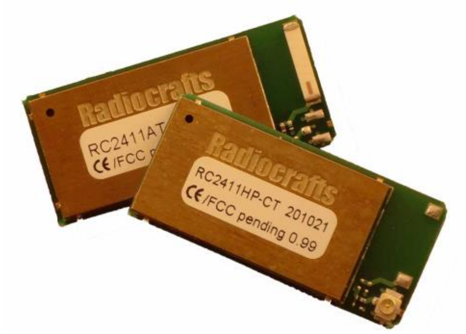


Table of contents

Introduction

This document includes or refers to all the needed information to develop solution with the RC24xx-ZNM and RC24xxHP-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 done with the tool and platform most convenient to the developer.

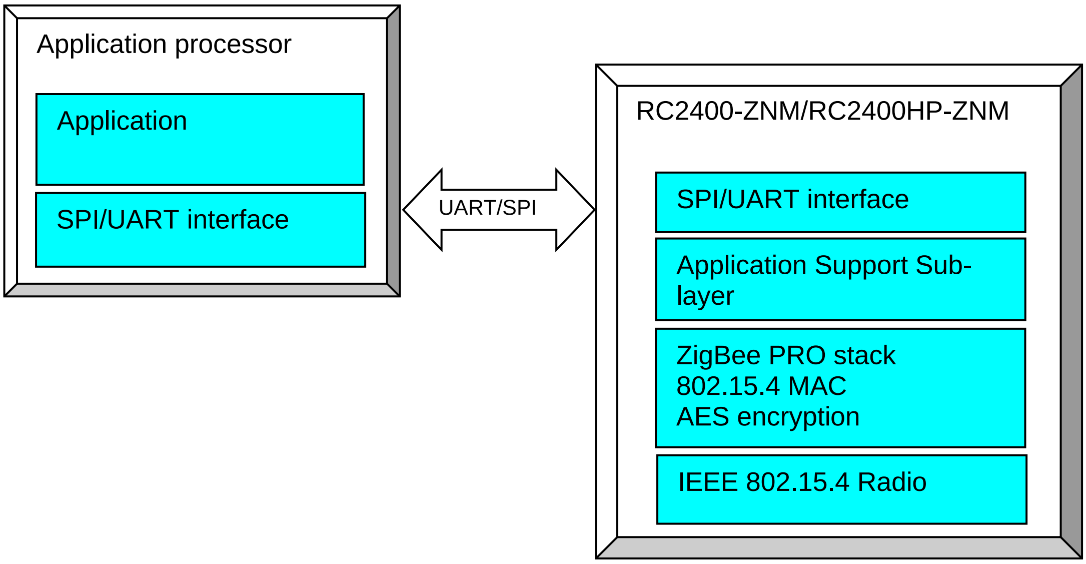


Figure 1 ZNM Module concept

Using a pre-qualified module is the fastest way to make a ZigBee product with shortest time to market. With all the RF HW and MCU resources you need in a 100% RF tested and pre-qualified module the qualification and approval process is shortest possible. No RF design or expertise is required to add powerful wireless networking to any product.

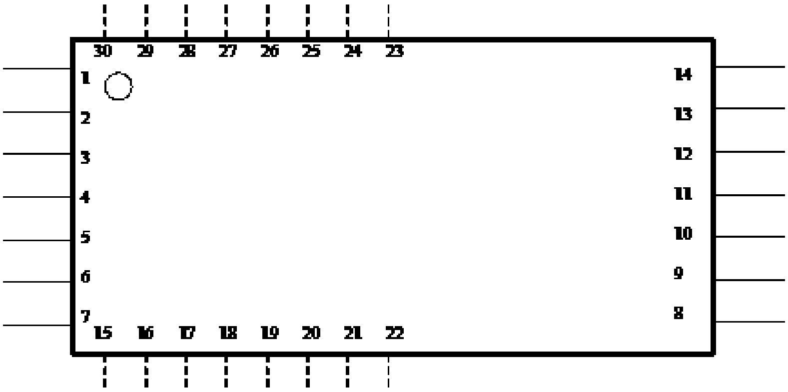
Documentation structure

This document is one part of the documentation for the module. The data sheet describes the electrical parameters, RF performance, footprint and PCB layout and regulatory information. Depending on the selected FW solution, additional User Manuals should be used. The available documents for the RC24xx product series are:

- RC2400/RC2400HP Data sheet
- RC241x Data sheet
- RC2400/RC2400HP Firmware Development User Manual - Details on how to develop customer specific firmware for RC2400 HW platform
- RC24xx/RC24xxHP-ZNM User Manual (This document)

