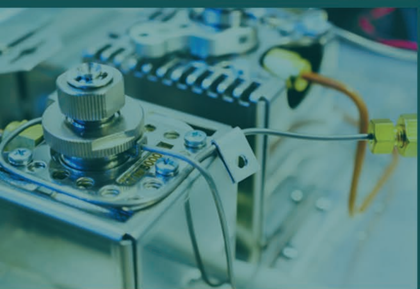


Radiocrafts
Embedded Wireless Solutions

AN024: Wireless M-Bus in Industrial Sensor Networks

APPLICATION NOTE



We Make Embedded Wireless
Easy to Use

Using Wireless M-Bus in Industrial Sensor Networks

By P.M.Evjen

Introduction

Wireless M-Bus (EN 13757-4:2013) is a wireless communication standard that was developed for the reading of electricity, gas-, water-, heat-meters and heat allocators. However, the reading of utility meters is very similar to applications in the industry where sensors or actuators need to be read or operated at a distance. Very often these sensors are battery operated, it is difficult to install wires to reach them, and there are requirements to the security and the reliability of the data communication. All these issues have already been addressed in the development of the Wireless M-Bus standard. Therefore we will find that Wireless M-Bus is also a very good alternative as a communication standard for industrial wireless sensor networks in general.

The advantage of using an established standard is that it has been developed by teams of industry experts. The solutions are well established and proven in the field. Reading billing information from remote utility meters puts very high requirements to the protocol in terms of security, reliability and battery lifetime. The standard is like a toolbox where we can pick and use the right tool in different situations; that is, operating frequency, power consumption, range, encryption and other security elements.

This Application Note gives an understanding of how to use a Wireless M-Bus module in an industrial sensor network. First we describe the communication protocol and its different layers. Next, the details in the Wireless M-Bus messages are described, and the details of the interface between an external application processor and the RF module. And finally a few words about the module configuration.

Wireless M-Bus overview

Wireless M-Bus has its root in the wired M-Bus user group that was very active in the 90's. Initially they created the wired M-Bus standard for meter reading. Later that standard became part of the European standard EN1434 for reading heat energy meters.

The M-Bus specification was later enhanced, and *Wireless* M-Bus was introduced. This standardization work was transferred to CEN, technical committee TC294, that created the new European standard EN 13757 "Communication system for meters and remote reading of meters". This standard currently (as of May 2017) consists of:

- EN13757-1:2014 Data exchange
- EN13757-2:2004 Physical and link layer
- EN13757-3:2013 Dedicated application layer
- EN13757-4:2013 Wireless meter readout
- EN13757-5:2015 Relaying
- EN13757-6:2015 Local bus

As seen, the part -4 introduced wireless meter reading that will be discussed in this document. Wireless M-Bus is a radio and data link specification. The application layer for M-Bus is specified in EN 13757-3.

To get all the details, you need to buy the standard. It is widely available online from the national standardization organizations. Here is one example:

http://shop.bsigroup.com/SearchResults/?q=EN13757

The Wireless M-Bus network

Wireless M-Bus specifies a meter device (Slave), or in our case a sensor; and an "Other" device or “Communication partner” (Master). The "other" is often also referred to as a concentrator or gateway. As this Application Note focus on the use of sensors, we will refer to the Slave device as a Sensor.

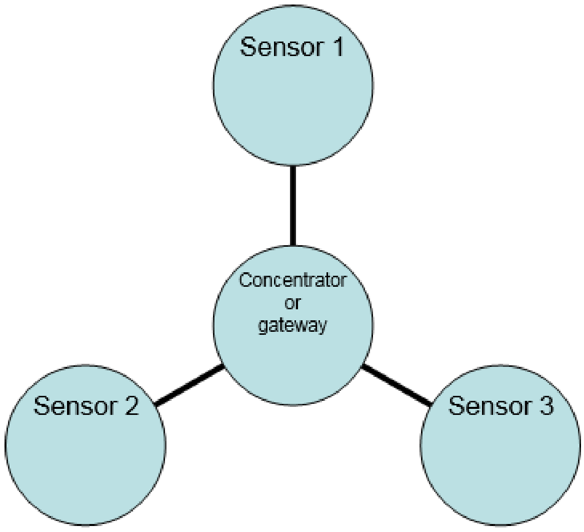


Figure 1. Wireless M-Bus network

The EN13757-4 specifies the physical layer (PHY) and the data link layer (DLL) including the medium access (MAC) for communication between the sensor and the concentrator. This includes the specification of:

- Radio parameters (frequency, data rate, modulation)
- Packet frame format (preamble, synchronization, CRC)
- Access method (transmit, then listen)
- Link Layer messages (request, response, acknowledge)

The Wireless M-Bus specification has several options for the radio parameters. The different Wireless M-Bus modes define data rates and operating frequency (radio channel), and both one-way and two-way communications are included. For example, “T1” means T-mode one-way, while “T2” means T-mode two-way.

Operating frequency and modes

Wireless M-Bus devices can operate in the VHF band at 169 MHz, or in the UHF band at 868 MHz. (Modes are also defined for 433 MHz, but that is outside the scope of this Application Note.)

Mode N is operating at 169 MHz. N stands for “narrowband”, indicating the signal bandwidth is “narrow”, that is 12.5 kHz. The data rate can be 2.4 kbps or 4.8 kbps. And there are 6 radio channels within the 169 MHz band to choose from. (There is a 50 kHz channel for 19.2 kbps, but this option is not commonly used.)

Mode T and C, are the mostly used modes at 868 MHz. The T-mode is intended for systems with frequent transmissions keeping transmissions short (compared to mode S), using a high data rate of 100 kbps. The bits are coded using a 3of6 code, which means the effective data rate (bit rate) is 67 kbps. The C-mode was introduced in the standard to allow a more efficient coding using NRZ (bits are modulated without additional coding). Because mode T and C share the same frequency, and the pre-ambles are aligned, it is possible to make a receiver that can detect both modes without reconfiguration. The RC1180-MBUS3 is able to do this, using “T + C” mode.

In general, 169 MHz gives a longer range, but requires a larger antenna to be efficient. Transmission power up to 500 mW can be used, but this also increase the current consumption and peak load from the battery. Still it is possible to build battery operated devices with 15-20 years lifetime. Due to the low data rate, each transmission will be longer. It is therefore most suitable for non-frequent transmitters. For example transmitters that operates less than once per hour.

For lower power consumption, 868 MHz with mode T or C, can be used. The higher data rate gives shorter transmissions, and it is therefore feasible to have more frequent transmissions. In a meter reading application we talk about “walk-by” or “drive-by” reading which require transmissions every 10-60 seconds. The same can apply for sensors that need to be monitored every few seconds.

Medium access

One important concept in Wireless M-Bus, as least for battery operated sensors, is that it is always the “slave” – the sensor – that initiates the communication. This is especially important to note if two-way communication is required.

The typical use is that the sensor is in sleep mode (saving battery power), but wakes up on a timer or an event. Then in will do a transmission of the sensor data. If it is a one-way system, the sensor simply goes back to sleep.

If it is a two-way system, either used to configure the sensor, ask for more data or to update an actuator, the sensor will listen for data from the concentrator only in a very short time window after the transmission was done.

In T-mode, this window is only open 2-3 ms after the transmission. If no message preamble is received within this window, the sensor will go back to sleep. This is the key technique used in Wireless M-Bus to keep the current consumption at a very low level, even for two-way communication. The response time, or latency, from the concentrator to the sensor is therefore determined by how often the sensor will do its transmission and open its listening window.

On the other side, the concentrator module will always listen for new messages. But it must also be prepared to send any command very quickly after it received a message, in order to catch the listening window of the sensor. The Wireless M-Bus modems from Radiocrafts include some patented techniques to meet these timing requirements by using an intelligent message generator and mailbox system.

Radio modules supporting Wireless M-Bus

Radiocrafts provides a selection of modules based on the Wireless M-Bus protocol. These can be divided into two groups; radio modems and autonomous transmitters.

A radio modem is controlled by a host MCU (application processor) over the UART interface. Typically the application processor sends the application data to the modem, which will do the encryption, coding, modulation and transmission on the desired radio channel. The RC1180-MBUS3 and RC1701HP-MBUS4 are examples of such modems. The “modem” will also handle the time critical tasks when doing two-way communication.

In the autonomous transmitter, the “application” is also embedded in the module. The module will read the sensor (or count pulses from a meter), build the message and do the transmission based on a time scheduler. The RC1180-MPC1 and RC1701HP-MPC1 are examples of such modules with integrated pulse counters. They will also report temperature based on an internal sensor, and the battery voltage.

In the following we will look closer into how the modem module can be used, when an external application processor is connected to the actual sensor.

The tables below are summaries of the Wireless M-Bus modules currently provided by Radiocrafts. For the most recent offering, please check the web page.