Figure 2 Document structure

Pin Assignment RC2400/RC2400HP

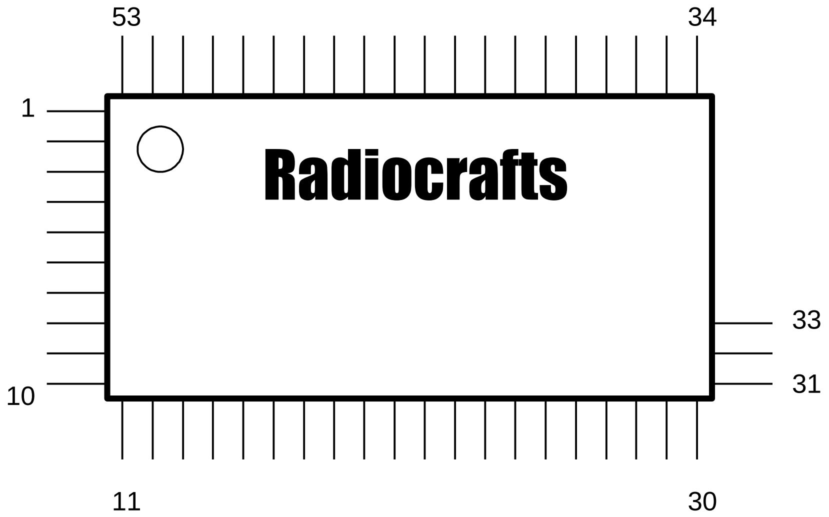


Pin Description

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		**LNA High Gain mode for RC2400HP. Do not connect**
16	ZNM-Cfg0	ZnmCfg0 0 = 32 kHz RTC crystal oscillator 1= 32 kHz RC oscillator
17		GPIO
18	ZNM-Cfg1	ZnmCfg1 '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. Do not connect**
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. Do not connect**
30		GPIO with optional ADC input. LED Driver

Pin Assignment RC241x/RC241xHP



Pin Description RC241x/RC241xHP

Pin no	Pin name	Description and internal MCU connection
1	GND	System ground
2	NC	Not connected
3	NC	Not connected
4	GND	System ground
5	CTS	UART Clear to Send / SPI SRDY
6	RTS	UART Request to Send.
7		
8	TXD	UART TX Data / SPI MRDY
9	RXD	UART RX Data
10	GND	System ground
11	GND	System ground
12		**HGM** for PA CTRL IN HP VERSION Do not connect for HP version
13	ZNM-Cfg0	ZnmCfg0 0 = 32 kHz RTC crystal oscillator 1= 32 kHz RC oscillator
14	GPIO	
15	NC	Not connected
16		**ENABLE(LNA_ENABLE)** FOR PA CTRL IN HP VERSION Do not connect for HP version
17	RESET_N	RESET
18	NC	Not connected
19	NC	Not connected
20	NC	Not connected
21	NC	Not connected
22	NC	Not connected
23	NC	Not connected
24	NC	Not connected
25	NC	Not connected
26	NC	Not connected
27	NC	Not connected

28	NC	Not connected
29	NC	Not connected
30	GND	System ground
31	GND	System ground
32	RF_TEST	RF I/O connection for Automatic test purposes. - For components intended for use with UFL connector, do not connect this pad.
33	GND	System ground
34	GND	System ground
35	VCC	VCC
36	NC	Not connected
37	NC	Not connected
38	NC	Not connected
39	NC	Not connected
40	NC	Not connected
41	NC	Not connected
42	NC	
43	NC	
44	DC	DC, used for Firmware upgrade
45	DD	DD, used for Firmware upgrade
46	ZNM-Cfg1	ZnmCfg1 '0' = UART '1' = SPI
47		SPI MI
48		SPI MO
49		SPI C
50		SPI SS
51		**PA ENABLE** FOR PA CTRL IN HP VERSION Do not connect for HP version
52	GPIO	GPIO with optional ADC input. LED Driver
53	GND	System ground

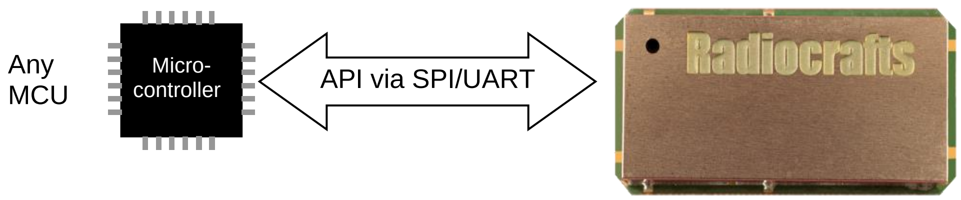
Pin configuration

There are two pins of RC2400 that are used to hardwire the configuration of the module:

RC2400/ RC2400 HP pin	Signal name	Result
16	ZNM_Cfg0	'0' low = 32 kHz RTC crystal oscillator. '1' high = 32 kHz RC oscillator
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:

- RC24xx-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)

*Contact sales@radiocrafts.com for other Baud rates

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 1 byte	Command ID CMD0	 CMD1	Data 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.