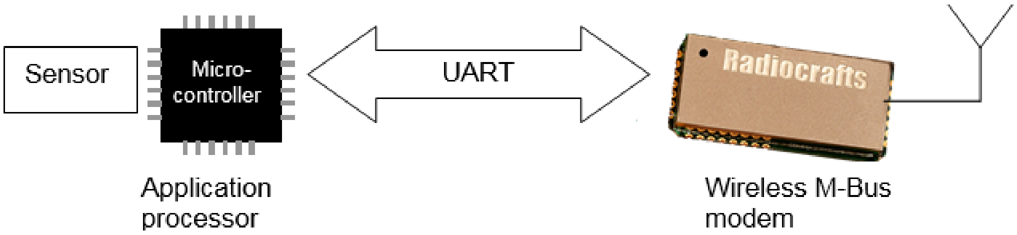
Wireless M-Bus modems			
Part number	**Frequency band**	**Mode**	**Data rate**
RC1140-MBUS3	433 MHz (non-Europe)	S1 / S2 T1 / T2	32.768 kb/s 100 kb/s*
RC1160-MBUS3	868 MHz (Russia)	T1 / T2	100 kb/s*
RC1170-MBUS3	865 MHz (India)	S1 / S2 T1 / T2	32.768 kb/s 100 kb/s*
RC1180-MBUS3	868 MHz	S1 / S2 T1 / T2 C1 / C2	32.768 kb/s 100 kb/s* 100 kb/s
RC1701HP-MBUS4	169 MHz	N1 / N2	2.4 kb/s 4.8 kb/s 19.2 kb/s

* T2 mode use 32.768 kbps in the downlink, same as S mode.

Wireless M-Bus autonomous transmitters			
Part number	**Frequency band**	**Mode**	**Data rate**
RC1140-MPC1	433 MHz (non-Europe)	S1 / S2 T1 / T2	32.768 kb/s 100 kb/s*
RC1160-MPC1	868 MHz (Russia)	T1 / T2	100 kb/s*
RC1170-MPC1	865 MHz (India)	S1 / S2 T1 / T2	32.768 kb/s 100 kb/s*
RC1180-MPC1	868 MHz	S1 / S2 T1 / T2	32.768 kb/s 100 kb/s*
RC1701HP-MPC1	169 MHz	N1 / N2	2.4 kb/s 4.8 kb/s 19.2 kb/s

* T2 mode use 32.768 kbps in the downlink, same as S mode.

Using the Wireless M-Bus modem



When using the Wireless M-Bus modem, all communication with the module is done over the UART interface. The modem can be used on the sensor side, as “transmitter”, or at the Concentrator side, as a “receiver”. In fact the modules are able to both transmit and receive (transceivers).

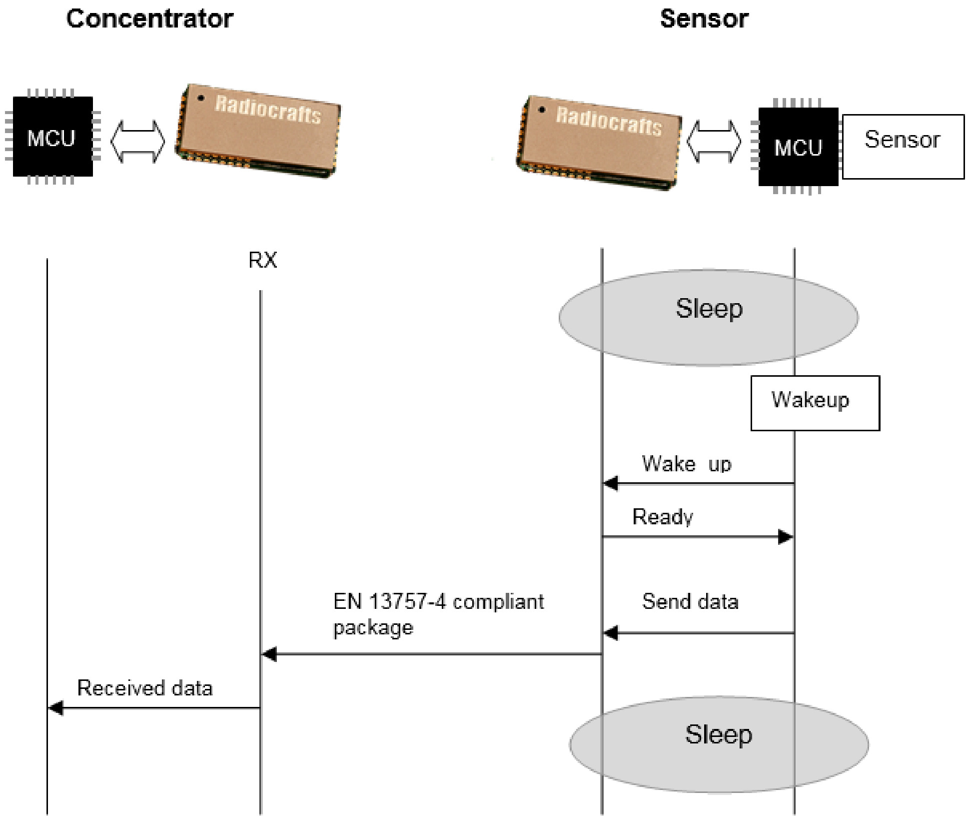


Figure 3. One way communication from sensor to concentrator

Protocol layer control information

Each layer in the protocol starts with a control byte. The Link Layer starts with the C-field, which is used to indicate what kind of message it is. For example, C = 0x44 means SND-NR (Send-no-Reply) and is used to send data from the sensor without expecting a response. C = 0x43 means SND-UD (Send User Data) and is used by a concentrator to send data or a command to the sensor.