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

States of operation

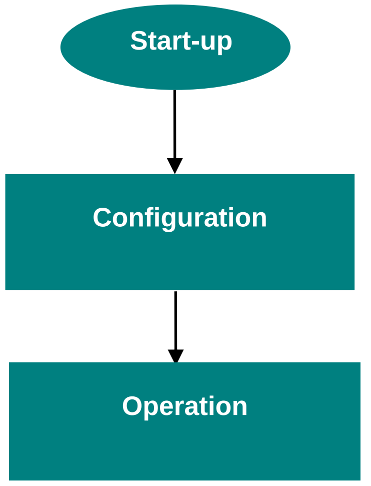


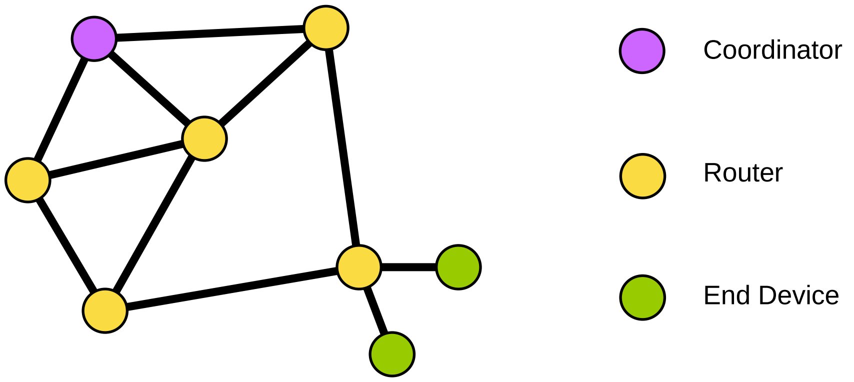
Figure 3 States of operation

The module has three distinct phases of operation.

- Start-up: At this transient phase configuration I/O pins are checked to enable UART or SPI and whether 32 kHz 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

This chapter describes some of the features configured in Configuration state.



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.

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 profile.

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. an 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 IDs for the given commands are

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

Operation

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

The module will automatically join or create a network based on the configuration parameters given above. The state of this joining process will be reported with state messages via serial API. Routers are default set up to act as coordinator is no coordinator is found.

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 next step is to identify the devices to communicate with. This can be done in several different ways.

- Hard coded.
    Application in external MCU has 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.

API command set

The API command set is defined in *CC2530-ZNP Interface Specification* with following changes and additions.

SET_TX_POWER

SREQ

1	1	1	1	1
Length = 0x02	CMD0 = 0x21	CMD1 = 0x0F	00	TX_POWER

SRSP

1	1	1	1
Length = 0x01	CMD0 = 0x61	CMD1 = 0x0F	Status

TX_POWER	Output power RC2400HP (dBm)	Output power RC2400 (dBm)
0xED	20	3
0xEE	19	1
0xEF	18	-1
0xF0	17	-2
0xF1	15	-4
0xF2	14	-5
0xF3	13	-6
0xF4	13	-6
0xF5	11	-8
0xF6	9	-10
0xF7	9	-10
0xF8	9	-10
0xF9	7	-12
0xFA	7	-12
0xFB	5	-14
0xFC	5	-14
0xFD	3	-16
0xFE	3	-16
0xFF	1	-18

Table 1 Typical output power levels

RF_TEST_MODE

To set the module in test modes the module must be reset after the SREQ/SRSP communication below.

To escape test mode a physical reset is required.

SREQ

1	1	1	4	1	1	1	1
Length = 0x02	CMD0 = 0x21	CMD1 = 0x09	0x07 0F 00 04	MODE	CHANNEL	TX_POWER	MDMTEST0

MODE	
0x01	RX
0x02	TX Carrier
0x03	TX Modulated signal

CHANNEL	Frequency (MHz)
0x0B	2405
0x0C	2410
0x0D	2415
0x0E	2420
0x0F	2425
0x10	2430
0x11	2435
0x12	2440
0x13	2445
0x14	2450
0x15	2455
0x16	2460
0x17	2465
0x18	2470
0x19	2475
0x1A	2480

TX_POWER	Typical output power RC2400HP* (dBm)	Typical output power RC2400 (dBm)
0xF5	20	3
0xE5	19	2
0xD5	18	1
0xC5	17	-1
0xB5	16	-3
0xA5	15	-4
0x95	13	-6
0x85	12	-7
0x75	10	-9
0x65	8	-11
0x55	6	-13
0x45	4	-15
0x35	2	-17
0x25	0	-19
0x15	-2	-21
0x05	-4	-23

*See datasheet for regulatory information on allowed output power

SRSP

1	1	1	1
Length = 0x01	CMD0 = 0x61	CMD1 = 0x09	Status

AF_DATA_REQUEST
The ***Option*** byte in AF_DATA_REQUEST is interpreted with the following bit mask

Bit 7	6	5	4	3	2	1	0
Skip routing	APS security	Discover route	APS ACK	Reserved, Set to '0'			

ZDO callback

The ZNM firmware is setup to give callbacks according to RSP and IND messages in CC2530ZNP Interface Specification. There is an option to default disable these and to force the application to register for the specific ZDO callbacks the application want to receive. To disable the RSP and IND messages write (using SYS_OSAL_NV_WRITE) value 0x00 to address 0x008F.