The transport layer (which acts as a header for the application layer) starts with the CI-field (Control Information). For example, CI = 0x7A indicates a transmission from the sensor using a *short* header (no address, as the address is contained in the link layer). CI = 0x5B means it is a command to the sensor from the Concentrator using a *long* header (including the address of the sensor).

Inside the application layer, each data point starts with a DIF (Data Information Field) and VIF (Value Information Field). These are used to specify the actual meaning of the data value that follows.

Communication Scenarios

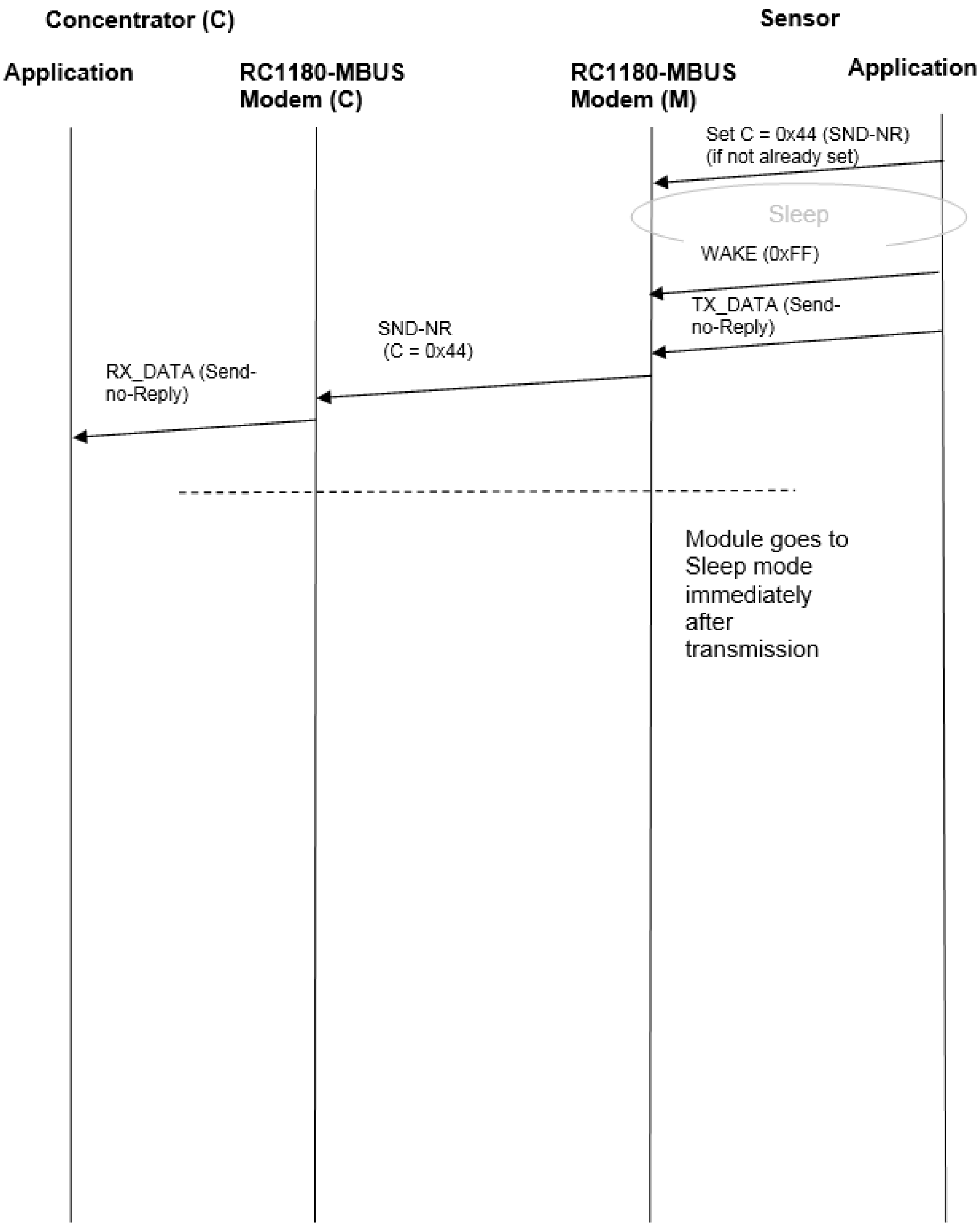
The two-way communication described in the following is always initiated by the Sensor application. For the standard Communication Module ("modem") implementation it is an external processor that runs the application. Communication is initiated by the module itself only if the application is integrated in the module as for the Application Module implementation.

The following signal diagrams show the communication for typical scenarios.

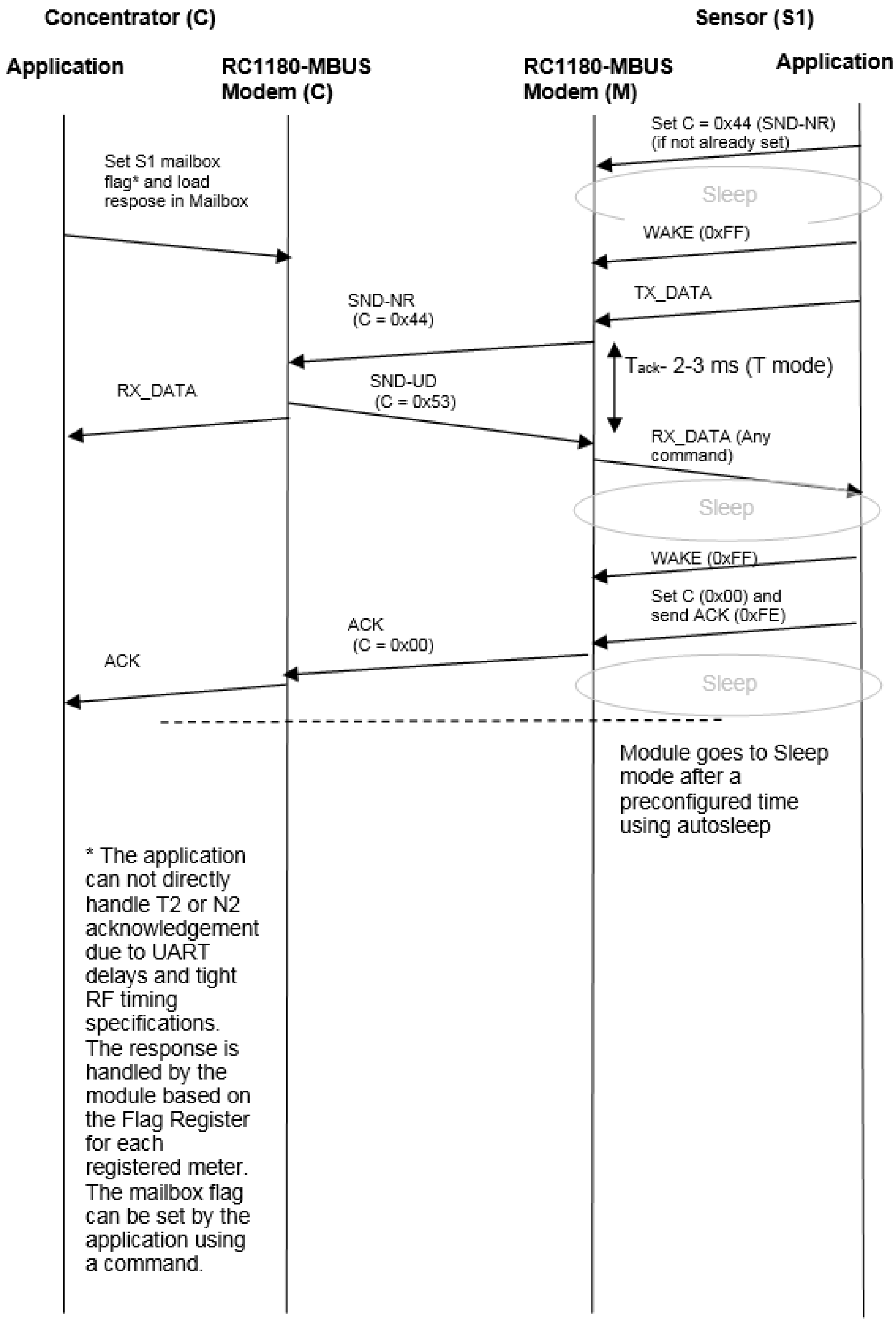
In the first example we show one-way communication, using a mode such as T1 or N1. The C-field defining the link layer message type is set to 0x44, which means Send-no-Reply (SND-NR). That is, the sensor does not expect any response or reply to the message. It is like "send and forget". Normally acknowledge is not used for regular transmissions in Wireless M-Bus. To improve the reliability it increases the number of transmissions instead.

In the second example we show two-way communication, using a mode such as T2 or N2. The sensor will transmit a SND-NR message, but will also listen for any command from the concentrator. The SND-UD, Send User Data, message is used to send a command to the sensor. Finally the sensor responds with an Acknowledge, so that the concentrator can delete the command from its outgoing mailbox. If the Acknowledge was not sent or lost, the concentrator should re-send the command when the Sensor calls in the next time.

Normal one-way data transmission (without ACK)



Data exchange with two-way communication



M-Bus data points

The M-Bus application layer specified in EN13757-3, describes how data points are coded. That is, how the data from a certain meter or sensor is actually transmitted. Each data point (for example a temperature reading) consists of descriptors and the actual value. The descriptors (DIF and VIF) give information on how to understand the value. For example, that the value is volume in litres using an 8 digit number with BCD coding.