To register for the specific callback use the ZDO_MSG_CB_REGISTER function. The callback will in this case be received as ZDO_MSG_CB_INCOMING, and not with IND and RSP messages.

Packet sniffer
For evaluating and testing an 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.

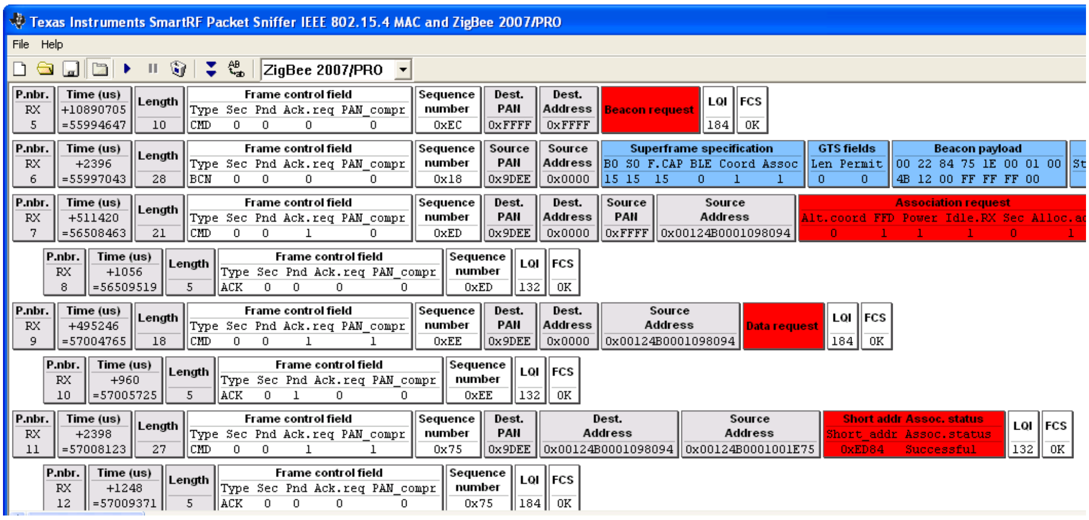


Figure 4 Screenshot from packet sniffer

KEY_ESTABLISHMENT_INIT

SREQ

1	1	1	1	1	1	1	2/8
Length = 0x0?	CMD0 = 0x27	CMD1 = 0x80	TASK ID	SECUENCE NUMBER	END POINT	ADDR Type	Address

ADDR TYPE = 0x02 = short address (In this case address field is 2 bytes)
0x03= 64 bits address (In this case address field is 8 bytes)

SRSP

1	1	1	1
Length = 0x01	CMD0 = 0x67	CMD1 = 0x80	Status

KEY_ESTABLISHMENT_IND

AREQ

1	1	1	1	1	1	1	2
Length = 0x06	CMD0 = 0x47	CMD1 = 0xE1	TASK ID	EVENT	STATUS	WAITTIME	SUITE

KEY_ESTABLISHMENT_ECDSA_SIGNATURE

SREQ

1	1	1	1	INPUT LENGTH
Length = 0x0x	CMD0 = 0x27	CMD1 = 0x81	INPUT LENGHT	INPUT

SRSP

1	1	1	1	42
Length = 0x2B	CMD0 = 0x67	CMD1 = 0x81	STATUS	Key

CERTIFICATES

In order for the key establishment algorithm to work the device need to have a valid certificate. Certificates are currently only available from Certicom (www.certicom.com). There are both test-certificates (free) and productions certificates available.

The certificate is tied to the IEEE address of the devices.

The certificate can be written to the module with the SYS_OSAL_NV_WRITE command with the following addresses. Note that these are written as MSB first (in contradiction to other parameters in ZNM)

Address 0x0069 = Certificate
Address 0x006A = Private Key
Address 0x006B = CA Public key

For simplicity, the tools from Texas Instruments called Z-Converter and Z-Tool can assist in writing the certificate into the module on the demo boards.

Document Revision History

Document Revision	Changes
1.0	First release
1.1	Added info on ZNM-SE variant
1.2	Added info on RC241x modules
1.3	ZNM Update

Disclaimer

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.

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