Here is an example for a room sensor (temperature or humidity). The Device Type (part of the link layer address) should be set to 0x1B indicating "room sensor".

The temperature can then be coded using DIF = 0x09 which means 2 digit BCD, and
VIF = 0x64 (external temperature in degrees).

The humidity can be coded using DIF = 0x09 which means 2 digit BCD, and VIF = 0x3A (which is dimensionless, here used for percentage).

To set a digital output in an actuator, the DIF = 0x01 which means binary, and VIF = 0xFD, VIFE = 0x1A which is a digital output. In this case the VIF consist of 2 bytes (using an extension byte).

Wireless M-Bus Messages

In the following Wireless M-Bus messages on the air is described, and how the packet is built based on the application data from the host processor.

The following abbreviations are used:

CRC	Cyclic Redundancy Check
DIF	Data Information Field
DIFE	Extended Data Information Field
VIF	Value Information Field
VIFE	Extended Value Information Field

The first block in the message frame is always added by the Communication Modem (e.g. RC1180-MBUS3). Also, CRC (2 bytes) is always added for every 16 bytes (Frame format A). Encryption and filler bytes are added by the Communication Modem if encryption is enabled.

Block	Data	Field	Hex	Remark
First Block	Length	L	XX	Length field
	Control	C	XX	Control field
	Address	M	XX XX	Manufacturer ID
		A	XX, XX, XX, XX, XX, XX,	Address (6 bytes) Identification number (4), Version Number (1), Device Type (1)
	Checksum	CRC	XX, XX	
Second Block, Etc.	Control Information	CI	XX	Second block always start with CI, Control Information field
	The rest of the message is added here, with encryption and additional Filler Bytes if applicable. CRC will be added after encryption.			

The Application processor shall send data to the module as from the CI field. The first byte shall be a Length byte. Filler Bytes shall not be added, CRC shall not be added (as both are added by the module). This is shown here:

Block		Field	Hex	Remark
	Length	L	XX	Length field
	Control	CI	XX	Control Information field
	Data	Variable data field	XX…..	The rest of the message is added here.

\ In the following, this legend is used:

Added by the module	Encrypted data

The Wireless M-Bus specification allows different encryption schemes. In the following examples security mode 05h is shown. Type 05h use AES-128-CBC block cipher.

Note: Encrypted data is not shown as cipher. Actually transmitted data are encrypted in blocks of 16 bytes.

Send-no-Reply (Encrypted)

This is the regular sensor data transmission.

Block		Field	Hex	Remark
First Block	Length	L	XX	Length field
	Control	C	48	Control field
	Address	M	XX XX	Manufacturer ID
		A	XX, XX, XX, XX, XX, XX,	Address (6 bytes) Identification number (4), Version Number (1), Device Type (1)
	Checksum	CRC	XX, XX	
Second Block	Control	CI	7A	Control Information
	Short Header	AN	XX	Access Number
		St	XX	Status (Error info)
		CW	XX, 05	Configuration Word (encryption info)
		Encryption check	2F, 2F	
	Temperature	DIF	09	
		VIF	64	
		Sensor Data	XX, XX	
	Humidity	DIF	09	Variable length ASCII
		VIF	3A	Equipment Identifier
			XX, XX	Equipment Identifier, f.ex. ABCD1234567891234
	Filler	DIF	2F	Encryption Filler as required (added by module)
		CRC	XX, XX	

Module interface

The module interface between application and modem is a UART. This interface is used for both configuration and data exchange.

Data transmissions and configuration
The general format from the application processor to the module is:

1 byte	1 byte	n bytes
Length	CI	APP_LAYER

For general data communication (TX_DATA and RX_DATA shown in the signal diagrams) the Length byte indicates the total length of the string (Length byte itself not included). The CI byte is according to the M-Bus Standard, followed by APP_LAYER application data (typically described by DIF and VIF fields).

When the length byte is 00h the module enters Configuration mode. In Configuration mode the module accepts commands and configuration parameters can be set.

When the length byte is FEh the module will transmit the first block only, and no application data. This can for example be used to send an acknowledgement (set C = 00h first).

Data reception
As the receiver in the concentrator (gateway) will handle several slave meters, it needs the address information from the header. This header is included if DATA_INTERFACE = 0.

From the module to the gateway host controller (with meter address) the format is:

1 byte	1 byte	2 bytes	6 bytes	1 byte	n bytes	1 byte
Length	C	ManID	Address	CI	APP_LAYER	RSSI (opt)

An RSSI byte is appended if RSSI_MODE = 1. The RSSI can be used by the application to determine the link quality.

If DATA_INTERFACE = 1, the address information in the header is omitted. This is typically used in the sensor, or in the concentrator receiver, if only one meter is bound.

From the module to the sensor application controller (without meter address) the format is:

1 byte	1 byte	n bytes	1 byte
Length	CI	APP_LAYER	RSSI (opt)

When the concentrator receives a valid message, the address is automatically extracted from the message, and this address is used for the next transmission. If the concentrator wants to send a message to a device different from the last one received, the destination address must be changed using the “T” command.

Module Configuration Parameters and Commands

The module can be configured with parameters stored in non-volatile or in volatile memory. Frequently changed parameters (like the C-field) are stored in volatile memory. Non-volatile parameters (like Address and Manufacturer ID) are set using the 'M' command, while volatile parameters are set using direct access commands.

Refer to the respective Data Sheet and User Manual for a complete overview of the available commands and configuration parameters.

Some examples of commands used for changing some basic settings are:
G – Set Wireless M-Bus mode (for fast switching between S and T/C mode)
C – Set Channel
P – Set Power
S – Read Signal Strength (RSSI)
Q – Read Quality Indicator
M – Memory config
X - Exit
Z - Sleep
1 - Test mode used to transmit a signal, unmodulated (carrier)
2 - Test mode used to transmit a signal, modulated (PN9)

Some examples of commands used for two-way communication, binding, acknowledgement and encryption:
F – Control Field can be set using this configuration command (default value set using 'M')
I – Installation mode, receive all messages or use address filter
B – Bind new sensors in the Address register
K – Set new sensor encryption key in Key register

For example, to set the Control Field C=48h, use this sequence:
00h – sets module in configuration mode
46h – 'F' command for setting the Control Field
48h – new value for Control Field
58h – 'X' command to Exit from configuration mode

Another example is how the configuration parameters will control the behaviour of the module. If SLEEP_MODE = 1, the module will automatically enter Sleep mode after transmission (in T1 mode) or after 3 ms listen time (in T2 mode). To wake the module, send FFh to the module. The module can also be set in Sleep mode using the 'Z' command.

Some other parameters that can be set in configuration memory:
RX_TIMEOUT: Timeout after data transmission in T2 (combined with Sleep mode enabled).

DATA_INTERFACE:
0 = normal mode sending the complete UART frame
1 = remove header (first block) from UART frame
3 = send 00:3Eh as Acknowledge to the application when an empty application data field is received with C = 00h.

Again, the *complete* overview of the available commands and configuration parameters are found in the respective Data Sheet and User Manual.

Example 1: Temperature monitoring in a building

In this example we consider an application with 100 temperature sensors in a building. We would like to monitor the average temperature every hour. The size of the building would be in the range of 30 by 30 meters, with several floors.

The sensor itself would be read by a small microcontroller. The microcontroller could read the sensor every minute and calculate the average temperature after one hour. Every hour the microcontroller would wake up the Wireless M-Bus module to make an RF transmission. A SND-NR message should be transmitted containing the last two hourly average values. This scheme would provide some redundancy in case some transmission message was lost.

As the range is limited to some 30 meters indoors, the mode T at 868 MHz could be a good solution and compromise of power consumption, antenna size and cost. Using the RC1180-MBUS3 configured as

- T1 mode
- C-field for SND-NR
- Automatic sleep mode after transmission

The application microcontroller should wake up the module (0xFF) and send the application data string as described previously.

Depending on the accuracy requirement for the temperature measurement, a simpler solution could be made based on RC1180-MPC1. This module is autonomous and contains a temperature sensor with an accuracy of +/- 2 degrees. Using this variant would save the external microcontroller and sensor. It would be an instantaneous measurement every time the module does a transmission. In this case we could configure the module to transmit every 30 minutes, giving us some redundancy with respect to hourly data.

- Slave
- T1 mode
- Transmission interval 30 minutes
- Temperature sensor enabled (Device type 0x1B)

Somewhat depending on the building construction and materials, either repeaters or several gateways could be used to ensure full coverage over several floors. Repeaters (RC1180-MBUS3 configured with NETWORK_ROLE = Repeater) should be connected to mains power, and placed in central locations in the building.

Several receivers (gateways) could also be used, to improve the coverage. For example one receiver per floor, connected to mains supply and the building LAN (Ethernet).

A quick estimate of the power consumption suggest that the sensor could run off an 3.6 Volt AA cell (2400 mAh) only limited by the self-discharge of the battery itself:

TX for approx. 5 ms (total 40 bytes) at 37 mA
Sleep mode 0.3 uA
I_average = 0.3 uA + (5ms/30min) x 37 mA = 0.4 uA
Battery lifetime 2400 mAh / 0.4 uA = 6 Mh (which is hundreds of years)

Example 2: Long range outdoor sensor monitoring

In this second example we consider an application with 30 gas and particle sensors in an urban environment. We would like to monitor these sensors every hour. The distance to the sensors would be up to 1 km.

Due to the longer range requirement, we would choose a solution based on 169 MHz using narrowband radio. The Wireless M-Bus mode N would be a good solution providing the required range even in an urban environment with tall buildings.

Using the RC1701HP-MBUS4, the module should be configured for

- Slave
- N1 mode, 2.4 kbps
- C-field for SND-NR
- Automatic sleep mode after transmission

An application controller should read the gas and particle sensor at a 1 hour interval, and wake up the RF module to do the transmission.

In the design of the sensor unit, special attention should be paid to the antenna and the power management. The relatively low frequency (169 MHz) requires an antenna of a certain size to be efficient. Ideally the antenna should be 44 cm to achieve a quarter wave monopole. But this size is very often not practical, and a compromise must be done reducing the size of the antenna. A helical antenna from 5 – 20 cm can be used, yielding a gain of -9 to -3 dB.

Caution must also be taken in the design of the power supply. The 500 mW transmitter uses some 400 mA during transmissions. Even a Lithium battery with high capacity (size AA) may not be able to deliver this high current, and it must therefore be supported by a capacitor (supercap). The typical size of the supercap is 470 mF with low ESR (Equivalent Series Resistance).

In this case we can estimate the battery lifetime:

TX for approx. 100 ms (total 30 bytes) at 400 mA
Sleep mode 2.0 uA
I_average = 2.0 uA + (100ms/60min) x 400 mA = 13 uA
Battery lifetime 2400 mAh / 13 uA = 185 kh (which is over 20 years)

Because of the high peak current, the usable capacity will be limited by the voltage “dip” during transmission (which is reduced by the supercap). But still we see that the battery lifetime can be many years. Radiocrafts has implemented similar solutions for water meters with a lifetime of more than 15 years.

Example 3: Parking lot sensors

In this design example we consider a parking lot sensor system that is used to monitor free parking space in a garage with 500 slots. The actual occupancy sensor could be placed over the parking space (using ultrasound), or even beneath the cars in the floor. The sensor should detect and report a free or occupied space within 30 seconds.

In this case there are a large number of sensors in a limited area, so we need to take caution of the radio channel occupancy. The simplest system would use the sensor to check if the space is free every 30 seconds, and then send a message. For such a solution to work reliably with a large number of sensors, we need to consider the number of radio transmission collisions. Fortunately the amount of data is very small, actually only one bit, indicating if the space is free or not. So each transmission can be very short when using a relatively high data rate. In this case we could use mode T or C, and the transmission would be some 3 ms. If 500 sensors transmit every 30 seconds, the total occupancy is 1.5 sec/30sec, or 5%.

A system which simply transmits a data packet irrespective of whether some other transmission is taking place at the same time is known as the Aloha protocol for time-division multiple access (TDMA). To avoid consecutive collisions the transmission should be done with some randomization of the transmission timing. In our example we assume a random transmission time, but with an average of one transmission every 30 seconds. The probability of a transmission taking place when another transmission is ongoing, leading to a collision, is described by the Poisson distribution. Using the example above, with 5% channel occupancy rate, we will find that the probability for a successful transmission is 80%. Or 20% of packets are lost due to collisions. That is, using this scheme we would have to accept that every fifth packet is lost.

A somewhat more intelligent system, could report a change in occupancy, thereby reducing the traffic and collision rate substantially. That is, transmitting a message only when a car arrived, or left the parking space. A two-way protocol requiring a reception acknowledgment from the receiver would also increase the reliability of the system. If the message was not acknowledges, the sensor should retransmit the message after a short random delay.

Such a system, based on RC1180-MBUS3, would be configured to

- Slave
- T2 mode for two way communication
- Automatic sleep mode after transmission and reception

A garage full of cars is not a very good environment for radio signals. To further improve the reliability of the system, it would be advisable to make use of several gateways to ensure reception of the messages.

An interesting variant of this system is a solution for open air parking lots, where the detector should be placed in the ground under the car. In this case the placement of the antenna is a challenge. But it can be solved by placing the antenna in the ground surface, potted in a brick of epoxy, so it can actually be run over by cars without being damaged.

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.

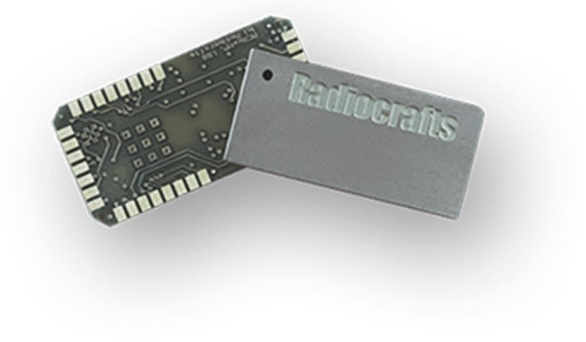
Trademarks

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.



For More Information,
Please Visit Our Website!

www.radiocrafts.com

Email: sales@radiocrafts.com

Tel: +47 4000 5195